ISBN 978-82-471-2461-1 (printed ver.)
ISBN 978-82-471-2462-8 (electronic ver.)
ISSN 1503-8181

Doctoral theses at NTNU, 2010:235

Giancarlo Marafioti

Enhanced Model Predictive Control: Dual Control Approach and State Estimation Issues

NTNU
Norwegian University of
Science and Technology
Thesis for the degree of
philosophiae doctor
Faculty of Information Technology, Mathematics and
Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Doctoral theses at NTNU, 2010:235

NTNU

NTNU
Norwegian University of
Science and Technology

Giancarlo Marafioti

Enhanced Model Predictive Control: Dual Control Approach and State Estimation Issues

Thesis for the degree of philosophiae doctor

Trondheim, November 2010

Norwegian University of
Science and Technology
Faculty of Information Technology, Mathematics and
Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the degree of philosophiae doctor

Faculty of Information Technology, Mathematics
and Electrical Engineering
Department of Engineering Cybernetics

ISBN 978-82-471-2461-1 (printed version)
ISBN 978-82-471-2462-8 (electronic version)
ISSN 1503-8181

ITK Report 2010-15-W

Doctoral Theses at NTNU, 2010:235

Printed by Tapir Uttrykk

Summary

The main contribution of this thesis is the advancement of Model Predictive Control (MPC). MPC is a well known and widely used advanced control technique, which is model-based and capable of handling both input and state/output constraints via receding horizon optimization methods. The complex structure of MPC is delineated, and it is shown how improvements of some of its components are able to enhance overall MPC performance.

In more detail, in Chapter 3, the definition of a state dependent input weight, in the cost function, shows satisfactory controller performance for a large region of working conditions, compared to a standard MPC formulation. The robustness of the optimization problem is improved, i.e., this particular weight configuration yields a 'better' conditioned problem than the standard MPC will produce. This is demonstrated by a simulation study of a particular Autonomous Underwater Vehicle.

Chapter 4 presents a novel formulation named Persistently Exciting Model Predictive Control (PE-MPC). The fundamental idea is that because of its use of a model, MPC should be amenable to adaptive implementation and on-line tuning of the model. Such an approach requires guaranteeing certain signal properties, known as 'persistent excitation', to ensure uniform identifiability of the model. An approach to augment the input constraint set of MPC to provide this guarantee is proposed. It is also shown, how this formulation has properties that are typical of Dual Control problems. Theoretical results are proven and simulation-based examples are used to confirm the strength of this new approach.

Chapter 5 takes state estimation issues into account. Kalman filtering for nonlinear systems is described, and throughout the chapter, a comparison of the more recent Unscented Kalman Filter (UKF), with the well known Extended Kalman Filter (EKF) is given. Two examples are presented, first a photobioreactor for algae production, and then a simple but effective example of locally weakly unobservable nonlinear system. In both cases, simulation-based results prove the advantages of using UKF as an alternative to EKF.

Summary

Preface

This thesis is submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at the Norwegian University of Science and Technology (NTNU).

The research presented is the result of my doctoral studies, mostly carried out at the Department of Engineering Cybernetics (NTNU) in the period August 2006 - May 2010. During this time I was privileged to stay at other research institutions. In September 2008 - February 2009 I was a scholar at the Department of Mechanical and Aerospace Engineering at the University of California San Diego (UCSD). In April 2008, and in April 2009, I visited the Department of Automatic Control at the Ecole Supérieur d'Electricité (Supélec) in Gif-sur-Yvette, France.

Acknowledgment

Most of all, I would like to thank my supervisor Professor Morten Hovd, who has given me the opportunity to take PhD studies in the beautiful city of Trondheim. He has been of great support and trust during the entire period of my studies. He provided me with a good working environment, and he was always helpful in any circumstances, it was a great support to know that I could always rely on him. He also introduced me to the world of academic research, and guided me through the various stages.

I would like also to thank Professor Robert R. Bitmead for taking good care of me during my stay at UCSD, and especially for guiding me during my research period at his department. With his admirable experience and knowledge he has contributed to my research and on improving my skills as researcher and person.

I am grateful to all my coauthors, it has been stimulating to work with all of them. Their remarks, ideas, and all discussions we have had, have contributed to my research. I would like also to thank my former office mate Steinar Kolås for sharing his experience and knowledge with me.

I am glad to have had many interesting colleagues at the departments I have worked/ visited. They all have contributed on increasing my knowledge and I have particularly enjoyed all the discussions during lunch and coffee breaks. In particular I am very happy I shared time and experiences with my colleagues and friends, Francesco Scibilia, Milan

Milovanovic, Luca Pivano and Ilaria, Roberto Galeazzi, Gullik A. Jensen, Johannes Tjønnås, Mernout Burger, Aksel A. Transeth, Esten I. Grøtli, Selvanathan Sivalingam, Hardy B. Siahaan. They made my life at the department, and around Norway memorable.

I acknowledge the assistance from Stewart Clark at NTNU for editing this thesis.

Many thanks also to the department secretaries, Tove, Unni, Eva, and Bente for their appreciable help on all the not 'so friendly' bureaucracy. Thanks also to Stefano Bertelli for organizing the challenging cage-ball training, it is nice that he cares about the physical condition of the department's employees.

I am grateful to the Research Council of Norway for providing the funding necessary for my research.

Finally, I am warmly grateful to my parents, they have always trusted and supported me, in all my choices, although some decision I made resulted in moving away from them.

Un caloroso ringraziamento alla mia famiglia che ha sempre creduto in me e mi ha supportato in tutte le mie scelte, anche se alcune hanno significato l'allontanarsi da loro per andare a vivere lontano dalla mia terra.

Trondheim, August 2010 *Giancarlo Marafioti*

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis presents an investigation of Model Predictive Control (MPC) and some related issues. MPC has found widespread application in industry, in particular in the chemical processing industries. The main objective of this thesis is to contribute to further enlarge the application area of MPC. In order to put the contributions of the thesis in context, and make it easier for the reader to appreciate these contributions, background material covering the fundamental ideas, general classifications and some properties of MPC are presented. This thesis touches three large branches of control theory, i.e., Model Predictive Control, Dual Control, and State Estimation. Thus it is practically impossible to present a comprehensive exposition of those control areas of interest. However, in Chapter 2, and the first sections in Chapters 3 and 5, background results, general formulations, technical definitions and concepts are given. In addition, the references to the literature given should be suitable as starting points for more in-depth examinations of background concepts and development.

In the following part of this chapter, the main contributions, thesis structure and associated publications are indicated.

1.1 Motivation

Due to the great success of MPC, see for example Qin & Badgwell [2003], the demand of applying this advanced control technique to a larger set of plants is constantly increasing. In Chapter 2, and references therein, some of the most important advantages of this control approach are described.

The results obtained in this thesis, are inspired by the desire to extend MPC applications to new areas, and in general to improve its performance. Therefore, an attempt has been made to enhance MPC, by exploiting its structure and properties.

Figure 1.1 presents a general structure of MPC. Clearly, there are several parts (shown as blocks) which are interconnected (as illustrated by the arrows), resulting in a complex

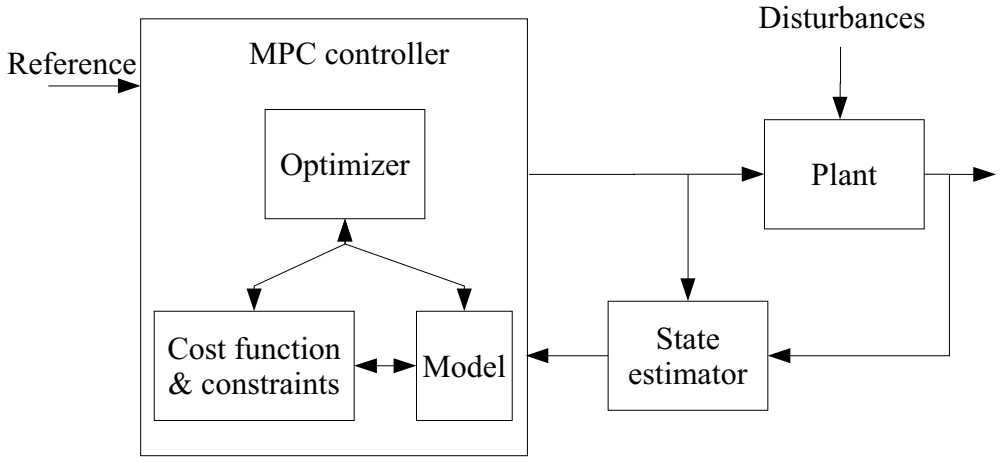


Figure 1.1: General MPC structure.

structure. In general, it is desired to control a specific plant by manipulating its inputs and using its outputs to gather state information. However, an external disturbance may affect plant operations. A controller, based on a model of the plant, tries to optimize a specific control cost function, while satisfying a set of constraints and following a given reference. The model is generally unable to exactly represent the plant dynamics, thus a receding horizon is introduced, and the control function is re-optimized at every time step, as means to obtain some feedback. This is necessary to reduce the effects of model/plant mismatch, and also to counteract the effect of unknown disturbances. Also, it is likely that the measured plant output does not contain all information needed for the controller, thus a state estimator is often used to overcome this problem.

To enhance MPC performance, and extend its range of application, focus is mainly placed on the 'cost function & constraints', 'model', and 'state estimator' blocks of Figure 1.1. More in detail, the results obtained in Chapter 3, enhance the MPC cost function to increase the region of operation where the controller has satisfactory performance. This is desired, and in the past gain-scheduling techniques were developed to obtain similar results. The particular nonlinear input weight introduced, that depends on the state of the system, has also benefits for the implicit robustness of the problem formulation with respect to numerical solutions. Finally, the chapter illustrates the use of different model representations in the MPC formulation.

In Chapter 4 the constraint set is properly augmented to incorporate dual control features. This allows the particular MPC to work in an adaptive context, where 'learning' algorithms may be implemented to estimate, and update the MPC model. Finally, in

Chapter 5, the focus is on state estimation, and it is shown how improving this part of the system has direct impact on the proper control operation. If the estimate has poor quality, the controller may be not able to achieve its task.

1.2 Contribution

The main contributions in this thesis are:

- The state dependent input weight introduction for MPC applied to an Autonomous Underwater Vehicle (AUV). The standard MPC formulation is enhanced by the introduction of a state dependent cost function input weight. This has similarity to gain-scheduling controllers, where control of nonlinear systems is obtained by several sub-controllers, designed to provide satisfactory performance, each one in a sub-set of operating conditions. The AUV example also results in an ill-conditioned optimization problem. The introduction of the novel input weight has benefits on the Hessian condition number, improving robustness for optimization algorithms.

- The formulation of Persistently Exciting Model Predictive Control (PE-MPC), to deal with parameter estimation in an adaptive context. In general, when adaption is required in closed-loop systems, persistence of excitation becomes an issue due to the conflict between the control and adaption actions. PE-MPC includes both actions in its formulation, yielding a control signal that is obtained as a trade-off between these two considerations. This is a feature that is present in dual control problems.

- Methods are studied for state estimations in nonlinear systems. The commonly used Extended Kalman Filter is compared to the more recently developed Unscented Kalman Filter. The first comparison scenario is the implementation of both filter in a photobioreactor that produces microalgae. The operational data are available and are used to validate the results. The second comparison scenario consists of a simulation study of a *locally weakly unobservable* system. In both examples the UKF is found to give superior performance.

1.3 Thesis Organization

The remainder of the thesis is organized as follows:

- *Chapter 2* - A general introduction to Model Predictive Control is presented. A historical discussion is given about its origins and the receding horizon principle. For linear state space models, and for Finite Impulse Response (FIR) models, standard MPC formulations, and their equivalent Quadratic Programming problems are presented. Thereafter, there is a description of a general Nonlinear MPC formulation (NMPC), and the classification into Convex or Non-convex optimization problems. Explicit MPC is briefly introduced. These formulations are important because are used in the subsequent chapters to present the main contributions.

- *Chapter 3* - A novel MPC formulation with state dependent input weight in the cost function is given. This is analyzed in a simulation study controlling an Autonomous Underwater Vehicle (AUV). Its nonlinear model description is presented. Several linear model approximations are obtained. With respect to the AUV example a state dependent input weight is defined, and its application benefits are shown. Combining different model approximations, the standard MPC, and the particular input weight, six different controllers are obtained. Their performance are compared by simulation, and finally discussed from a robust MPC point of view.

- *Chapter 4* - A novel MPC formulation, named Persistently Exciting Model Predictive Control, is presented. The MPC problem is discussed in an adaptive context, where data quality requirements for model adjustment are built into the MPC formulation. The relationship to the dual control problem is explained. Results on dual control theory, and the Persistence of Excitation Condition (PEC) are considered fundamental for understanding the context of the new formulation. Therefore they are given in a background section. The proposed approach augments the input constraint set of MPC to guarantee a sufficient excitation level such that adaptive estimation algorithms can be used to estimate and update the model uncertain parameters. Finally, an example using FIR models and the Recursive Least Square algorithm for parameter estimation is presented.

- *Chapter 5* - The state estimation problem is discussed. This mainly focuses on Kalman filtering for nonlinear systems. Two popular nonlinear estimators, the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF) are compared in two different examples. The first performance comparison is presented by estimating the state of a photobioreactor for microalgae production. The obtained results are then validated using plant operational data. The second filter perfor-

mance comparison is obtained in an NMPC scenario defined by a nonlinear system with observability issues. Simulation-based examples show the advantage of using UKF over EKF.

- *Chapter 6* - General conclusions, and recommendations for further work are given.

1.3.1 Notation

In this thesis, conventional notation for control theory is used. For instance, scalars will be denoted by plain symbols, vectors by lowercase bold symbols, matrices by uppercase bold symbols, vector/matrix transpose by $(\cdot)^T$, positive definiteness by $(\cdot) > 0$ or negative definiteness by $(\cdot) < 0$, the n-dimensional identity matrix, by $\boldsymbol{I}_n$. Through the exposition of results, when new notation is introduced, then it is properly defined. For ease of consultation, some of the most used notation is listed in Table 1.1.

Table 1.1: Notation

Symbol	Representation
s, S	Scalar, plain character
$\boldsymbol{v}, \boldsymbol{M}$	Vector or matrix, bold character
$(\cdot)^T$	Vector or matrix transpose
$(\cdot) > 0$	Positive definite for matrices, positive for scalars
$(\cdot) < 0$	Negative definite for matrices, negative for scalars
I_n	Identity matrix of $(n \times n)$ dimensions

1.4 List of publications

Most of the material contained in this thesis has been either published or recently submitted for publication. The following is the list of publications related to the work presented in this thesis.

Book chapter

- [Marafioti et al., 2009a] - G. Marafioti, S. Olaru, M. Hovd. State Estimation in Nonlinear Model Predictive Control, Unscented Kalman Filter Advantages. In *Nonlinear Model Predictive Control Towards New Challenging Applications Series: Lecture Notes in Control and Information Sciences*, Vol. 384 - L. Magni, D. M. Raimondo, F. Allgöwer (Eds.), 2009 - ISBN: 978-3-642-01093-4.

Journal papers

- [Marafioti et al., 2010a] - G. Marafioti, R.R. Bitmead, M. Hovd. Persistently Exciting Model Predictive Control. Submitted to *International Journal of Adaptive Control and Signal Processing*.

Conference papers

- [Marafioti et al., 2010b] - G. Marafioti, R.R. Bitmead, M. Hovd. Persistently Exciting Model Predictive Control using FIR models. In *Proceedings of Cybernetics and Informatics International Conference*, Vyšná Boca, Slovak Republic, February 10-13, 2010.

- [Marafioti et al., 2009b] - G. Marafioti, S. Tebbani, D. Beauvois, G. Becerra, A. Isambert, M. Hovd. Unscented Kalman Filter state and parameter estimation in a photobioreactor for microalgae production. In *Proceedings of International Symposium on Advanced Control of Chemical Processes*, Koç University, Istanbul, Turkey, July 12-15, 2009.

- [Marafioti et al., 2008b] - G. Marafioti, S. Olaru, M. Hovd. State Estimation in Nonlinear Model Predictive Control, Unscented Kalman Filter Advantages. In *Proceedings of the International Workshop on Assessment and Future Directions of Nonlinear Model Predictive Control*, Pavia, Italy, September 5-9, 2008.

- [Marafioti et al., 2008a] - G. Marafioti, R.R. Bitmead, M. Hovd. Model Predictive Control with State Dependent Input Weight: an Application to Underwater Vehicles. In *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, July 6-11, 2008.

The results, either published or submitted, are partially or fully related to the thesis chapters as follow.

- Chapter 3: Marafioti et al. [2008a]

- Chapter 4: Marafioti et al. [2010a,b]

- Chapter 5: Marafioti et al. [2008b, 2009a,b]

Chapter 2

An introduction to Model Predictive Control

Model Predictive Control (MPC), also referred as receding horizon control or moving horizon control, is an advanced control technique. In more detail, it is an optimal control procedure that easily allows engineers to enter constraints directly into the control problem formulation. Furthermore, it explicitly uses a process model to predict the future response of a plant. In general, MPC is applied in a discrete time framework, although in the literature continuous time formulations are available. At each control interval, the future plant behavior is optimized according to some cost criterium, and an associated constraint set. As result of the optimization, a sequence of manipulated variables is obtained. Thus, the first element of this sequence is sent into the plant, and the entire computation is repeated at subsequent control intervals.

MPC-based technology was originally developed for controlling petroleum refineries and power plants. Nowadays, it is possible to find this technology in a wide range of applications, such as chemical industries, food processing, automotive, and aerospace industries. The development of MPC techniques is due to the increasing research effort from both the academic community and industry. Thus, by understanding MPC properties, it is possible to improve performance, to add new features, and expand applicability.

The importance of taking constraints into account arises from the reason that most of real world systems are subject to physical constraints. Thus, the ability of handling constraints in a straightforward way makes MPC very attractive. For instance, this is particularly helpful in process industries where the most profitable operations are often obtained when running at one or more constraints. Constraints are often associated with available range of actuators, safe operating regions, product quality specifications. For example, when manufacturing a product that requires heat, being able to minimize the energy needed for heating can reduce the production cost, however a certain amount of heat is needed to guarantee that the final product is correctly manufactured. This defines a

quality specification that the addition of heat is intended to ensure. Therefore, this may be represented by a constraint. For a general plant, there may be input constraints (saturation on actuators), state and/or output constraints (safety limits on pressure or temperatures), and they may be represented as inequality or equality and then included into MPC formulation.

Another interesting feature of MPC is the adoption of a model to predict the future plant behavior. These models are obtained from simple plant tests, or by applying some system identification technique to the plant operational data. However, the effort of designing better plant models, the increasing availability of computational power, the improvement of optimization algorithms, the constant pursuit of better performance, all contribute to increasing the trend of using first-principle models. In general these are nonlinear models obtained directly from the application of well established laws of physics and chemistry. For instance, in a chemical plant they can be derived starting from the chemical transformations and reactions occurring inside a process.

In a fairly recent survey on industrial MPC technology (Qin & Badgwell [2003]), the authors claim that there are thousands of successful applications, and this number is still increasing. An interesting introduction, with a complete mathematical formulation, is given in Findeisen & Allgöwer [2002], where theoretical, computational, and implementation aspects are discussed. A less mathematically formal introduction, but comprehensive and straightforward to understand is Camacho & Bordons [2007]. Several books have been written about linear MPC. Some of these are Maciejowski [2002], Rossiter [2003], Camacho & Bordons [2004]. They describe and analyze MPC theory and its application in slightly different ways. In Maciejowski [2002] more effort is spent on how to handle constraints, and the author starts from a simple formulation adding step by step more details such as integral effect and constraint softening. In Rossiter [2003] MPC is viewed from a practical approach, avoiding jargon, and giving interesting examples about MPC tuning and numerical issues. Finally, in Camacho & Bordons [2004] an extensive review of MPC controllers is given, with focus on a variant of MPC known as Generalized Predictive Control. The authors attempt to reduce the distance between the way practitioners use control algorithms and the more abstract way researchers formulate the control technique.

A more recent book, and perhaps the one that can be considered the most complete, is Rawlings & Mayne [2009]. There, it is finally possible to find a comprehensive and fundamental analysis of MPC theory and design, with an in-depth examination and comparison of the most used, and recent, state estimation techniques.

This chapter gives a general MPC description. A brief historical introduction is presented first. Then starting with the well known linear quadratic regulation problem, the concept of a receding horizon is introduced as way to introduce feedback. Finally, general MPC formulations are given, and used in the remaining part of the thesis, where several problems of relevance to MPC are addressed. It is assumed that that the inputs are con-

tinuously valued (in space). Moreover, the focus in this chapter is on quadratic objective functions and linear constraints, because this particular MPC is predominant in industry and it is related to the LQR problem.

2.1 Origins

The control action of MPC is determined as the result of an online optimization problem. As consequence, the behavior of an MPC controller may be rather complex. The earliest MPC implementations were limited by the low computational resource availability. Although only steady-state models were used, depending on the plant dimension, several hours could be needed for the optimization problem solution.

The first ideas on receding horizon control and model predictive control can be traced back to the 1960s [García et al., 1989]. In Propoi [1963] the receding horizon approach was proposed. However, real interest in this innovative approach started later, in the 1980s, after the first publications on Model Algorithmic Control (MAC) [Richalet et al., 1978], Dynamic Matrix Control (DMC) [Cutler & Ramaker, 1979, 1980], and a complete description on Generalized Predictive Control (GPC) [Clarke et al., 1987a,b].

DMC was developed mostly for the oil and chemical industries. Its goal was to tackle multivariable constrained control problems. In fact, the previous approach to these problems, was to solve them by integrating several single loop controllers augmented by many time-delay compensators, overrides, selectors, etc. With DMC, a time domain model, either finite impulse or step response, was used to predict the plant behavior. The original DMC formulation was completely deterministic with no explicit disturbance model. GPC formulation was designed as a new adaptive control scheme, where transfer function models were adopted. Since it does not come naturally to use GPC with multivariable constrained systems, DMC was adopted by most of the oil and chemical companies.

As mentioned in Qin & Badgwell [2003], the success of DMC in the oil and chemical industries attracted the academic community. Therefore, primordial research on MPC was intended as an attempt to understand DMC.

2.2 The Receding Horizon Principle

Before discussing the receding horizon principle, let us briefly introduce the well known Linear Quadratic Regulation (LQR) problem for discrete systems, [Naidu, 2003]. Simply

stated, the LQR problem is

$$\min_{\boldsymbol{x},\boldsymbol{u}} \sum_{i=0}^{N} \boldsymbol{x}_i^T \boldsymbol{Q} \boldsymbol{x}_i + \boldsymbol{u}_i^T \boldsymbol{R} \boldsymbol{u}_i \tag{2.1a}$$

$$\boldsymbol{x}_{i+1} = \boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{B}\boldsymbol{u}_i, \quad \text{given } \boldsymbol{x}_0, \tag{2.1b}$$

where the optimization horizon N may be either finite or infinite. It is possible to write the optimal solution of (2.1) in the feedback form

$$\boldsymbol{u}_i^{opt} = \boldsymbol{K}_i \boldsymbol{x}_i^{opt}, \quad i = 1, \ldots, N \tag{2.2}$$

where the state feedback gain $\boldsymbol{K}_i$ is calculated from the corresponding Riccati equation. For example, for an infinite horizon LQR the optimal state feedback gain is

$$\boldsymbol{K}_\infty = -\left(\boldsymbol{R} + \boldsymbol{B}^T \boldsymbol{P} \boldsymbol{B}\right)^{-1} \boldsymbol{B}^T \boldsymbol{P} \boldsymbol{A}, \tag{2.3}$$

where $\boldsymbol{P}$ is the unique positive definite solution to the discrete time algebraic Riccati equation

$$\boldsymbol{P} = \boldsymbol{Q} + \boldsymbol{A}^T \left(\boldsymbol{P} - \boldsymbol{P}\boldsymbol{B}\left(\boldsymbol{R} + \boldsymbol{B}^T \boldsymbol{P} \boldsymbol{B}\right)^{-1} \boldsymbol{B}^T \boldsymbol{P}\right) \boldsymbol{A}. \tag{2.4}$$

Now, if we add inequality constraints to (2.1), it is not possible to find a closed-form solution such as (2.2). However, online optimization, and a finite length horizon can be a strategy to overcome this problem.

The idea is to re-formulate the constrained, infinite horizon, optimization problem as a constrained, finite horizon, optimization problem. This may be done by representing the 'tail' of the infinite horizon as a 'cost to go' term to be added to the existing cost function.

Note that this re-formulation results in an open-loop optimization, therefore if model errors and/or disturbances occur it is necessary to introduce some feedback technique to compensate for both. The function of receding (or moving) horizon is to introduce feedback. This finite horizon, which shifts as the time passes, is the basic idea of any MPC. Figure 2.1 illustrates this principle. Given the last measurement available at the present time k, the controller predicts the dynamic behavior of plant, over a prediction horizon N_p. This is based on an internal model of the system. To introduce feedback only the first element of the predicted optimal input sequence is applied to the plant until the next measurement is available. Then, horizons are shifted one step forward, and a new optimization problem is formulated and solved. Note in Figure 2.1, the difference between the set-point and the reference trajectory. A set-point is a constant reference that the output should ideally follow. Instead, a reference trajectory defines how the plant should reach the set-point from the current output.

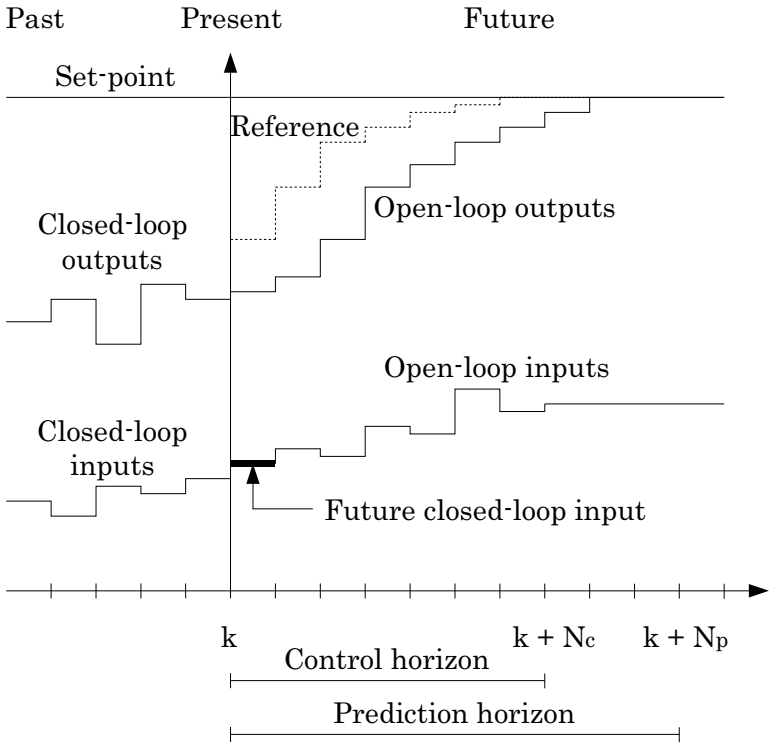


Figure 2.1: Receding Horizon principle.

Some considerations are needed regarding the model used for prediction. As mentioned for example in Bemporad [2006], it is necessary to observe that models are obtained as a tradeoff between their accuracy and the complexity of resultant optimizations. In fact, while models used for simulations tend to have the best accuracy to numerically reproduce the behavior with the minimum error, prediction models are usually more simple, but yet they are able to capture the main dynamics of the system.

Nowadays, an essential MPC model distinction must be made between linear and nonlinear models. The application of the first types result in linear MPC formulations, while the second types produce Nonlinear MPC (NMPC) formulations. In general, nonlinear models are more complex and the associated optimization problem solution is computationally demanding. Moreover, the most important intention of this classification is that in general when a nonlinear model is used the optimization loses its convexity property. Thus, NMPC is a non-convex problem, while MPC is a convex one. The implications of this distinction are discussed briefly in a succeeding section in this chapter.

In the following section, a detailed MPC formulation is presented and its properties are given.

2.3 Formulations

2.3.1 Linear Model Predictive Control

There are several linear MPC formulations in the literature. In this section, two very common discrete time formulations are described. The first one uses state space models, while the second one uses finite impulse response models for prediction.

Linear MPC is used in process control even if the process to control is nonlinear. The reason is that when the process is working in the neighborhood of some equilibrium point, a linear model is often a good approximation of its dynamics. In such applications, MPC is used primarily to reject disturbances, while optimizing the operation of the plant in some sense.

State space models

Consider the following discrete state space model

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \tag{2.5}$$

for prediction of the open-loop system, where $\boldsymbol{x}$ is the state vector, $\boldsymbol{u}$ is the input vector, and k is the time step index. Using (2.5) the MPC finite-time optimal control problem is

$$\min_{\bar{\boldsymbol{u}}} J(k, \bar{\boldsymbol{u}}) \quad = \boldsymbol{x}_{k+N_p}^T \boldsymbol{S} \boldsymbol{x}_{k+N_p} + \sum_{j=0}^{N_p-1} \boldsymbol{x}_{k+j}^T \boldsymbol{Q} \boldsymbol{x}_{k+j} + {\boldsymbol{u}_{k+j}^k}^T \boldsymbol{R} \boldsymbol{u}_{k+j}^k \tag{2.6a}$$

$$s.t. \qquad \boldsymbol{x}_{k+1+j} = \boldsymbol{A}\boldsymbol{x}_{k+j} + \boldsymbol{B}\boldsymbol{u}_{k+j}^k \,, \quad \text{from } \boldsymbol{x}_k, \qquad j = 0, \ldots, N_p - 1 \tag{2.6b}$$

$$\boldsymbol{u}_{min} \leq \boldsymbol{u}_{k+j}^k \leq \boldsymbol{u}_{max} \,, \qquad j = 0, \ldots, N_p - 1 \tag{2.6c}$$

$$\boldsymbol{x}_{min} \leq \boldsymbol{x}_{k+j} \leq \boldsymbol{x}_{max} \,, \qquad j = 1, \ldots, N_p \tag{2.6d}$$

$$\boldsymbol{l}_{min} \leq L(\boldsymbol{x}_{k+j}) \leq \boldsymbol{l}_{max} \,, \qquad j = 1, \ldots, N_p \tag{2.6e}$$

where N_p is the prediction horizon, $\bar{\boldsymbol{u}}_k = [{\boldsymbol{u}_k^k}^T, {\boldsymbol{u}_{k+1}^k}^T, \ldots, {\boldsymbol{u}_{k+N_p-1}^k}^T]^T$ is the sequence of manipulate variables or inputs, $\boldsymbol{S} = \boldsymbol{S}^T \geq 0$, $\boldsymbol{Q} = \boldsymbol{Q}^T \geq 0$, $\boldsymbol{R} = \boldsymbol{R}^T > 0$ are weight matrices, of appropriate dimensions, defining the performance index, and $\boldsymbol{u}_{min}$, $\boldsymbol{u}_{max}$, $\boldsymbol{x}_{min}$, and $\boldsymbol{x}_{max}$ are vectors of appropriate dimensions defining input, state, constraints, respectively. Finally, $L(\boldsymbol{x})$ and $\boldsymbol{l}_{min}$, $\boldsymbol{l}_{max}$ indicate the possibility to have constraints on linear combinations of states.

The state at time k can be written as a function of the initial state $\boldsymbol{x}_0$ and the previous inputs $\boldsymbol{u}_{k-1-i}$ $(i = 0, 1, \ldots, k-1)$

$$\boldsymbol{x}_k = \boldsymbol{A}\boldsymbol{x}_0 + \sum_{i=0}^{k-1} \boldsymbol{A}^i \boldsymbol{B} \boldsymbol{u}_{k-1-i}. \tag{2.7}$$

Thus, now (2.6) can be reformulated as the following QP problem

$$\bar{\boldsymbol{u}}^*(\boldsymbol{x}_0) \quad = \quad \arg\min_{\bar{\boldsymbol{u}}} \quad \frac{1}{2}\bar{\boldsymbol{u}}^T \boldsymbol{H} \bar{\boldsymbol{u}} + \boldsymbol{x}_0^T \boldsymbol{G}^T \bar{\boldsymbol{u}} + \frac{1}{2}\boldsymbol{x}_0^T \boldsymbol{Y} \boldsymbol{x}_0 \tag{2.8a}$$

$$s.t. \qquad \boldsymbol{L}\bar{\boldsymbol{u}} \leq \boldsymbol{W} + \boldsymbol{V}\boldsymbol{x}_0, \tag{2.8b}$$

where

$$\bar{\boldsymbol{u}}^*(\boldsymbol{x}_0) = \left[\boldsymbol{u}_0^{T*}, \ldots, \boldsymbol{u}_{N_p-1}^{T*}\right]^T \tag{2.9}$$

is the optimal solution, and $\boldsymbol{H} = \boldsymbol{H}^T \geq 0$, $\boldsymbol{G}$, $\boldsymbol{Y}$, $\boldsymbol{L}$, $\boldsymbol{W}$, and $\boldsymbol{V}$ are matrices of appropriate dimensions (see Bemporad et al. [2004]). Note that the term $\frac{1}{2}\boldsymbol{x}_0^T \boldsymbol{Y} \boldsymbol{x}_0$ can be dropped since it does not affect the optimization result. Simply, the difference is that the

QP cost function value will be different from the one in (2.8a), but in MPC, but the value of u at the optimal solution will be identical.

At every iteration k, the MPC algorithm consists of measuring or estimating the state $\boldsymbol{x}_0$, then solving the QP problem (2.8), and finally applying only the first element of the optimal sequence (2.9), $\boldsymbol{u}_k = \boldsymbol{u}_0^*(\boldsymbol{x}_0)$, to the process. Then the procedure is repeated for the subsequent time step $k+1$.

It is possible to extend (2.6) in several ways. For example, in order to obtain a tracking formulation such that an output vector $\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k$ follows a certain reference signal $\boldsymbol{r}_k$, the cost function (2.6a) may be replaced by

$$\sum_{j=0}^{N_p-1} (\boldsymbol{y}_{k+j} - \boldsymbol{r}_{k+j})^T \boldsymbol{Q}_y (\boldsymbol{y}_{k+j} - \boldsymbol{r}_{k+j}) + \Delta \boldsymbol{u}_{k+j}^k{}^T R \Delta \boldsymbol{u}_{k+j}^k, \tag{2.10}$$

where $\boldsymbol{Q}_y = \boldsymbol{Q}_y^T \geq 0$ is a matrix of output weights, and $\Delta \boldsymbol{u}_k^k \triangleq \boldsymbol{u}_k^k - \boldsymbol{u}_{k-1}^k$ is the new optimization variable, possibly constrained as $\Delta \boldsymbol{u}_{min} \leq \Delta \boldsymbol{u}_k^k \leq \Delta \boldsymbol{u}_{max}$. As for the QP formulation, it is sufficient to define the tracking vector $\left[\boldsymbol{x}_k^T,\ \boldsymbol{r}_k^T,\ \boldsymbol{u}_{k-1}^k{}^T\right]^T$ and use it in the same way in (2.7).

If it is necessary to reduce the optimization complexity, an MPC with multiple horizons may be used. It is possible to have a prediction horizon N_P, and a control horizon $N_c \leq N_p$. A shorter control horizon reduces the number of optimization variables, yielding a smaller optimization problem. Similarly, forcing the input(s) to be constant over several time steps (so-called 'input blocking') also reduces the size of the optimization problem.

A problem that may occur with MPC is that the optimization problem is infeasible. The feasibility of MPC is connected with constraint satisfaction. That is, a constraint set defines the space where candidate solutions (feasible solutions) may be searched to find the optimal solution with respect to the cost function, and outside the constraint set there exists no solution satisfying all constraints. An MPC may become infeasible for several reasons. For instance, a large disturbance may occur and as consequence it may be not possible to keep the plant within the specified constraints. Moreover, it is also difficult to anticipate when an MPC becomes infeasible, thus a strategy to recover from infeasibility is essential for practical implementations.

Constraint softening may be used as an expedient to deal with infeasibility. For example, to implement output constraint softening, it is necessary to modify the output constraint as follows

$$\boldsymbol{y}_{min} - \epsilon \boldsymbol{v}_{min} \leq \boldsymbol{y}_{k+j} \leq \boldsymbol{y}_{max} + \epsilon \boldsymbol{v}_{max} \tag{2.11}$$

where ϵ is a slack variable introduced to allow constraint violation, $\boldsymbol{v}_{min}$ and $\boldsymbol{v}_{max}$ are vectors with nonnegative elements. There are also terms (linear and/or quadratic) that

are added to the cost function in order to penalize the constraint violation. Typically, these terms are designed to ensure that any constraint violation is kept small. The additional terms in the cost function, together with the magnitude of the elements in v_{min}, v_{max}, determine how soft the corresponding constraint will be. Note that MPC with soft constraints as (2.11) can be still formulated as a QP problem.

Other techniques for dealing with infeasibility are to remove the constraints at the beginning of the horizon [Rawlings & Muske, 1993]. When input constraints represent a physical limitation, only state constraints can be softened. However, not all constraints may be relaxed at the same way, and generally softening one constraint instead of another may have less effect on the overall MPC performance. To prioritize which constraint has to be softened in an optimal fashion, it is possible to apply more complex and effective methods as in Scokaert & Rawlings [1999] or in Vada et al. [1999].

FIR models

It is well known that stable processes can be represented by Finite Impulse Response (FIR) models. For a Single-Input Single-Output (SISO) system, the FIR model is

$$y_k = \boldsymbol{\phi}_k \boldsymbol{\theta}^T + v_k, \tag{2.12}$$

where $\{v_k\}$ is a Gaussian measurement noise sequence, $\boldsymbol{\theta}^T = \begin{bmatrix}\theta_1 & \theta_2 & \dots & \theta_n\end{bmatrix}$ is the vector of impulse coefficients, the regressor vector is $\boldsymbol{\phi}_k = \begin{bmatrix}u_{k-1} & u_{k-2} & \dots & u_{k-n}\end{bmatrix}^T$ with $\{u_{k-i}\}_{i=1}^n$ past n inputs to the process, finally n is the number of Markov parameters.

The FIR-based MPC may be formulated as

$$\min_{\bar{\boldsymbol{u}}_k} J(k, \bar{\boldsymbol{u}}) \quad = \frac{1}{2} \sum_{j=0}^{N_p-1} \parallel y_{k+j+1} \parallel_Q^2 + \parallel u_{k+j}^k \parallel_R^2, \tag{2.13a}$$

$$s.t. \quad y_{k+j} = \boldsymbol{\phi}_{k+j} \hat{\boldsymbol{\theta}}_k^T, \qquad j = 1, \dots, N_p, \tag{2.13b}$$

$$u_{\min} \leq u_{k+j}^k \leq u_{\max}, \qquad j = 0, \dots, N_p - 1, \tag{2.13c}$$

$$y_{\min} \leq y_{k+j} \leq y_{\max}, \qquad j = 1, \dots, N_p, \tag{2.13d}$$

where $\bar{\boldsymbol{u}}_k = \begin{bmatrix}u_k^k & u_{k+1}^k & \dots & u_{k+N_p-1}^k\end{bmatrix}^T$ is the control sequence vector, and Q and R are the cost function output and input weights, respectively.

Clearly, the solution of (2.13) may be found by converting the problem into an equivalent Quadratic Programming (QP) problem. QP problems (with positive definite Hessian matrices) are known to be convex, and thus have a unique unique global optimum. Efficient and reliable algorithms are available to solve them. Straightforward manipulation

yield the standard QP formulation

$$\min_{\boldsymbol{U}} \quad \frac{1}{2}\boldsymbol{U}'\boldsymbol{H}\boldsymbol{U} + \boldsymbol{g}^T\boldsymbol{U} \tag{2.14a}$$

$$s.t. \quad \boldsymbol{U}_{min} \leq \boldsymbol{U} \leq \boldsymbol{U}_{max} \tag{2.14b}$$

where $\boldsymbol{H}$ is the Hessian matrix, and $\boldsymbol{g}$ the gradient vector.

2.3.2 Nonlinear Model Predictive Control

Assume that

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\omega}_k; \boldsymbol{\theta}) \tag{2.15a}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{v}_k; \boldsymbol{\theta}) \tag{2.15b}$$

is the plant to be controlled, where $\boldsymbol{x}_k \in \mathcal{R}^n$, $\boldsymbol{u}_k \in \mathcal{R}^m$, $\boldsymbol{y}_k \in \mathcal{R}^p$, $\boldsymbol{\omega}_k$ and $\boldsymbol{v}_k$ are sequences of independent zero-mean random variables, and $\boldsymbol{\theta} \in \mathcal{R}^N$ denotes the vector of unknown plant parameters.

Let $\bar{\boldsymbol{u}}_k = [\boldsymbol{u}_k^{kT}, \boldsymbol{u}_{k+1}^{k\ T}, \ldots, \boldsymbol{u}_{k+N_p-1}^{k\ T}]^T$ be the control sequence vector where N_p is the prediction horizon assumed coincident with the control horizon for simplicity. The standard Model Predictive Control is specified by solving the following finite horizon optimal control problem posed at time k from plant state value $\boldsymbol{x}_k$, see Mayne [2000], and using plant model parameter estimate $\hat{\boldsymbol{\theta}}$.

$$\min_{\bar{\boldsymbol{u}}_k} J(k, \bar{\boldsymbol{u}}) \quad = F(\hat{\boldsymbol{x}}_{k+N_p}) + \sum_{j=0}^{N_p-1} g(\hat{\boldsymbol{x}}_{k+j}, \boldsymbol{u}_{k+j}^k) \tag{2.16a}$$

$$s.t. \qquad \hat{\boldsymbol{x}}_{k+j+1} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k+j}, \boldsymbol{u}_{k+j}^k, \hat{\theta}), \quad \text{from } \boldsymbol{x}_k, \quad j = 1, \ldots, N_p - 1, \tag{2.16b}$$

$$\boldsymbol{u}_{k+j}^k \in \mathcal{U}, \qquad j = 0, \ldots, N_p - 1, \tag{2.16c}$$

$$\hat{\boldsymbol{x}}_{k+j} \in \mathcal{X}, \qquad j = 0, \ldots, N_p, \tag{2.16d}$$

where the following conditions are assumed to hold:

- the dynamics $\boldsymbol{f}(\cdot)$, the terminal cost $F(\cdot)$, and the stage cost $g(\cdot)$ are continuous;
- $\boldsymbol{f}(0,0) = 0$, $F(0) = 0$, $g(0,0) = 0$;
- the process noise in (2.16b) is replaced by its (zero) mean, and the parameter vector $\boldsymbol{\theta}$ is considered constant along the horizon N_p;
- $\mathcal{U} \subseteq \mathcal{R}^m$ contains the origin in its interior;
- $\mathcal{X} \subseteq \mathcal{R}^n$ contains the origin in its interior.

2.3.3 Convex and non-convex Model Predictive Control

The concept of convexity is very important in optimization. Problems that are convex are generally easier to solve. Convex optimization is a very well defined and studied area [Boyd & Vandenberghe, 2004]. Convex problems have the characteristic that if a local solution exists, it is also a global solution. Standard Quadratic Programming optimization problems (QPs) are a subset of the class of convex optimization problems. QPs have convex objective functions, linear equality constraints and linear inequality constraints.

MPC with a linear model, a convex cost function, and a convex set of constraints, results in a convex optimization problem. As seen previously, formulations (2.6) and (2.13) can be easily converted to standard QP problems. For solving these problems, efficient and reliable algorithms are available [Nocedal & Wright, 2000].

Nonlinear models in MPC lead to non-convex optimization problems. A method for solving non-convex optimization problems is Sequential Quadratic Programming (SQP) [Nocedal & Wright, 2000]. SQP is one of the most popular methods for solving nonlinearly constrained optimization problems. Fundamentally, an SQP algorithm decomposes the non-convex problem into a series of QPs, and sequentially solves them. The main drawback of SQP algorithms is their computational complexity, especially if compared with QP algorithms. Furthermore, SQP will typically find a locally optimal solution, but this local solution is not guaranteed to also be the globally optimal solution.

The next chapter describes an example of MPC, where a novel state dependent weight is defined. Despite the fact that the model is nonlinear, several approaches using linearized approximated models and a standard QP solver are used, instead of SQP solvers.

2.3.4 Explicit Model Predictive Control

When an MPC is implemented, it is well known that at every sampling time an optimization problem must be solved online. Although faster algorithms are developed and computational power availability increases constantly, for some systems with fast dynamics it is not possible to apply the standard MPC. Explicit MPC has been developed as a solution to this problem. The basic idea is to transfer the computation offline. This is done by computing a closed-form solution to the optimization problem, leading to a reduction in the complexity of the online algorithm.

An algorithm to explicitly determine the linear quadratic state feedback control law for constrained linear systems is developed in Bemporad et al. [2002]. It is shown that the control law obtained is piece-wise linear and continuous, and the only online computation consists of a simple evaluation of an explicitly defined piece-wise linear function. Another interesting algorithm is developed in Tøndel et al. [2003], where the results from Bemporad et al. [2002] are extended avoiding unnecessary partitioning, and improving the efficiency in the online evaluation.

One should note, however, that Explicit MPC algorithms are applicable in practice only to systems with few states (not much higher than five). This is caused mainly by the fast increase of the table dimensions, required to define the piece-wise linear functions. The offline computation for obtaining the explicit solution, and the organization for an efficient online evaluation, becomes prohibitive for systems with a large number of states. Present research is focused on sub-optimal Explicit MPC in order to find a simpler solution that can be applicable to larger systems. Extensions to nonlinear systems are investigated as well.

For the reader who wishes to find a more detailed introduction of this approach, a starting point can be Kvasnica [2009].

Chapter 3

Model Predictive Control with state dependent input weight, a simulation-based analysis

This chapter presents a simulation-based study that investigates and compares the effects of different cost function formulations, as well as several model simplifications using linearization, for application of linear(ized) MPC to a nonlinear system. An Autonomous Underwater Vehicle (AUV) is used for the study. The main contribution of this chapter is to introduce and illustrate the effects of a novel state dependent input weight, which is shown to improve performance compared to other simple MPC formulation modifications.

A nonlinear two-dimensional model is described, and it is used as the plant. Two different linearized models are implemented in the MPC, a Linear Time Invariant (LTI) and a Linear Time Varying (LTV), respectively. Controlling a nonlinear system by using linear models is common in MPC practice. MPC implementation for this particular AUV system is interesting, especially because it is possible to appreciate the significant control performance improvement when a nonlinear input weight for the MPC cost function is coupled with the LTV model.

For this application, the state dependent input weight idea in the cost function is a novelty, and resembles the use of a varying controller gain adopted in gain scheduling techniques [Rugh & Shamma, 2000]. In fact, due to the different input weights the controller gain, obtained at every time step after solving a QP optimization problem, varies as a function of AUV position. The state dependent input weight improves the solution robustness of QP problem.

The LTI model yields a standard linear MPC, which has the least computational demand. The LTV model yields an MPC which is still convex, but since the model state space matrices are computed every sampling time, and they are different for each time step

within the prediction horizon, the overall problem requires more computational effort. However, its solution requires less effort than an NMPC problem.

The underwater vehicle dynamics of the AUV are described in Sutton & Bitmead [1998]. It moves in a two-dimensional space, and it has a constant forward speed (surge). Thus, by controlling the vehicle rudder angle, that is physically constrained between $[-0.5, 0.5]$ radians, the goal is to follow the ocean floor at a certain constant distance. A rigorous mathematical formulation is given in next section.

3.1 The Autonomous Underwater Vehicle example

To illustrate the benefits of the state dependent input weight described in detail in Section 3.2.2 a simulation-based analysis of different approaches for controlling an Autonomous Underwater Vehicle (AUV) is presented.

The particular AUV model chosen has some features that make it of relevant interest. That is, due to the limited range of achievable control values the vehicle is unable to accelerate in an arbitrary direction. The response of the AUV to positive step changes in the rudder turning is initially to accelerate in the opposite direction, this behavior is reminiscent of non-minimum phase systems. These observations would suggest not attempting the use of feedback control which relies on high-gain or on system inversion. Finally, this class of systems may lead to difficulties in addressing issues on robustness and tuning of the controller. For more details see Santos & Bitmead [1995].

Note that the choice of taking this AUV example is made in order to illustrate the results using a relatively simple model, which still has the capability to display complex nonlinear behavior. The purpose of this chapter is *not* to make a realistic contribution to the control of underwater vehicles, which would clearly require a model in a three-dimensional space. However, this example makes clear the contribution of the state dependent input weight, highlighting the benefits.

For more details on underwater vehicles see Fossen [1994], which gives a comprehensive and extensive discussion on the guidance and control of ocean vehicles.

3.1.1 A two-dimensional model

Figure 3.1 shows a sketch of the vehicle. To describe the AUV dynamics a model is defined with respect to two frames of reference. This gives the possibility to know the position and orientation of the vehicle while moving in its environment. The vehicle dynamics, in continuous time, is given in the body-fixed frame defined by $\boldsymbol{Cxz}$. The surge and heave speed are defined by u[1] and w, respectively. The pitch angle rate is

[1]To have a standard description of the underwater vehicle model, an abuse of notation is required. In this case, u denotes the state variable surge, and not the system input.

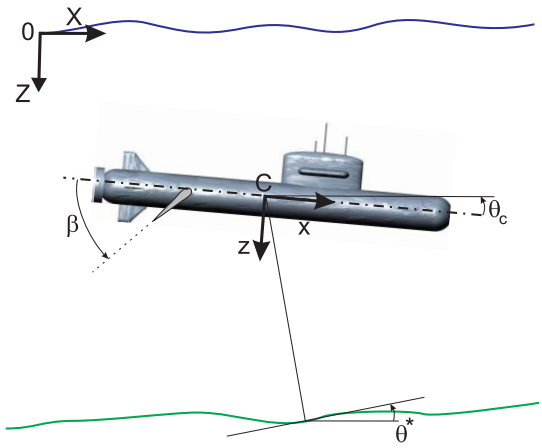


Figure 3.1: Underwater vehicle reference frames.

denoted q. Thus (u, w, q) are respectively, the forward, perpendicular and anti-clockwise rotational velocities of the submarine along, to and around its major axis in the body-fixed frame.

In the earth-fixed reference frame $\boldsymbol{OXZ}$ we describe the motion of the vehicle as shown in (3.2). The attitude of the vehicle is defined by θ_c. The vehicle dynamics is described, according to Sutton & Bitmead [1998], as

$$\boldsymbol{M_I}\begin{bmatrix}\dot{u}(t)\\ \dot{w}(t)\\ \dot{q}(t)\end{bmatrix} = mq(t)\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}u(t)\\ w(t)\\ q(t)\end{bmatrix} + u(t)\boldsymbol{D_h}\begin{bmatrix}u(t)\\ w(t)\\ q(t)\end{bmatrix} + \boldsymbol{\gamma_g}(t) + \boldsymbol{u_{cw}}(t) + \boldsymbol{u_{cp}}(t) \tag{3.1}$$

$$\begin{bmatrix}\dot{x}_c(t)\\ \dot{z}_c(t)\\ \dot{\theta}_c(t)\end{bmatrix} = \begin{bmatrix}\cos(\theta_c(t)) & \sin(\theta_c(t)) & 0\\ -\sin(\theta_c(t)) & \cos(\theta_c(t)) & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}u(t)\\ w(t)\\ q(t)\end{bmatrix} \tag{3.2}$$

where $\boldsymbol{M_I}$ is the inertia matrix including the hydrodynamic added mass, m is the vehicle mass, $\boldsymbol{D_h}$ is the damping matrix, and the buoyancy term $\boldsymbol{\gamma_g}(t)$ is zero because the vehicle is assumed to be neutrally buoyant. $\boldsymbol{u_{cw}}(t)$ and $\boldsymbol{u_{cp}}(t)$ are respectively the forces and the moments generated by the rudder and propeller. Their expressions are given by (3.3-3.12).

$$v^2 \triangleq u^2 + w^2, \tag{3.3}$$

$$\epsilon \triangleq \sin^{-1}\left(\frac{w}{v}\right), \tag{3.4}$$

$$J_p \triangleq \frac{u}{|\nu D_p|}, \tag{3.5}$$

where ϵ is the angle between the $\boldsymbol{Cx}$ axis and the velocity vector $\boldsymbol{v}$, which is not necessarily the same as θ_c. J_p is the propeller advancement coefficient, ν is the propeller shaft rotational speed, and D_p is the propeller diameter.

$$C_{xsw} \triangleq -C_{xow} - c_w \frac{C_{zow}^2(\epsilon+\beta)^2}{2\pi b_w}, \tag{3.6}$$

$$C_{zsw} \triangleq -C_{zow}(\epsilon+\beta) - 2.1(\epsilon+\beta)^2, \tag{3.7}$$

$$\boldsymbol{u}_{cw} = [u_{cw_{11}}, u_{cw_{21}}, u_{cw_{31}}]^T, \tag{3.8}$$

$$u_{cw_{11}} = 0.5\rho S_w v^2 (C_{zsw}\sin(\beta) + C_{xsw}\cos(\beta)), \tag{3.9}$$

$$u_{cw_{21}} = 0.5\rho S_w v^2 (C_{zsw}\cos(\beta) - C_{xsw}\sin(\beta)), \tag{3.10}$$

$$u_{cw_{31}} = -u_{cw_{21}}(0.2c_w\cos(\beta) + d_{aw}) - u_{cw_{11}}(0.2c_w\sin(\beta)), \tag{3.11}$$

$$\boldsymbol{u}_{cp} = \begin{bmatrix} \rho|\nu|\nu D_p^4(C_{t0p} + C_{t1p}J_p + C_{t2p}J_p^2 + C_{t3p}J_p^3) \\ -\rho|\nu|D_p^3 w C_n \\ \rho|\nu|D_p^3 w C_n D_{ap} \end{bmatrix}, \tag{3.12}$$

where $S_w = b_w c_w$ is the rudder surface, and ρ is the sea water density. C_{xow}, C_{zow}, d_{aw}, b_w, c_w, are the rudder characteristics, and C_{t0p}, C_{t1p}, C_{t2p}, C_{t3p}, C_n, D_{ap}, are the propeller characteristics. More details can be found in Santos & Bitmead [1995].

The vehicle has two inputs that can be used for control purposes. They are the rudder deflection β, and the propeller rotation frequency ν, respectively. The ocean bottom position is modeled as a disturbance to be rejected by the controller. Its model is described by state x_f which is the rate of change of the absolute angle of the ocean bottom defined by $\theta^*(t)$ at $x_f^*(t)$. The angular velocity of the sea floor $\dot{\theta}^*(t)$ is modeled as the negative output of a first order filtered white noise process driven by $\xi(t)$

$$\dot{x}_f(t) = A_f x_f(t) + B_f \xi(t) \tag{3.13a}$$

$$f(t) = C_f x_f(t) \tag{3.13b}$$

The relative angle θ between the ocean bottom and the vehicle is given by

$$\theta(t) = \theta_c(t) - \theta^*(t), \tag{3.14}$$

which gives

$$\begin{aligned}\dot{\theta}(t) &= \dot{\theta}_c(t) - \dot{\theta}^*(t) \\ &= q(t) + C_f x_f(t)\end{aligned} \tag{3.15}$$

where $\dot{\theta}^*(t) = -f(t) = -C_f x_f(t)$.

The relative distance between the ocean bottom and the vehicle center is given by $\kappa(t)$ and its rate of change is computed with

$$\dot{\kappa}(t) = u(t)\sin(\theta(t)) - w(t)\cos(\theta(t)). \tag{3.16}$$

It is assumed that the distance between the vehicle center and the ocean bottom is measured, and the measurement is affected by a white Gaussian noise η

$$y(t) = \kappa(t) + \eta(t). \tag{3.17}$$

For representing the system (3.1-3.17) in a more compact form, let us define the state vector

$$\boldsymbol{x}(t) = \begin{bmatrix} u \\ w \\ q \\ \theta \\ \kappa \\ x_f \end{bmatrix}, \tag{3.18}$$

thus (3.1-3.17) becomes

$$\dot{\boldsymbol{x}}(t) = f\left(\boldsymbol{x}(t), \beta(t), \xi(t)\right) \tag{3.19a}$$

$$y(t) = \boldsymbol{C}\boldsymbol{x}(t) + \eta(t) \tag{3.19b}$$

where the dynamics (3.19a) is nonlinear, and the measurement (3.19b) is linear, with

$$\boldsymbol{C} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}^T. \tag{3.20}$$

The vehicle parameters can be found in Santos & Bitmead [1995]. The system (3.19) has non-minimum phase behavior, in the sense that a positive input step change on β makes the vehicle accelerate in the opposite direction before converging eventually to the steady state value of the response. Furthermore, an inability to accelerate in an arbitrary direction, due to the limited range of the control value reachable $\boldsymbol{u_{cw}}$, makes the system nonholonomic. These considerations make the control of the vehicle interesting and suitable for investigating model predictive control features.

3.2 Model-Based Predictive Control

The standard linear MPC framework (2.6) is implemented. However, since the AUV model is nonlinear, (3.19) is linearized. Two different strategies are applied, using first an LTI internal model, and then an LTV model.

3.2.1 Internal models used for prediction

The internal linear model is obtained by linearizing the underwater vehicle dynamics (3.19). Thus, for a sufficiently small interval around the equilibrium point, it is possible to suitably approximate the nonlinear behavior of the AUV by the state space system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\beta(t) \tag{3.21a}$$
$$y(t) = \boldsymbol{C}\boldsymbol{x}(t) \tag{3.21b}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are the following Jacobians

$$\boldsymbol{A} = \left.\frac{\partial f(\boldsymbol{x},\beta)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*,\beta=\beta^*} \tag{3.22}$$
$$\boldsymbol{B} = \left.\frac{\partial f(\boldsymbol{x},\beta)}{\partial \beta}\right|_{\boldsymbol{x}=\boldsymbol{x}^*,\beta=\beta^*} \tag{3.23}$$

calculated about some point $(\boldsymbol{x}^*, \beta^*)$ in the combined state and input space. The measurement matrix $\boldsymbol{C}$ is given in (3.20).

In MPC using linearized models, is it common practice to linearize around an equilibrium point, and to 'move the origin to the equilibrium point' by the introduction of deviation variables. Here, the AUV is moving forward along some trajectory, and the deviation variables in (3.21) are therefore relative to the nominal trajectory.

The MPC linear model thus obtained is sufficiently accurate when AUV is close to the point where the linearization is obtained. However, when increasing the initial offset between the working point and the initial condition the LTI approach fails. Therefore, as a candidate strategy for solving this issue, an LTV model is implemented as follow.

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_t\boldsymbol{x}(t) + \boldsymbol{B}_t\beta(t) \tag{3.24a}$$
$$y(t) = \boldsymbol{C}_t\boldsymbol{x}(t) \tag{3.24b}$$

where $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ are computed by

$$\boldsymbol{A}_t = \left.\frac{\partial f(\boldsymbol{x},\beta)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^*(t),\beta=\beta^*(t)} \tag{3.25}$$
$$\boldsymbol{B}_t = \left.\frac{\partial f(\boldsymbol{x},\beta)}{\partial \beta}\right|_{\boldsymbol{x}=\boldsymbol{x}^*(t),\beta=\beta^*(t)} \tag{3.26}$$

and $\boldsymbol{C}_t$ is (3.20). Note that $\beta^*(t)$ and $\boldsymbol{x}^*(t)$ define the trajectory where the Jacobians are computed, and are generated by using the MPC optimal solution at the previous step, as the algorithm in Section 3.2.3 describes.

Since MPC requires a discrete time framework, (3.21) is discretized by using the standard Euler approximation

$$\dot{\boldsymbol{x}}(t) \simeq \frac{\boldsymbol{x}_{k+1} - \boldsymbol{x}_k}{\delta} \tag{3.27}$$

where δ is the sampling time.

Analogously, for LTV model we have that

$$\boldsymbol{A}_{k+i} = \boldsymbol{I} + \delta \boldsymbol{A}_t, \qquad \boldsymbol{B}_{k+i} = \delta \boldsymbol{B}_t, \tag{3.28}$$

where k is the current discrete time step, and $i = \{0, 1, \ldots, N_p - 1\}$ indicates the step interval where the discretized Jacobian is applied, within the MPC prediction horizon.

Although the LTV approach allows the AUV to meet the control goal, the transient response of the system is not satisfactory, as discussed more in detail in the simulations in Section 3.3. Thus a nonlinear state dependent cost function is defined as shown next.

3.2.2 Nonlinear state dependent cost function weight

In Sutton & Bitmead [1998] the following cost function is defined

$$J = \sum_{i=0}^{N_p-1} y_i^2 + R_c \beta_i^2 + Q_c \theta_i^2 \tag{3.29}$$

where R_c is the input weight, and Q_c is a penalty on the angle θ in order to avoid the vehicle heading backward.

To use (3.29) as the MPC objective function, it is then necessary to reformulate it in a more standard form, resulting in

$$\begin{aligned} J(\boldsymbol{x}, \beta) = &(\boldsymbol{x}_{N_p} - \boldsymbol{x}_{ref})^T \boldsymbol{S} (\boldsymbol{x}_{N_p} - \boldsymbol{x}_{ref}) \\ &+ \sum_{i=0}^{N_p-1} (\boldsymbol{x}_i - \boldsymbol{x}_{ref})^T \boldsymbol{Q} (\boldsymbol{x}_i - \boldsymbol{x}_{ref}) + (\beta_i - \beta_{ref}) R (\beta_i - \beta_{ref}) \end{aligned} \tag{3.30}$$

where

$$\boldsymbol{S} = \boldsymbol{Q} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q_c & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad R = R_c, \tag{3.31}$$

and

$$\boldsymbol{x}_i = \begin{bmatrix} u_i \\ w_i \\ q_i \\ \theta_i \\ \kappa_i \\ x_{f_i} \end{bmatrix}, \qquad \boldsymbol{x}_{ref} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \beta_{ref} = 0 \tag{3.32}$$

To improve control performance, a computationally effective solution is to define the nonlinear input weight

$$R_c = R_c(k) = a\kappa_k^2 + b \tag{3.33}$$

where κ_k is the relative depth, between the reference (that is a constant distance from the ocean floor) and the AUV center of gravity, at the time step k, a and b are positive constant design parameters. Thus, for every time step a new input weight is calculated.

From (3.19b) it is possible to note that κ is the only state directly measured. As general practice, when implementing MPC, a full state knowledge is needed. Thus, an Extended Kalman Filter (EKF) is implemented for estimating the AUV state. In particular, for calculating the nonlinear weight the estimate $\hat{\kappa}$ is used instead. With this framework, the measurement noise is also filtered by the EKF. Thus, the input weight is computed by

$$R_c(k) = a\hat{\kappa}_k^2 + b. \tag{3.34}$$

Finally, the use of (3.34) improves optimization solution robustness as discussed in the end of this chapter.

3.2.3 Algorithms

The discrete MPC algorithm is applied to the underwater vehicle system (3.19). For the linear MPC formulation with the LTI model, the MPC described in Chapter 2 is applied. In this section, we go through the LTV-MPC algorithm details. Thus, the tail of the optimal solution at the previous time step $k-1$ is defined as

$$\boldsymbol{\beta}^*_{k-1,N_p} = [\beta^*_{k|k-1}, \beta^*_{k+1|k-1}, \ldots, \beta^*_{k+N_p-1|k-1}, 0]^T, \tag{3.35}$$

which is used for computing the trajectory $\boldsymbol{x}^*_{k-1,N_p}$ about where the Jacobians (3.28) will be calculated.

LTV-MPC algorithm

Consider a generic sampling time instant k, then the LTV-MPC algorithm is defined by the following steps:

1. use β_{k-1}, y_{k-1} and EKF to estimate the state $\hat{\boldsymbol{x}}_{k-1}$;
2. use the nonlinear model (3.19) and the tail of the optimal solution at the previous step (3.35) to compute the trajectory

$$\boldsymbol{x}^*_{k-1,N_p} = [\boldsymbol{x}^*_{k|k-1}, \boldsymbol{x}^*_{k+1|k-1}, \ldots, \boldsymbol{x}^*_{k+N_p-1|k-1}, \boldsymbol{x}^*_{k+N_p|k-1}]^T; \tag{3.36}$$

3. compute the Jacobians (3.28), for $i = 0, 1, \ldots, N_p - 1$, about the predicted $\boldsymbol{x}^*_{k-1,N_p}$ and $\boldsymbol{\beta}^*_{k-1,N_p}$;
4. cast the MPC problem as a QP and then find its optimal solution;
5. from the optimal solution

$$\boldsymbol{\beta}^*_k = [\beta^*_{k|k}, \beta^*_{k+1|k}, \ldots, \beta^*_{k+N_p-1|k}]^T, \tag{3.37}$$

apply $\beta^*_{k|k}$ to the underwater vehicle, increment k and go to the algorithm step 1.

Moreover, when the state dependent weight is applied either to an LTI model approach, or to LTV-MPC, it is sufficient to include an extra step (between step 3 and step 4), which consist on calculating the state dependent nonlinear weight (3.34).

3.3 Simulation results

A series of MATLAB simulations are carried out to analyze and compare control performance of the different frameworks presented. The AUV is released some distance above the desired depth relative to the sea floor, and should quickly descend to the desired depth and then follow the sea floor at this depth. Note that in the following, the surge u is assumed constant and not estimated. This is a reasonable assumption since the task of bottom following will typically require a nominal constant value $u = u_0$. Furthermore, since this is primarily the concern of the rotational shaft speed ν control, the regulation of u can be separated from the regulation of the other variables, which focuses on the use of rudder angle β to permit bottom following.

The following corresponding MPC controllers are defined. In detail, their characteristics are:

(C1) - LTI internal model with constant input weight in the MPC cost function;
(C2) - LTI internal model with state dependent input weight in the MPC cost function;
(C3) - LTV internal model with constant input weight in the MPC cost function;
(C4) - LTV internal model with state dependent input weight in the MPC cost function.
(C5) - LTV-LTI internal model with constant input weight in the MPC cost function;
(C6) - LTV-LTI internal model with state dependent input weight in the MPC cost function.

Note that LTV-LTI is an MPC where the model used for prediction is linear time invariant within the prediction horizon, however it is re-linearized about the current state every time step, hence having a time-varying behavior.

All six controllers are re-formulated as QP problems, which are solved online, with the standard MATLAB QP solver 'quadprog'. Relevant simulation parameters are: sampling period $\delta = 0.2\ s$; input constraints $|\beta| \leqslant 0.5\ rad$; cost function state weight $Q_c = 10$, and input weight $R_c = 150$, when controllers (C1), (C3), and (C5) are used; when controllers (C2), (C4), and (C6) are implemented, the input weight is the one defined in (3.34), with $a = 1$ and $b = 150$, while the state weight is unchanged.

In all simulation instances, the same ocean bottom profile is considered. This is done by using the same noise sequence ξ in (3.13a). Finally, the common control goal is to follow the ocean bottom with a 10 m offset.

Simulation-based analysis

- In Figures 3.2, 3.4, 3.6, 3.8, 3.10, 3.12, 3.14, 3.16, 3.18, 3.20, 3.22, 3.24, 3.26, 3.28, 3.30, 3.32 $\theta_v = -\pi/2$ indicates the vertical trajectory, that would be the shortest path for the AUV to follow for reaching the desired depth.
- Figures 3.2 and 3.3 are obtained by applying the MPC approach (C1). Starting from an initial offset of 20 m, the controller is able to drive the AUV to the desired altitude (10 m distance from the ocean bottom) and to follow the seafloor profile. Note that in the first time steps the input constraint is hit (Figure 3.3(a)), moreover since the system has non-minimum phase, note how the vehicle starts going on the opposite direction for eventually converging on the desired path (Figure 3.3(b)). As visible in Figure 3.3(a), the EKF state estimation gives very satisfactory estimates.
- Figures 3.4 and 3.5 are obtained by applying the MPC approach (C2). These results are very similar to the previous analysis. Thus, when starting from a 20 m offset, the controller with the nonlinear input weight has the same performance as the one with the constant weight.
- Figures 3.6 and 3.7 are obtained by applying the MPC approach (C1). Starting from an initial offset of 50 m, the controller is not able to regulate the system. This is due to the poor approximation from the LTI model.
- Figures 3.8 and 3.9 are obtained by applying the MPC approach (C2). Starting from 50 m offset, the variable input weight is not a sufficient improvement and the controller fails as in the previous case.
  Thus to improve the prediction accuracy, the LTV model is used instead with the following results.
- Figures 3.10 and 3.11 are obtained by applying the MPC approach (C3). Starting from an initial offset of 20 m, the controller performance is very similar to the LTI model approach.

- Figures 3.12 and 3.13 are obtained by applying the MPC approach (C4). Again, from an initial offset of 20 m, the introduction of the state dependent input weight yield similar results to the LTI model approach with and without variable input weight, and also similar to the LTV approach.
- Figures 3.14 and 3.15 are obtained by applying the MPC approach (C3). This time, although starting from an initial offset of 50 m, the controller is able to make the AUV following the ocean bottom. However, as shown in Figure 3.15(a), in the transient part, the input presents large oscillations that could damage the actuator system. Thus, even if the control goal is reached, this solution it is not practicable.
- Figures 3.16 and 3.17 are obtained by applying the MPC approach (C4). Starting from 50 m offset, this time, due to the gain scheduling property of the variable input weight and to the better approximation of the LTV model, the AUV is able to follow the ocean bottom, without any potentially dangerous transient.
- Figures 3.18 and 3.19 are obtained by applying the MPC approach (C3). Starting from 100 m offset, and using only the LTV model with a constant input weight the control goal is met, but again with a not satisfactory input transient.
- Figures 3.20 and 3.21 are obtained by applying the MPC approach (C4). Also from 100 m offset, the combined approach is able to produce a very smooth control input following the ocean bottom.

Approaches (C5) and (C6) allow some interesting observations to be made. In fact, as shown in:

- Figures 3.22, 3.23, obtained by implementing the MPC (C5), starting from 20 m offset;
- Figures 3.24, 3.25, obtained by implementing the MPC (C6), starting from 20 m offset;
- Figures 3.26, 3.27, obtained by implementing the MPC (C5), starting from 50 m offset;
- Figures 3.28, 3.29, obtained by implementing the MPC (C6), starting from 50 m offset;
- Figures 3.30, 3.31, obtained by implementing the MPC (C5), starting from 100 m offset;
- Figures 3.32, 3.33, obtained by implementing the MPC (C6), starting from 100 m offset;

all controllers manage to follow the ocean bottom. This is due to better accuracy of time-varying models. However, it is possible to note the presence of input oscillations, that are larger when starting from a larger offset. These oscillations are due to the poor accuracy of the model within the prediction horizon.

Clearly, the best approach is to use the LTV model with the state dependent input weight. Here it is interesting to note how the variable weight in the cost function improves the Hessian condition number as explained in Section 3.2.2. Figures 3.34, and 3.35 show how switching from constant to state dependent input weight, reduces the Hessian condition number, yielding a well-conditioned optimization problem.

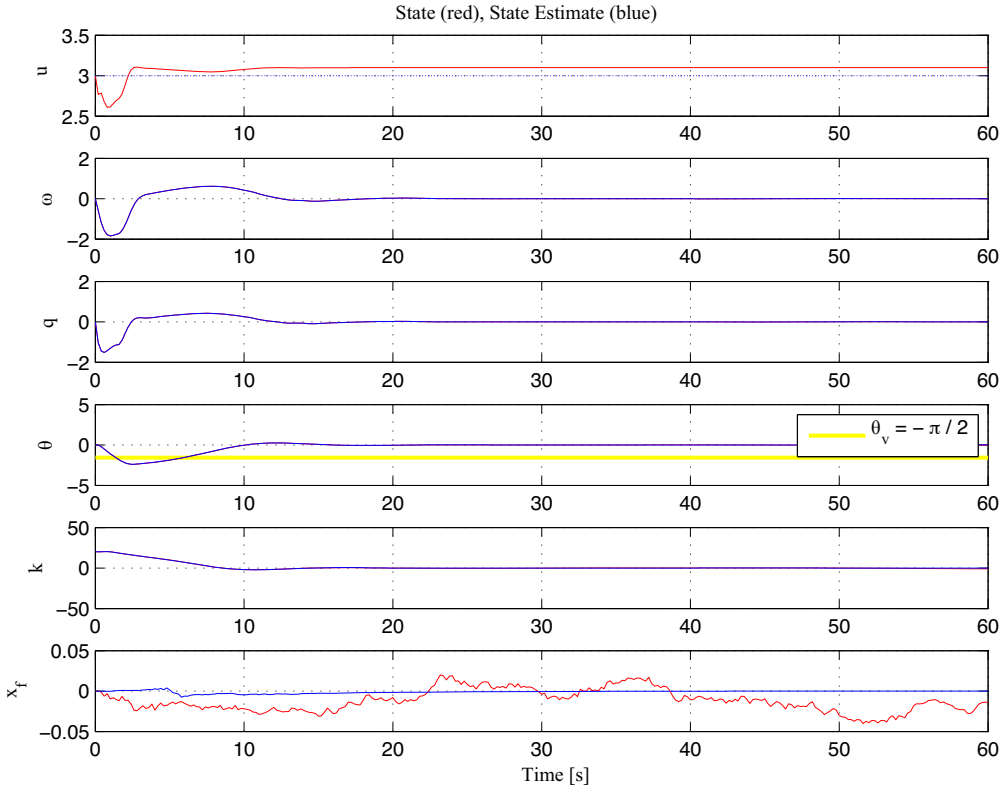
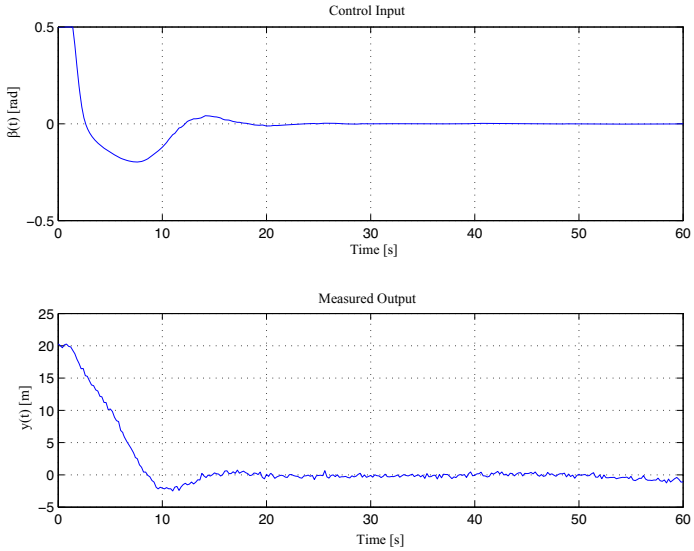
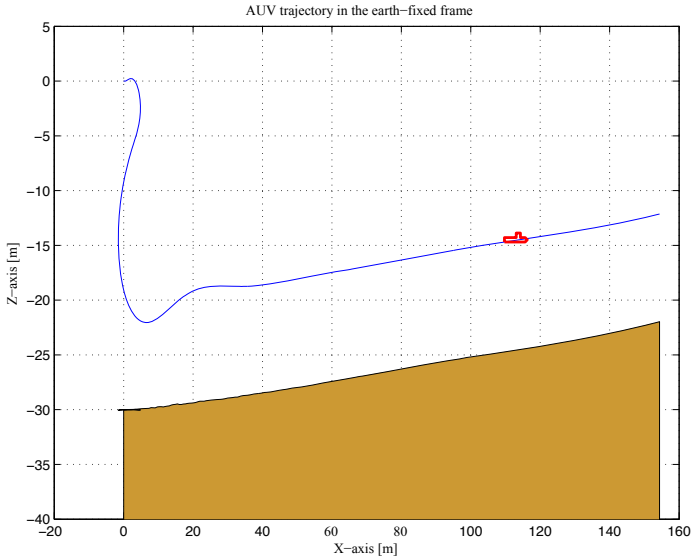


Figure 3.2: AUV real state, and EKF state estimate, using the LTI model and constant input weight. Initial condition of 20 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.3: Case with LTI model and constant input weight. Initial condition of 20 meters offset from the reference.

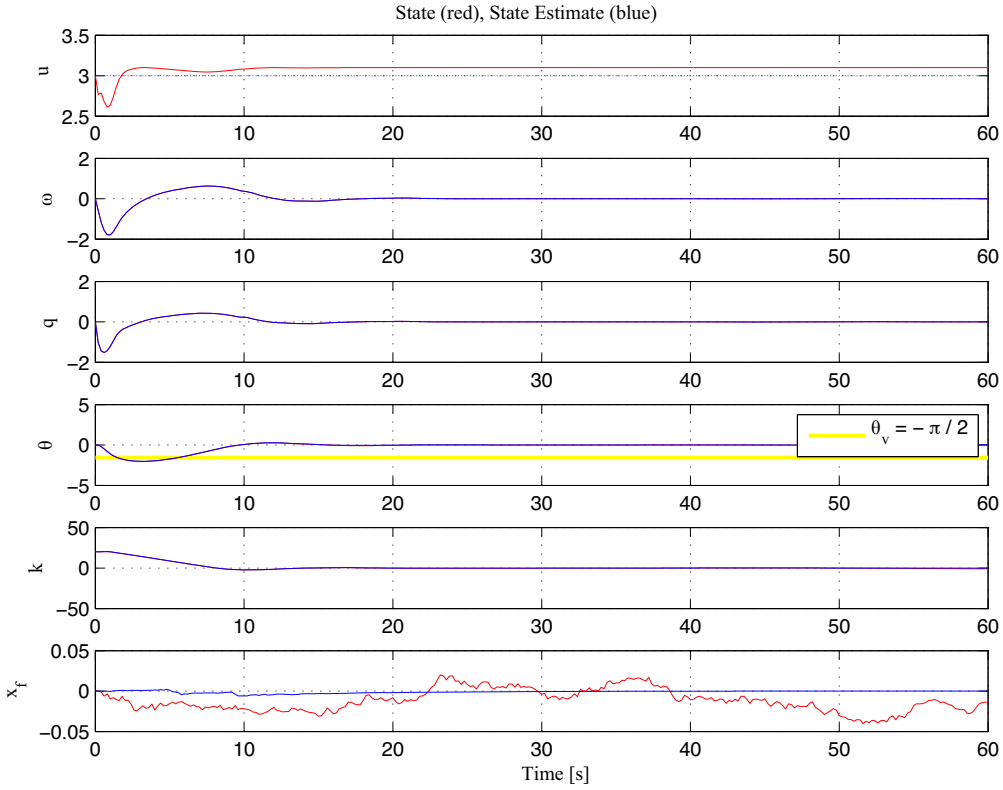
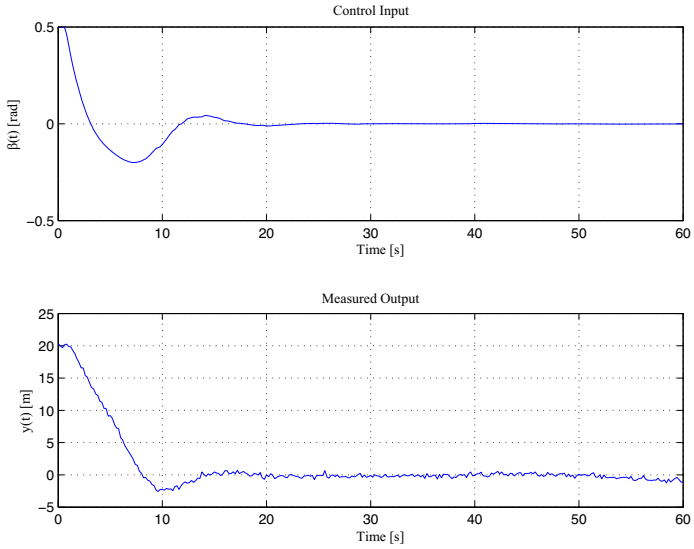
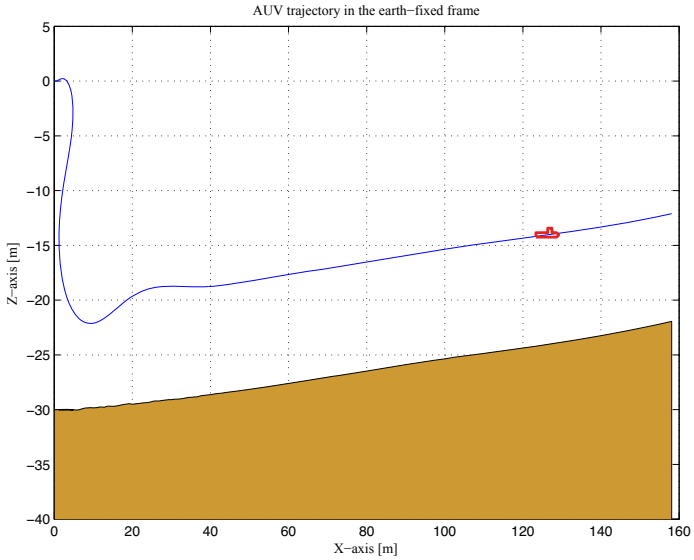


Figure 3.4: AUV real state, and EKF state estimate, using the LTI model and state dependent input weight. Initial condition of 20 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.5: Case with LTI model and state dependent input weight. Initial condition of 20 meters offset from the reference.

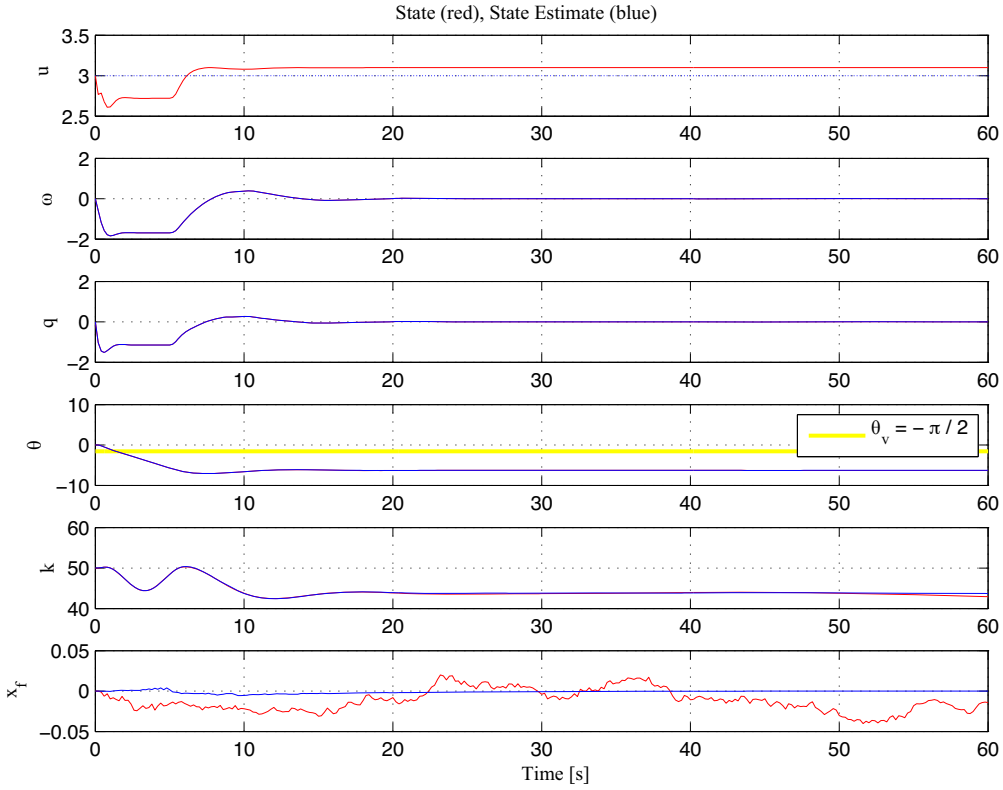
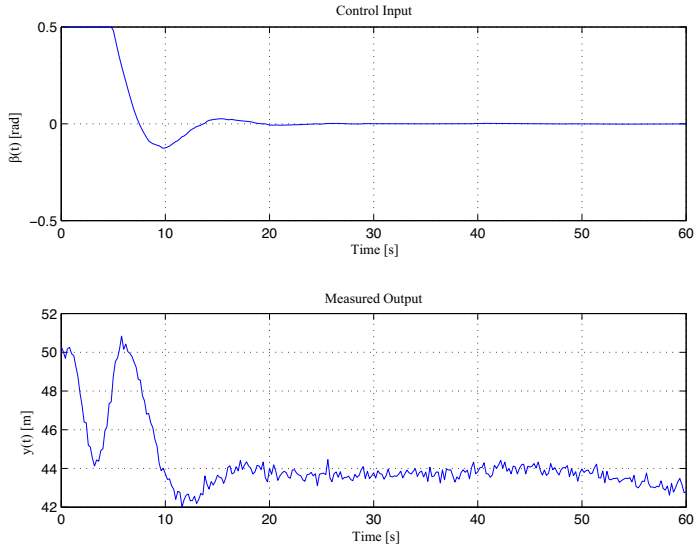
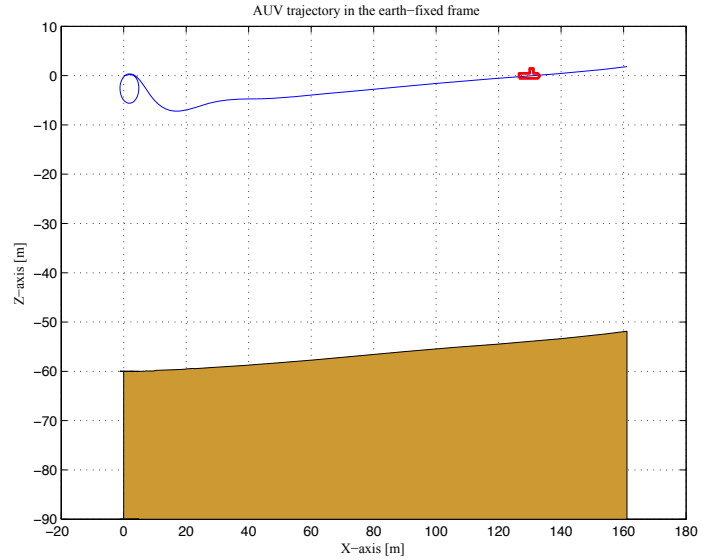


Figure 3.6: AUV real state, and its EKF state estimate, using the LTI model and constant input weight. Initial condition of 50 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.7: Case with LTI model and constant input weight. Initial condition of 50 meters offset from the reference.

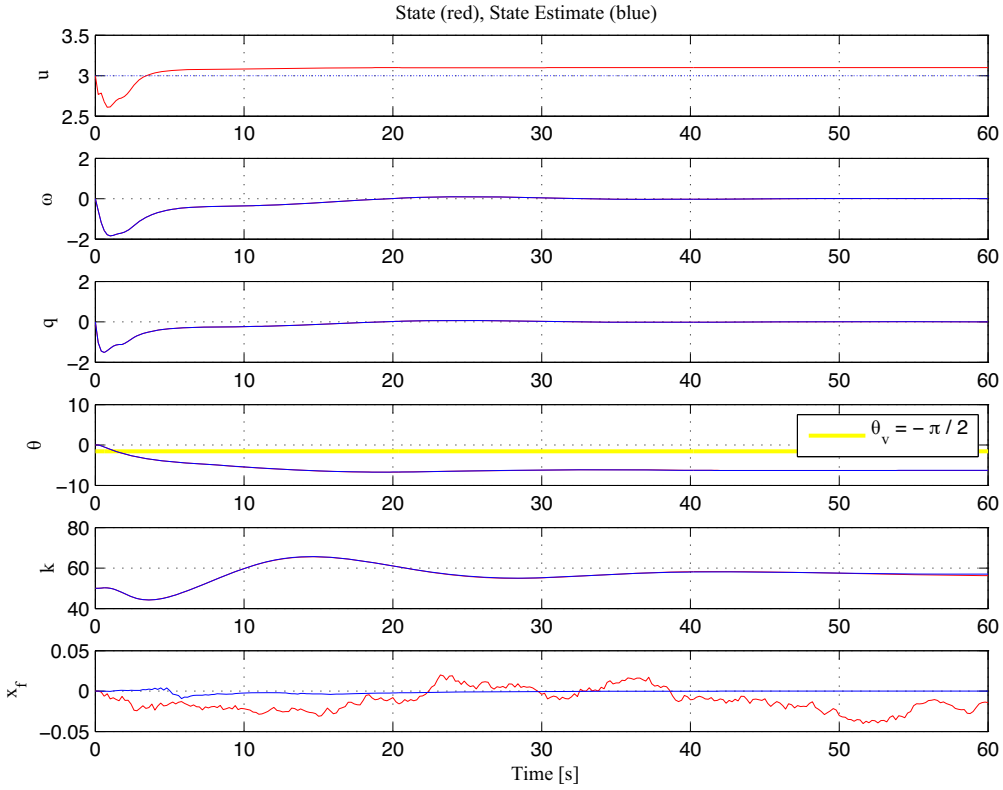
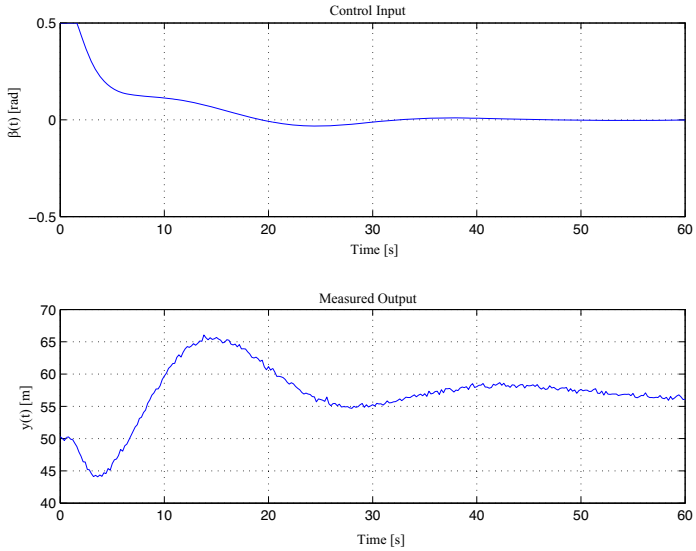
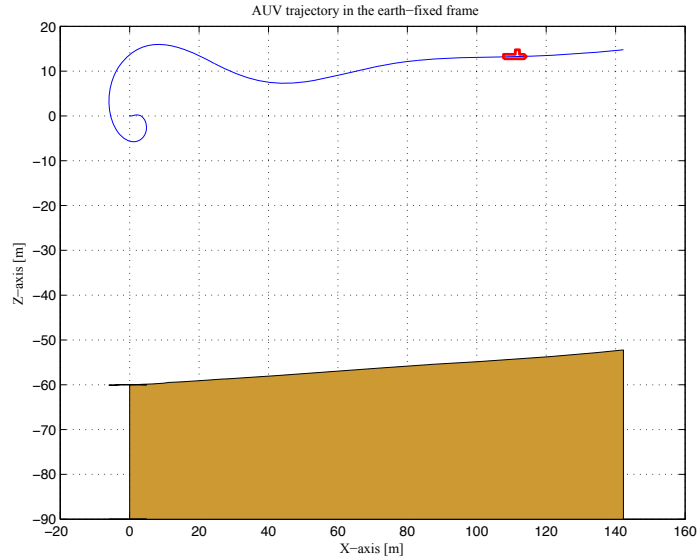


Figure 3.8: AUV real state, and EKF state estimate, using the LTI model and state dependent input weight. Initial condition of 50 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.9: Case with LTI model and state dependent input weight. Initial condition of 50 meters offset from the reference.

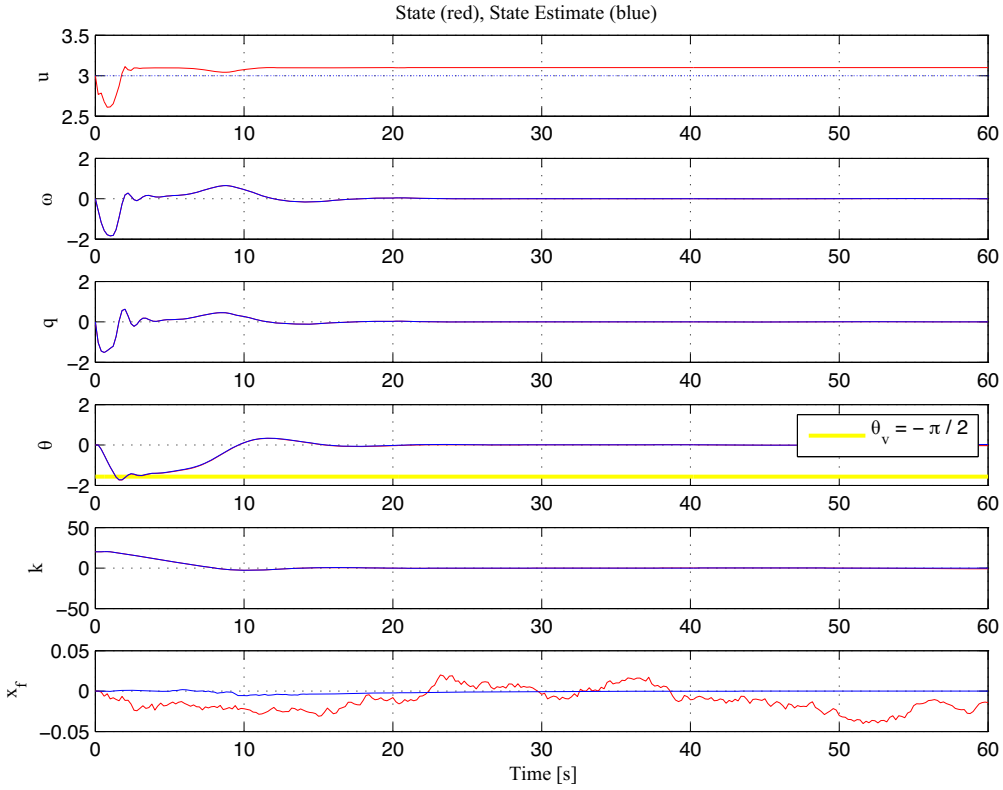
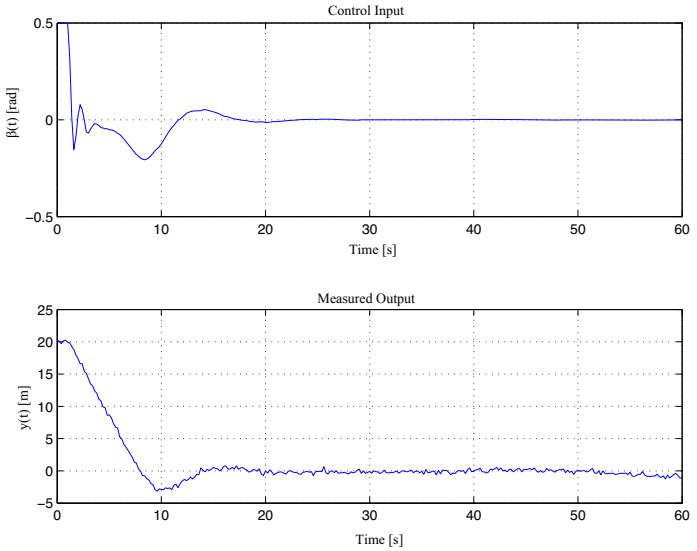
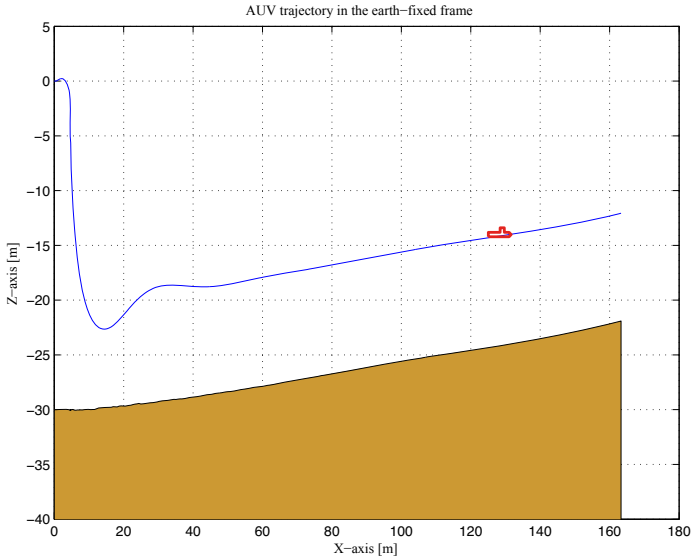


Figure 3.10: AUV real state, and EKF state estimate, using the LTV model and constant input weight. Initial condition of 20 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.11: Case with LTV model and constant input weight. Initial condition of 20 meters offset from the reference.

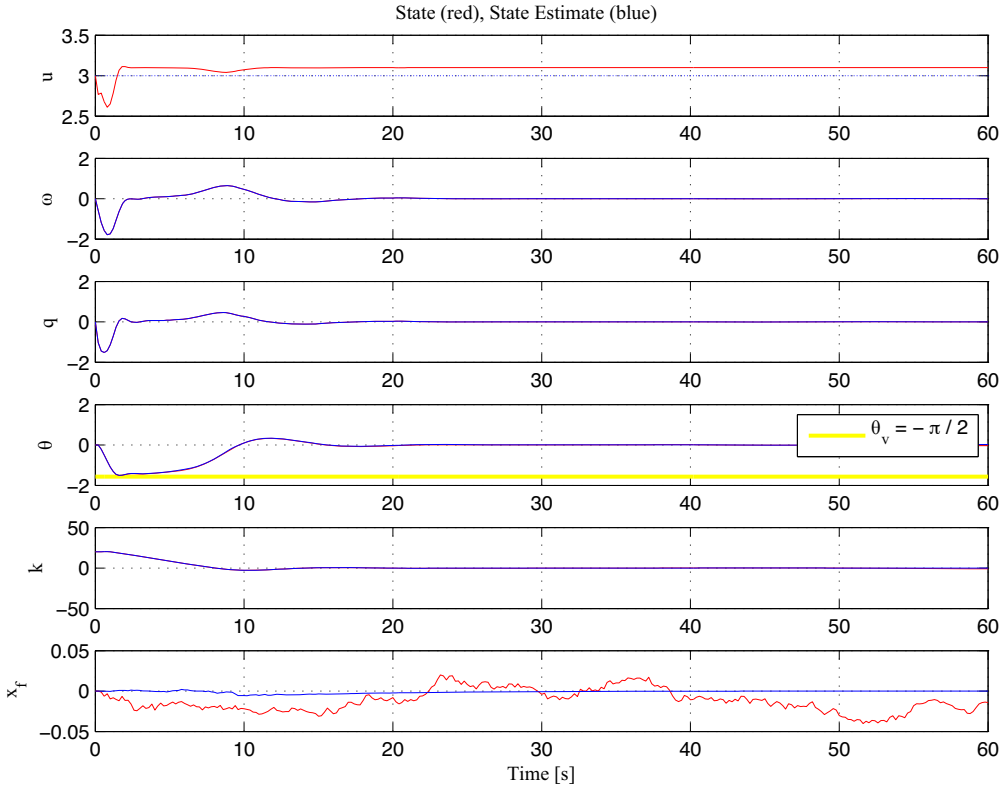
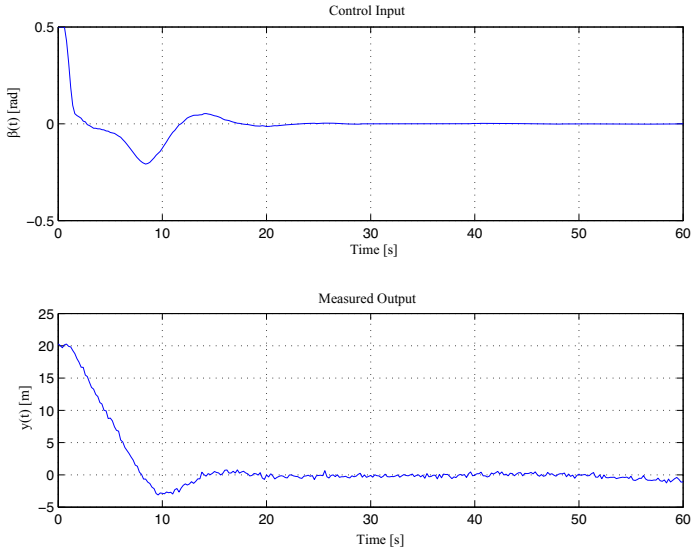
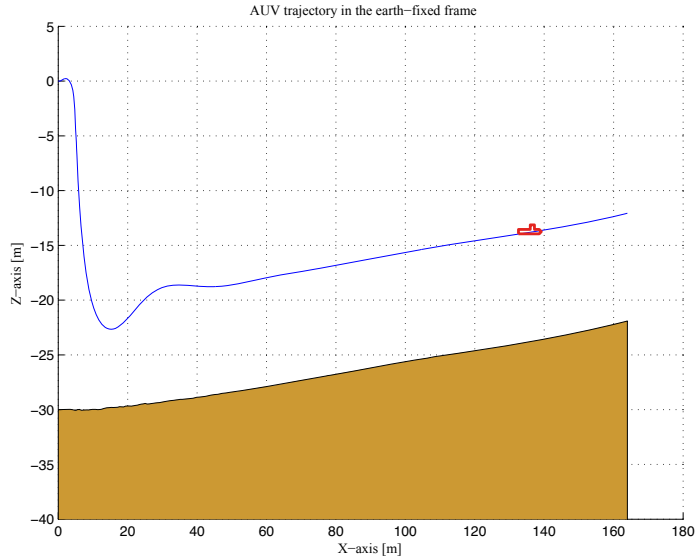


Figure 3.12: AUV real state, and EKF state estimate, using the LTV model and state dependent input weight. Initial condition of 20 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.13: Case with LTV model and state dependent input weight. Initial condition of 20 meters offset from the reference.

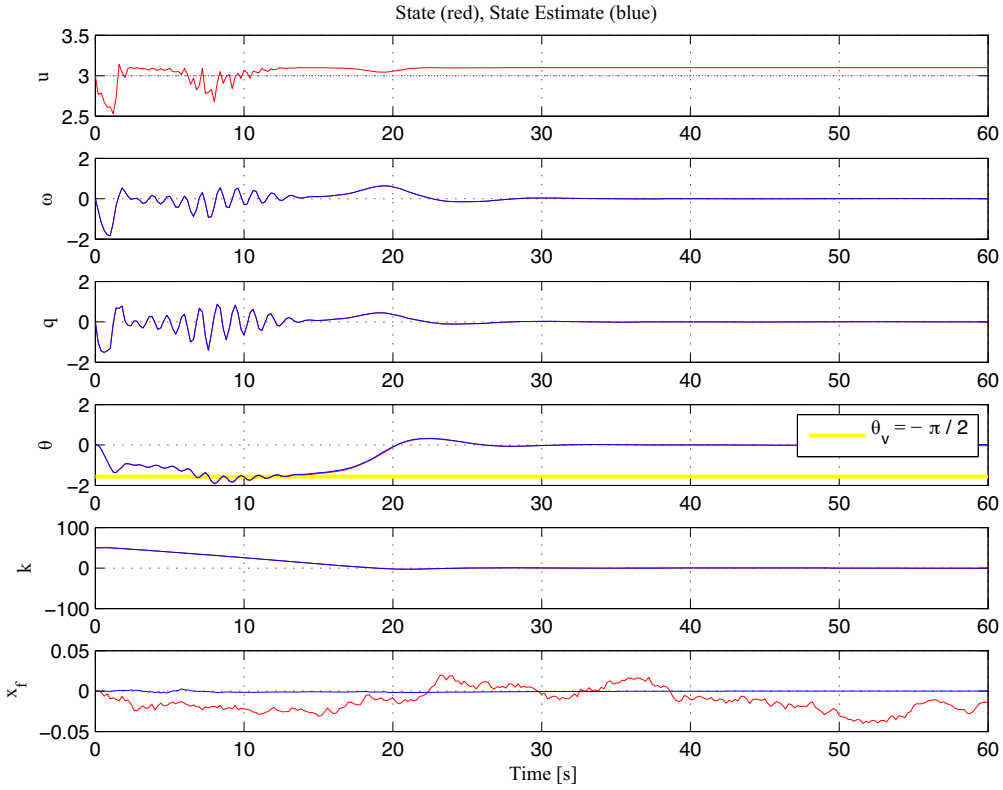
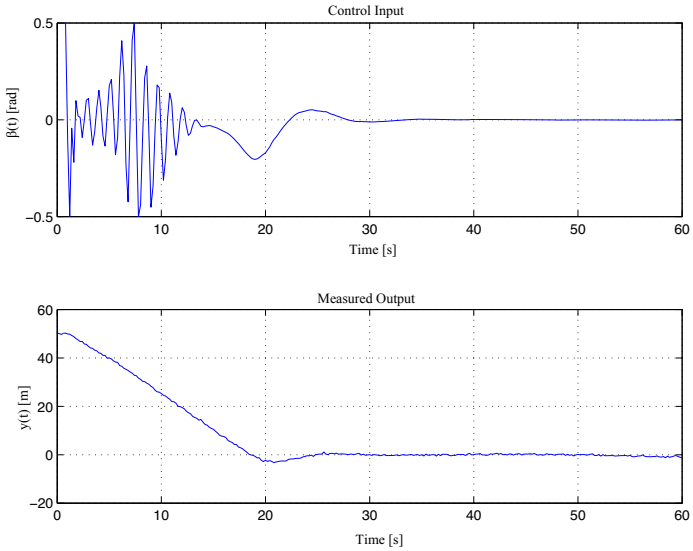
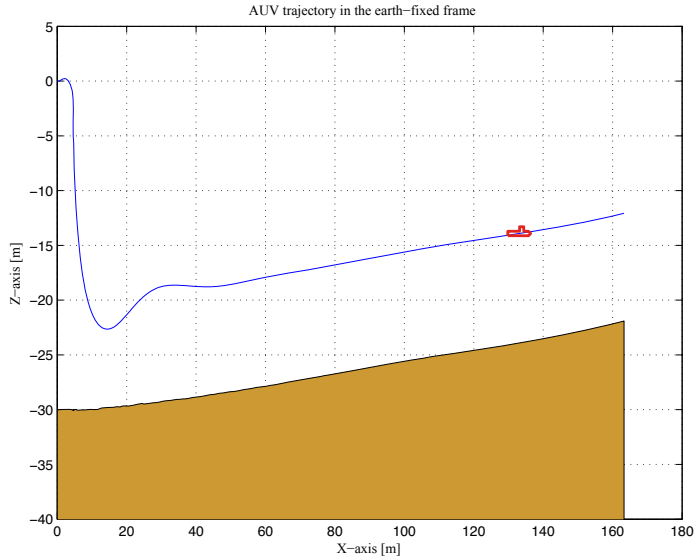


Figure 3.14: AUV real state, and its EKF state estimate, using the LTV model and constant input weight. Initial condition of 50 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.15: Case with LTV model and constant input weight. Initial condition of 50 meters offset from the reference.

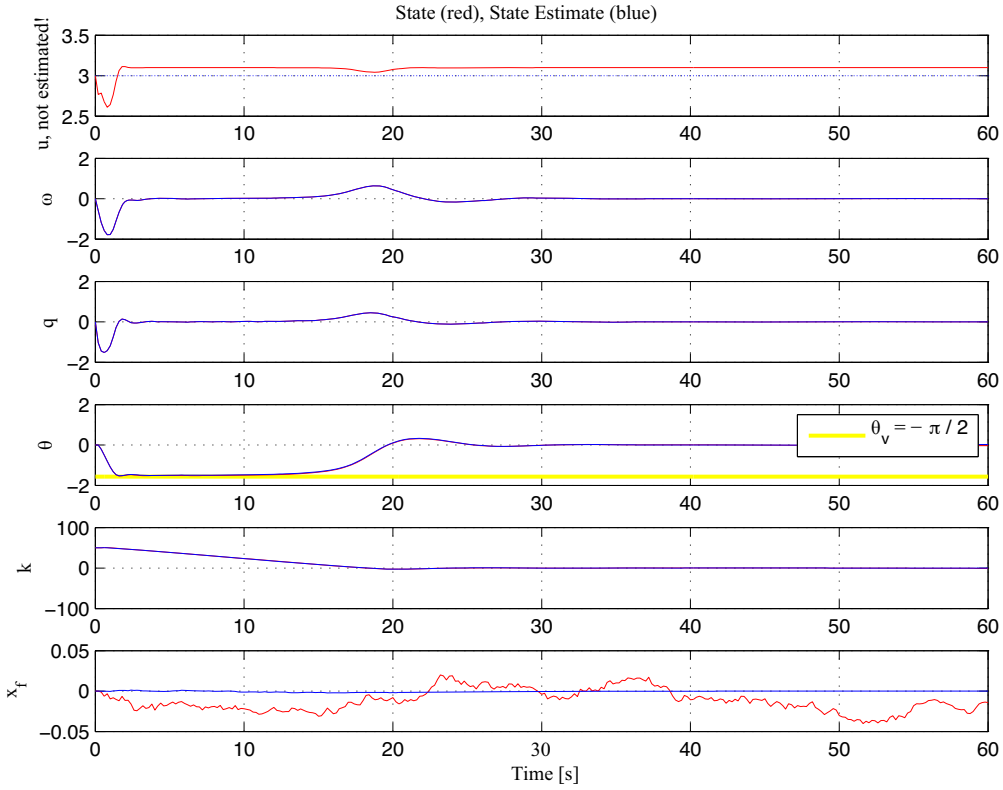
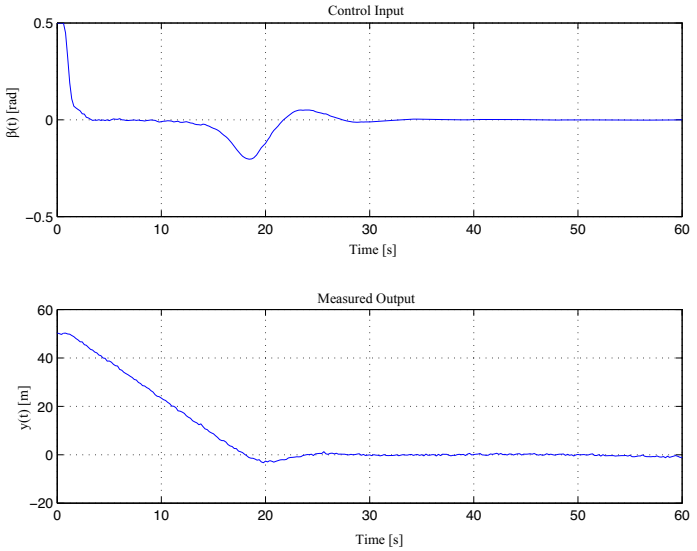
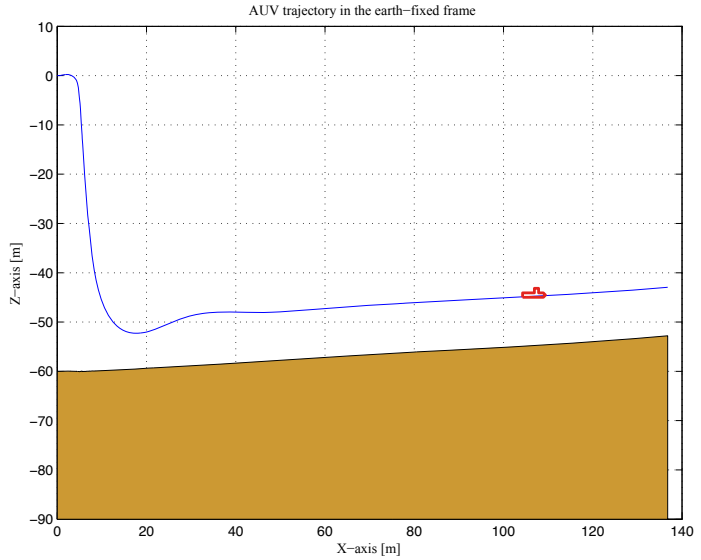


Figure 3.16: AUV real state, and EKF state estimate, using the LTV model and state dependent input weight. Initial condition of 50 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.17: Case with LTV model and state dependent input weight. Initial condition of 50 meters offset from the reference.

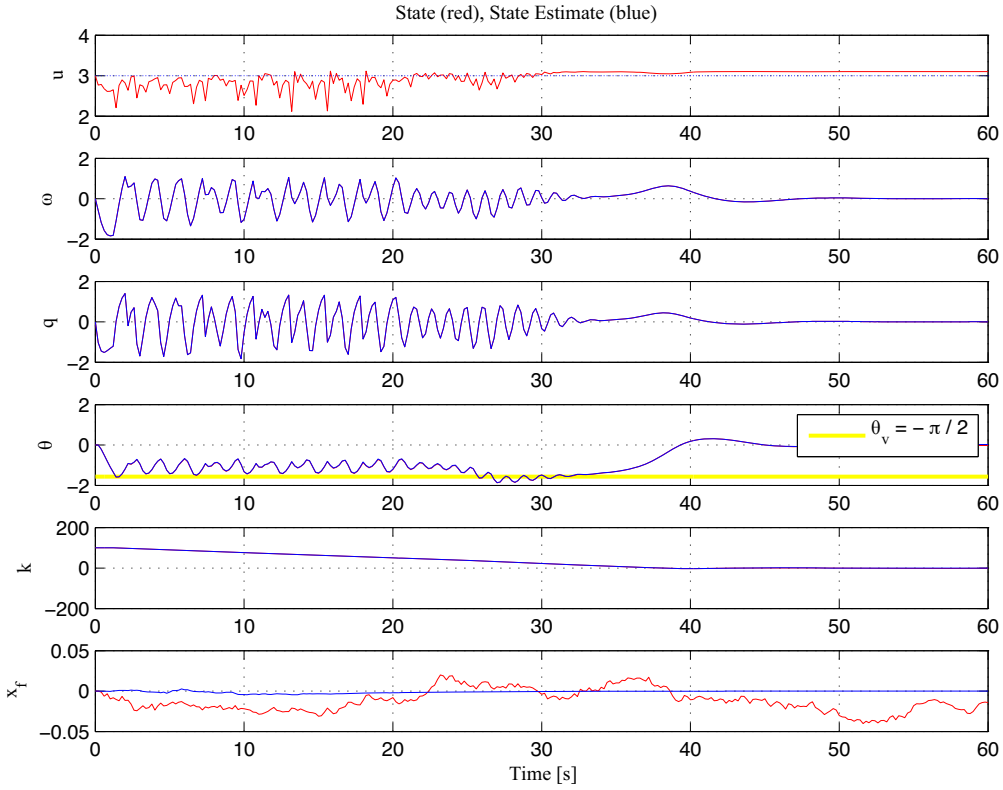
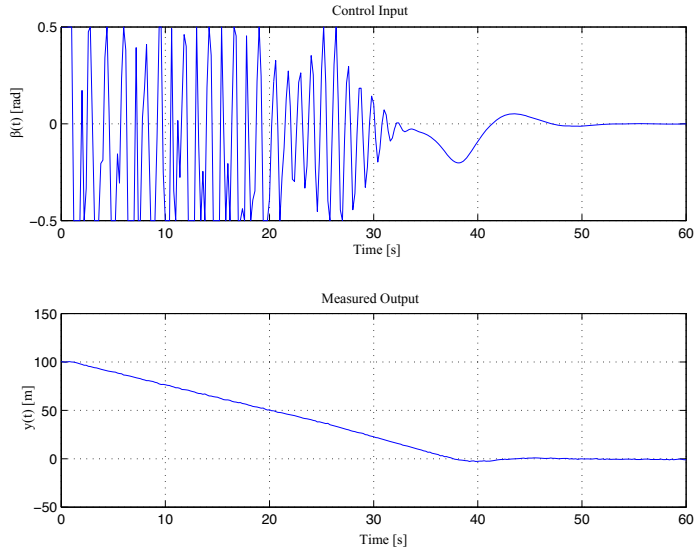
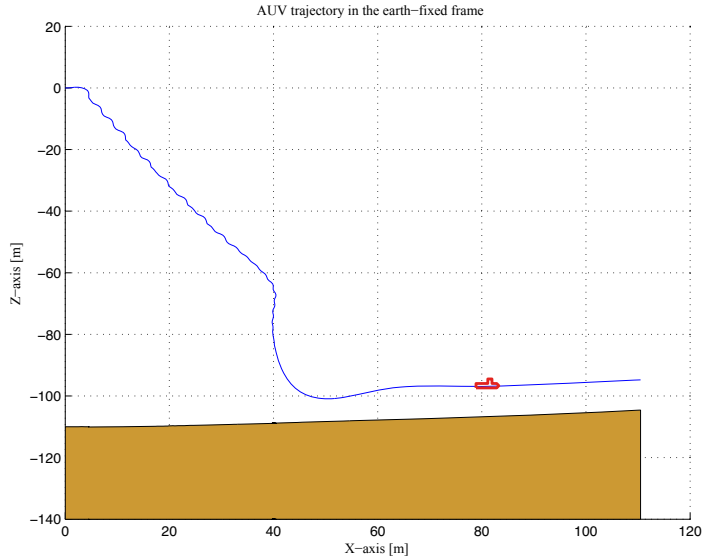


Figure 3.18: AUV real state, and its EKF state estimate, using the LTV model and constant input weight. Initial condition of 100 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.19: Case with LTV model and constant input weight. Initial condition of 100 meters offset from the reference.

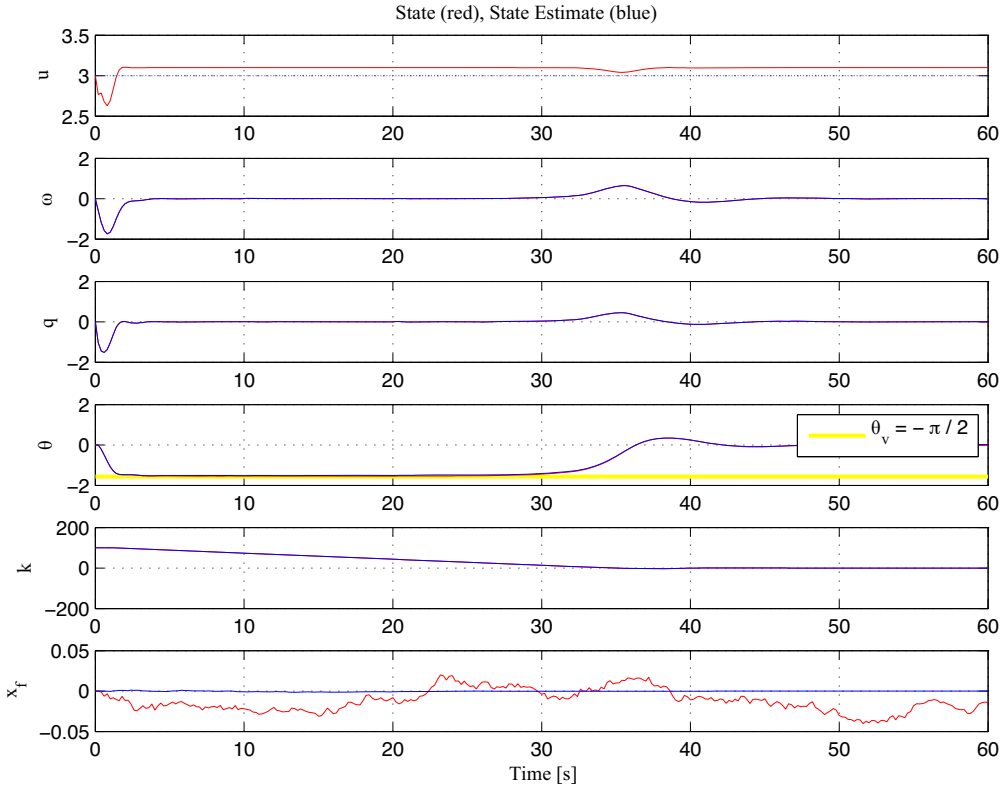
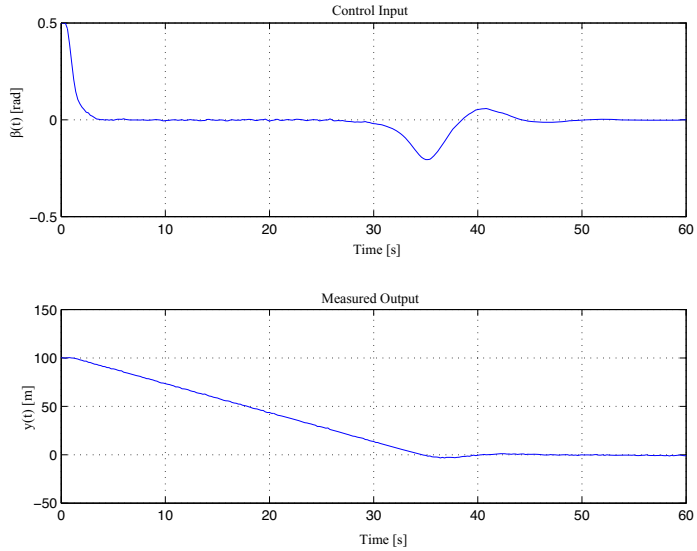
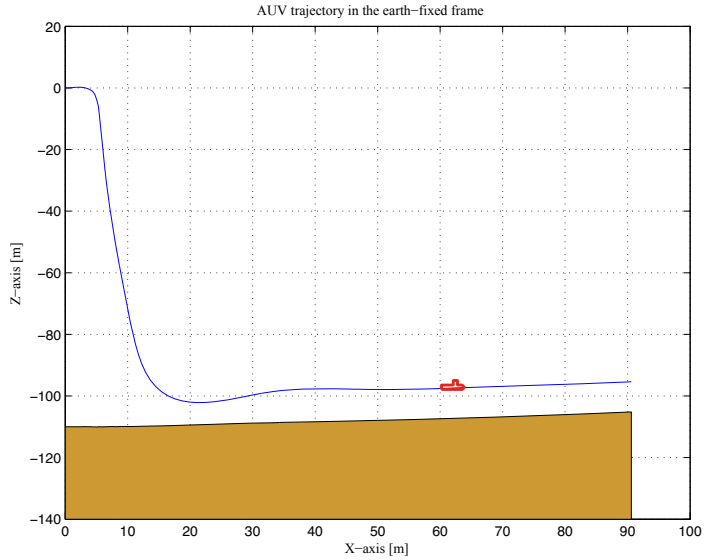


Figure 3.20: AUV real state, and EKF state estimate, using the LTV model and state dependent input weight. Initial condition of 100 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.21: Case with LTV model and state dependent input weight. Initial condition of 100 meters offset from the reference.

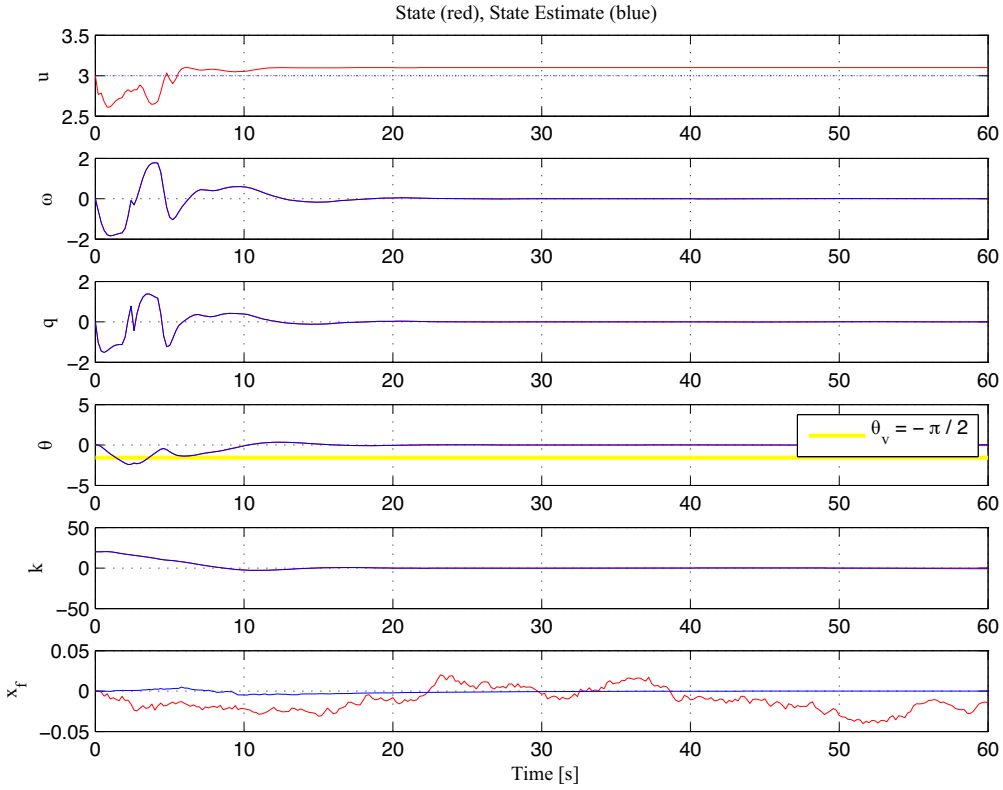
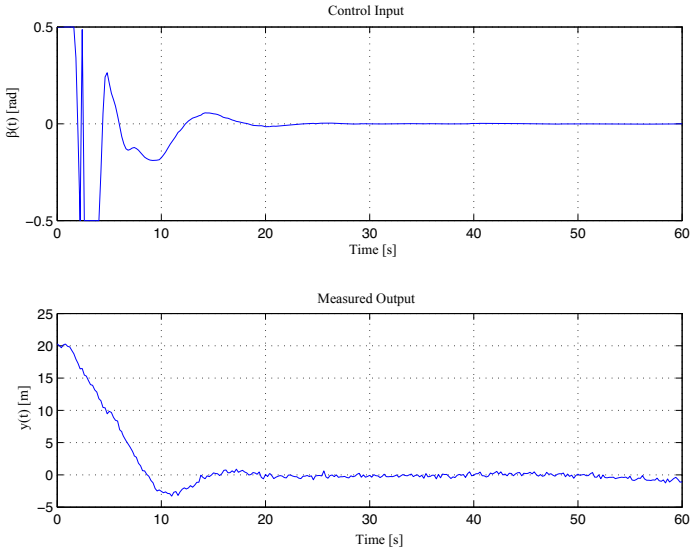
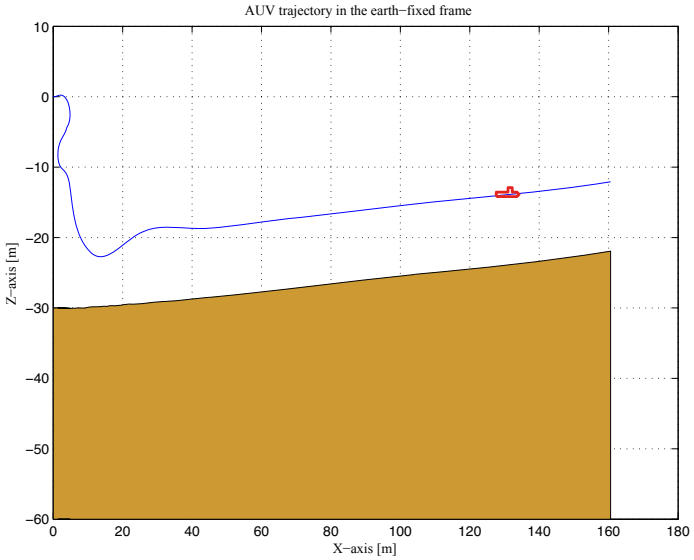


Figure 3.22: AUV real state, and EKF state estimate, using the LTV-LTI model and constant input weight. Initial condition of 20 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.23: Case with LTV-LTI model and constant input weight. Initial condition of 20 meters offset from the reference.

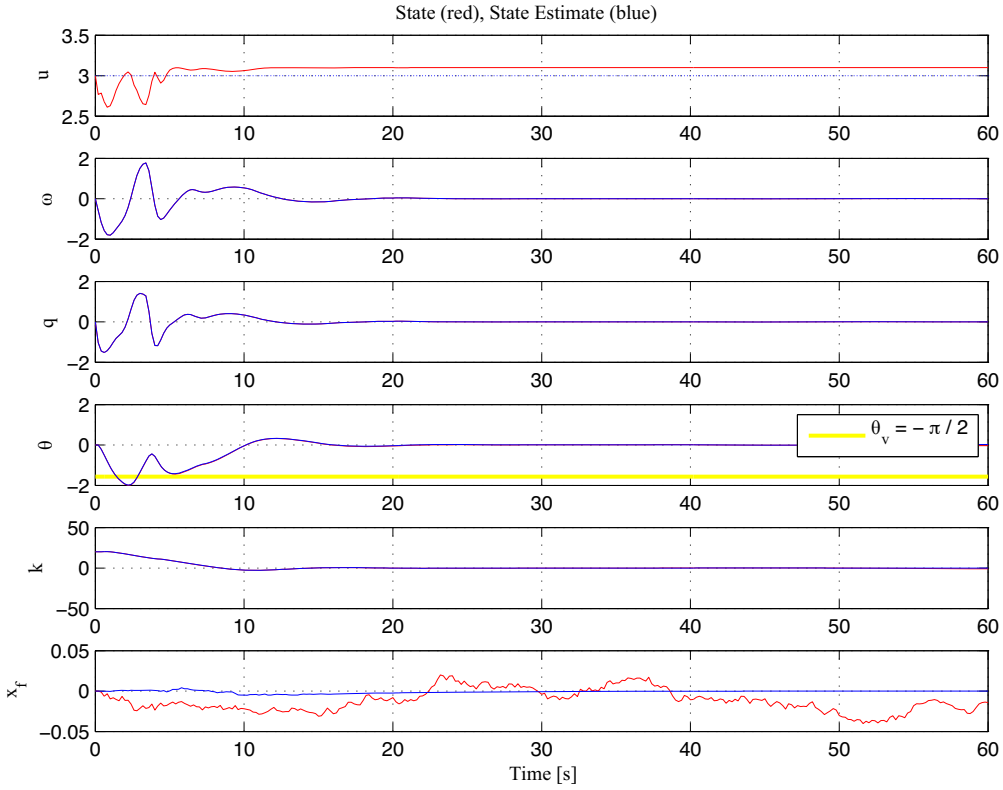
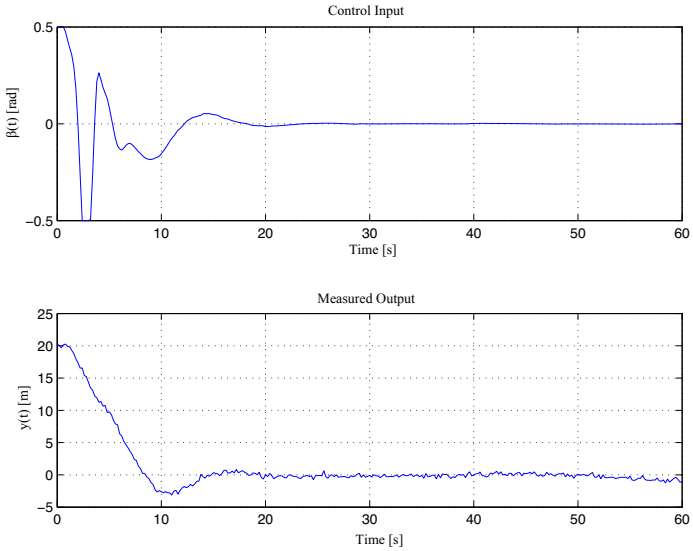
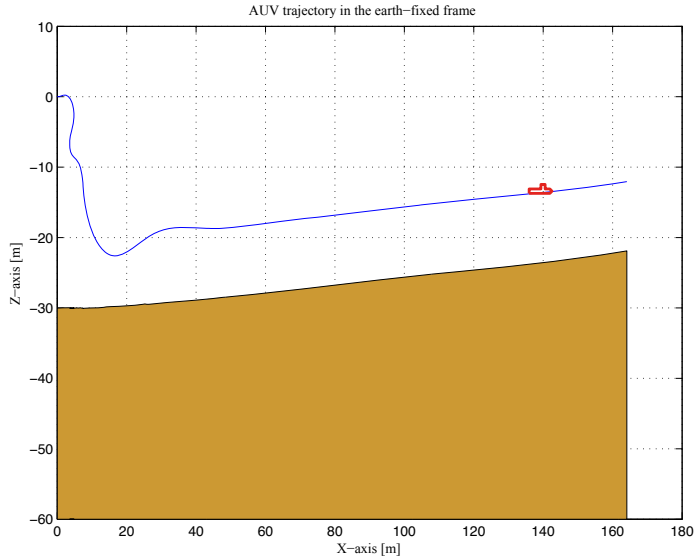


Figure 3.24: AUV real state, and EKF state estimate, using the LTV-LTI model and state dependent input weight. Initial condition of 20 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.25: Case with LTV-LTI model and state dependent input weight. Initial condition of 20 meters offset from the reference.

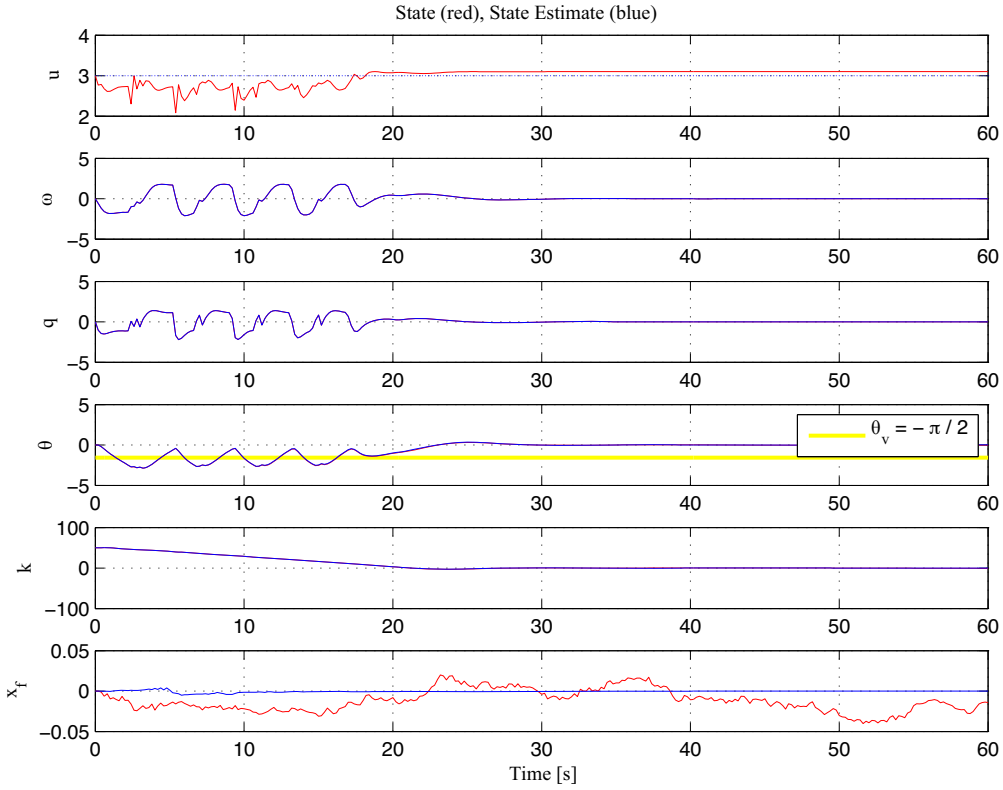
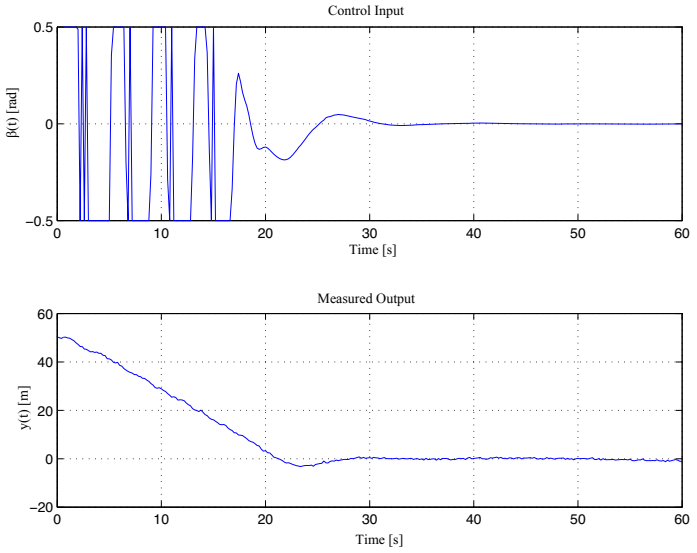
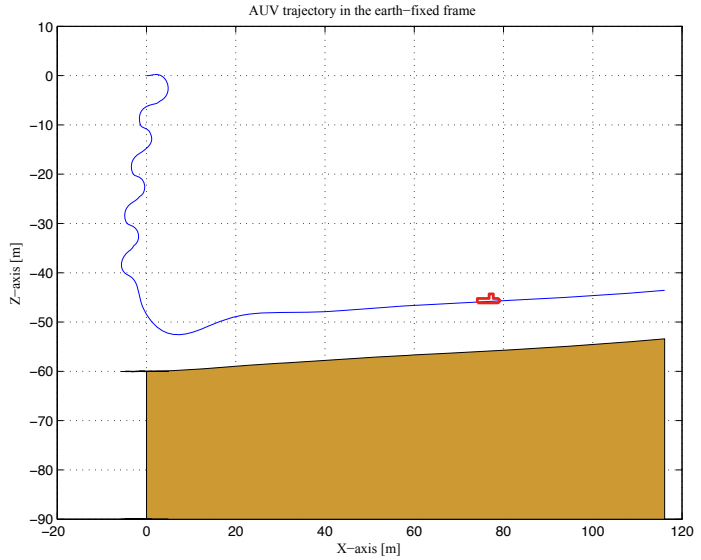


Figure 3.26: AUV real state, and its EKF state estimate, using the LTV-LTI model and constant input weight. Initial condition of 50 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.27: Case with LTV-LTI model and constant input weight. Initial condition of 50 meters offset from the reference.

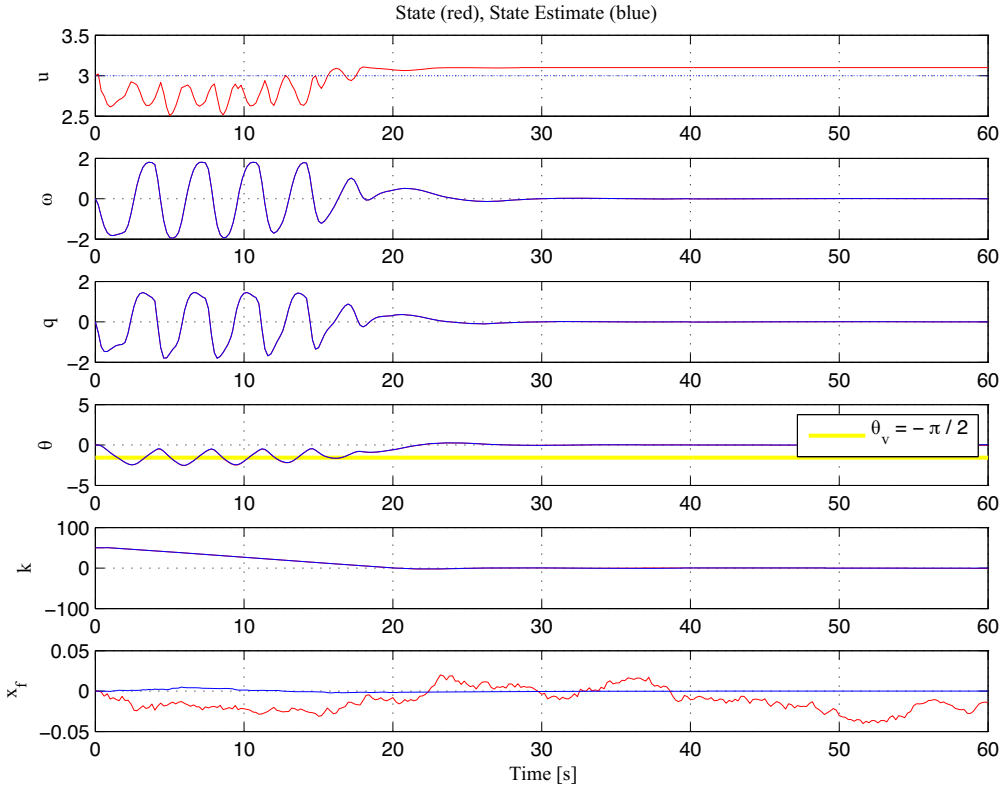
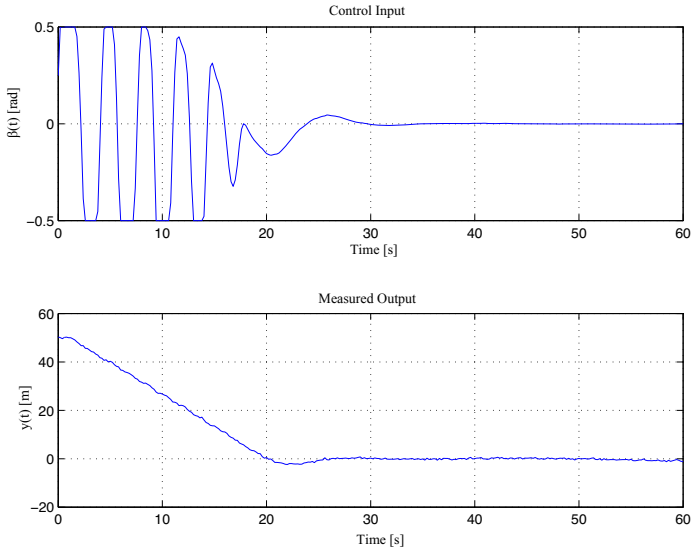
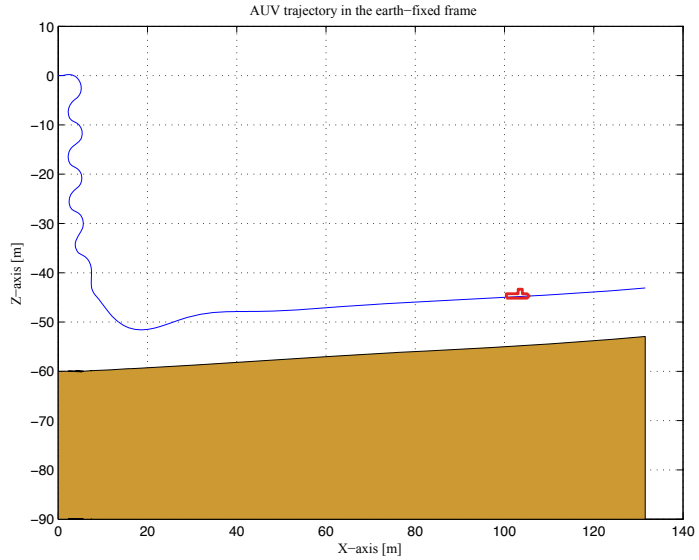


Figure 3.28: AUV real state, and EKF state estimate, using the LTV-LTI model and state dependent input weight. Initial condition of 50 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.29: Case with LTV-LTI model and state dependent input weight. Initial condition of 50 meters offset from the reference.

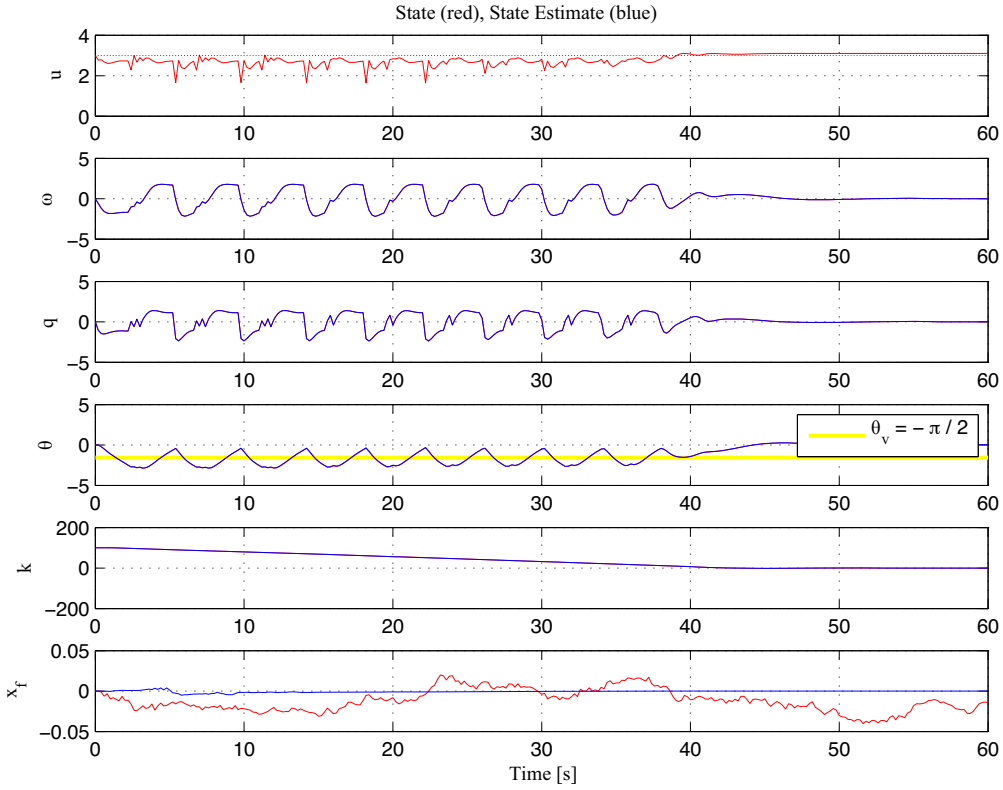
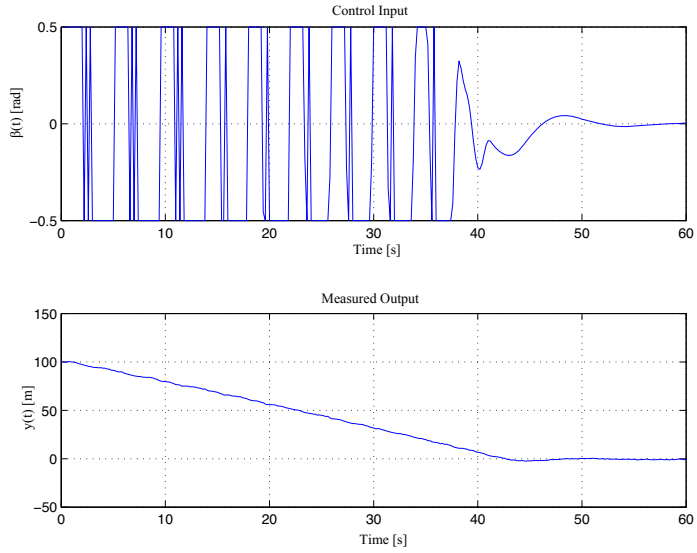
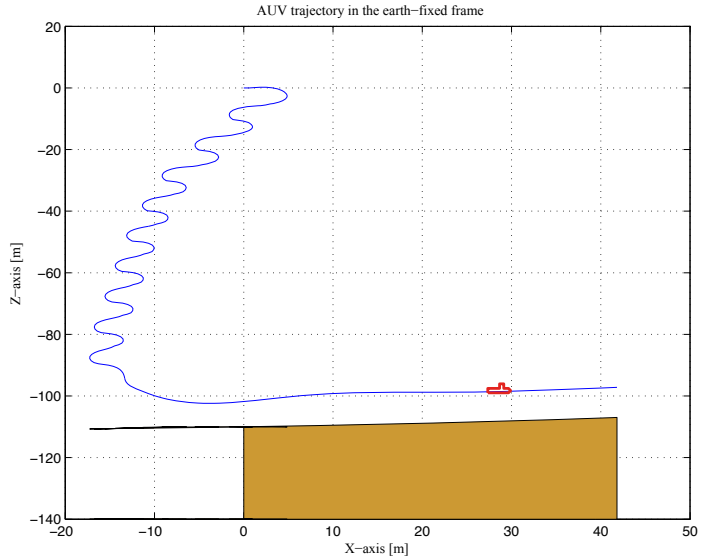


Figure 3.30: AUV real state, and its EKF state estimate, using the LTV-LTI model and constant input weight. Initial condition of 100 meters offset from the reference.



(a) AUV Control Input and Measured Output.



(b) AUV trajectory in the earth-fixed frame.

Figure 3.31: Case with LTV-LTI model and constant input weight. Initial condition of 100 meters offset from the reference.

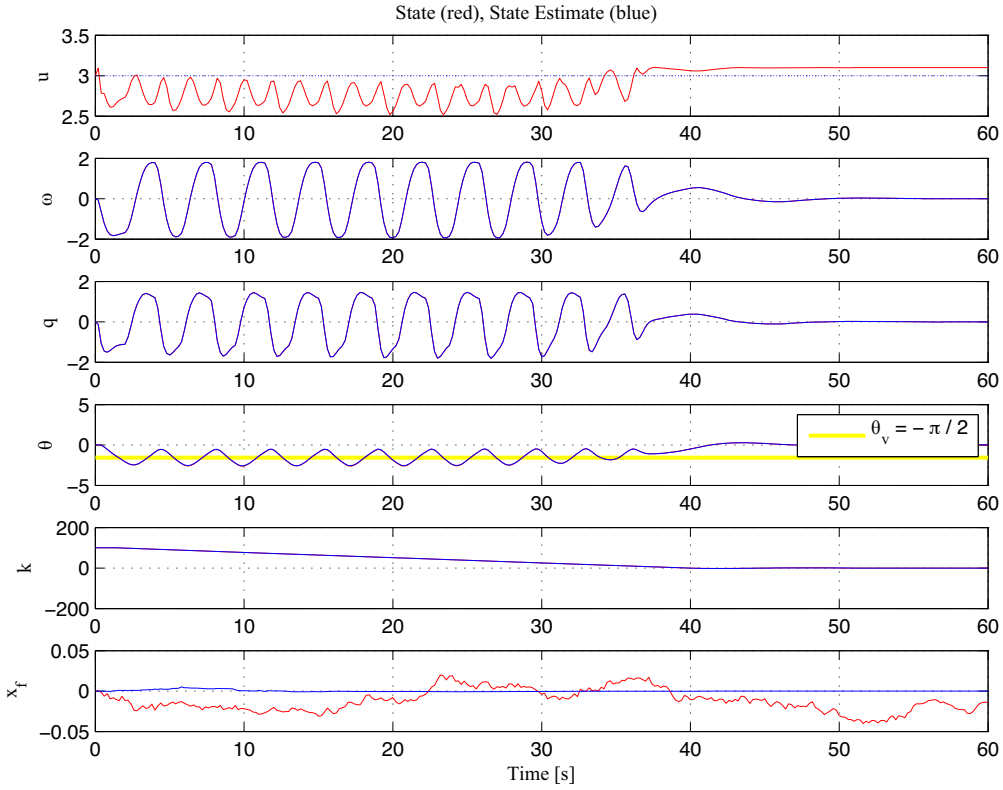
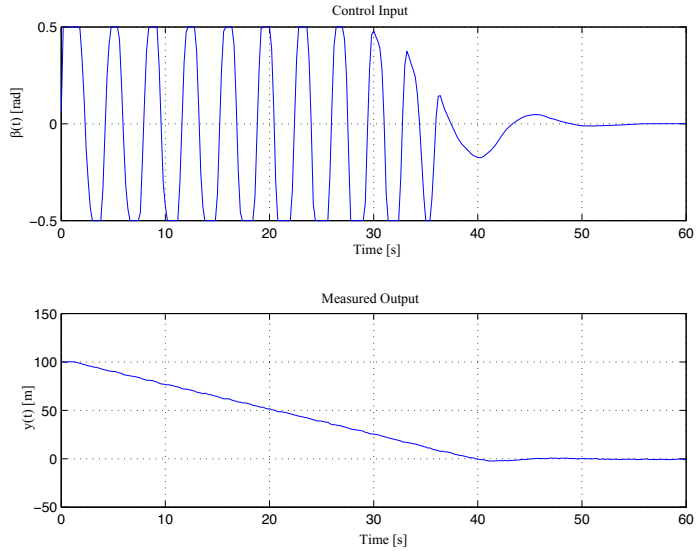
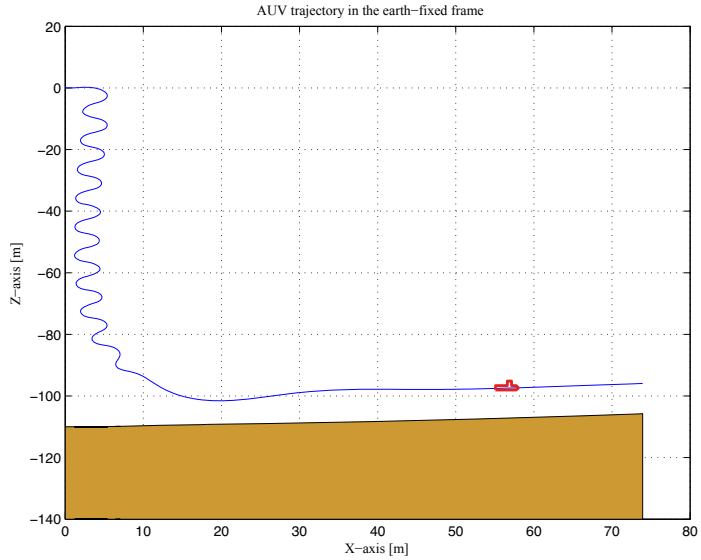


Figure 3.32: AUV real state, and EKF state estimate, using the LTV-LTI model and state dependent input weight. Initial condition of 100 meters offset from the reference.

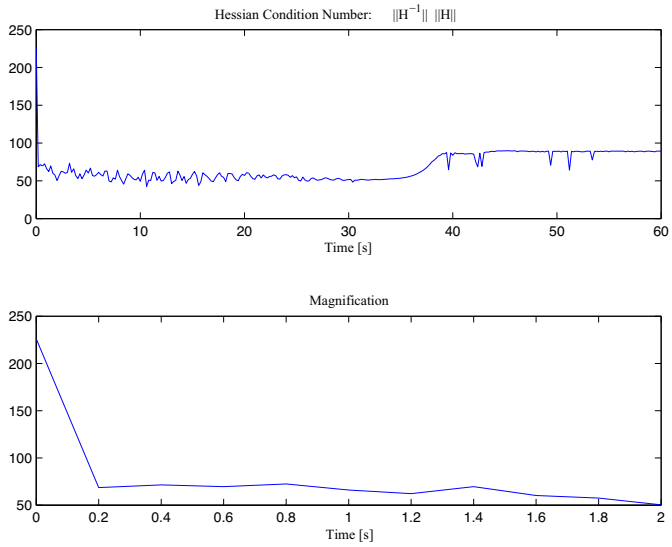


(a) AUV Control Input and Measured Output.

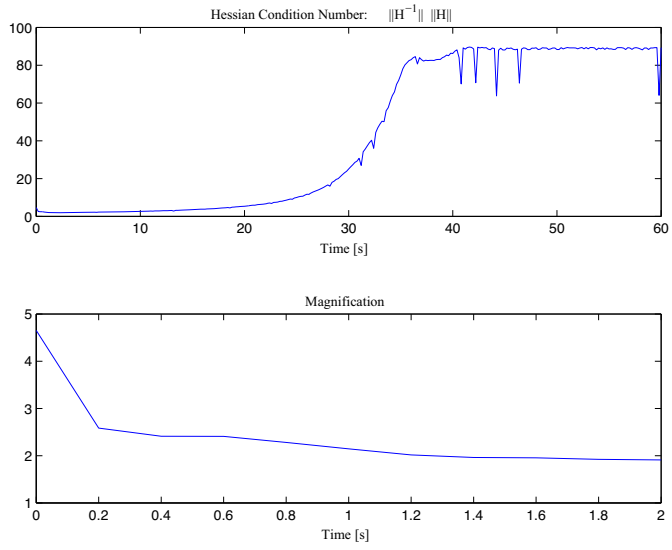


(b) AUV trajectory in the earth-fixed frame.

Figure 3.33: Case with LTV-LTI model and state dependent input weight. Initial condition of 100 meters offset from the reference.

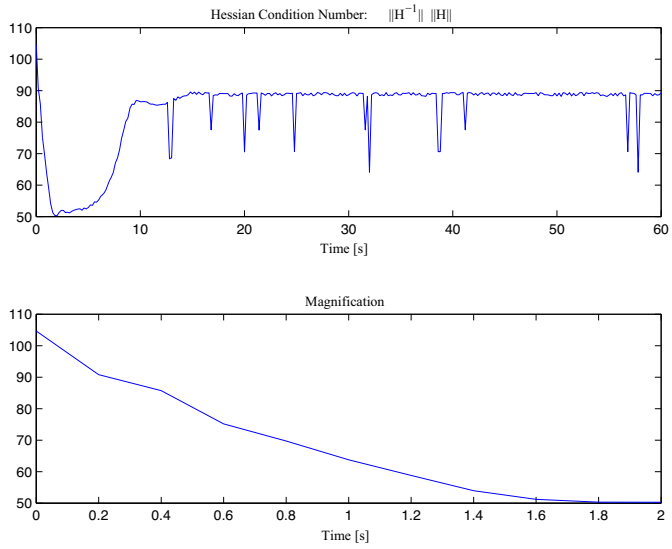


(a) Constant input weight.

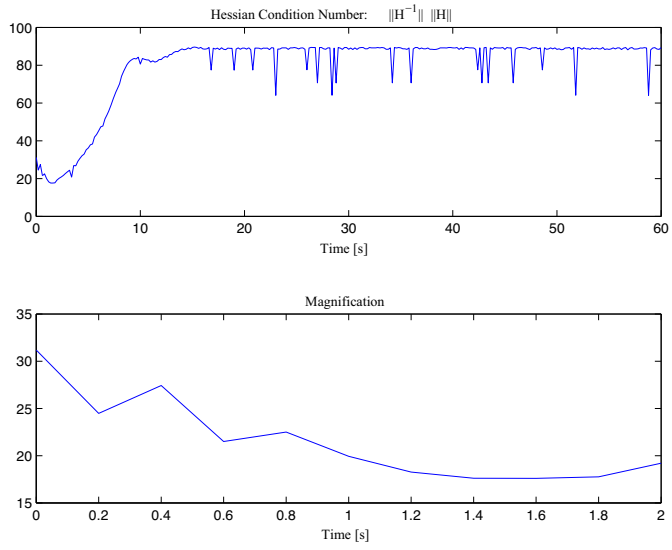


(b) State dependent input weight.

Figure 3.34: Hessian Condition Numbers, when using the LTV model. Initial condition of 100 meters offset from the reference.



(a) Constant input weight.



(b) State dependent input weight.

Figure 3.35: Hessian Condition Numbers, when using the LTV model. Initial condition of 20 meters offset from the reference.

3.4 Discussion

In order to clarify the contribution of this chapter, a brief introduction to the robustness of MPC is given. In addition the use of the state dependent input weight is discussed. Note that no attempt is made at providing an exhaustive exposition of the area of robust MPC. Instead, the focus will be on MPC for constrained, linear systems, which covers the majority of industrial applications [Qin & Badgwell, 2003]. The analysis of robustness for nonlinear systems is generally much more difficult, the interested reader should consult Rawlings & Mayne [2009] and references therein where some results in this direction are described.

3.4.1 A brief introduction to robustness of MPC controllers

The *robustness* of a closed-loop system indicates the extent to which the behavior of the system is sensitive to *uncertainty*. Here the term uncertainty refers to anything that may affect system behavior that is unknown at the controller design stage, such as errors in the model on which the controller design is based, unknown disturbances or measurement noise.

In linear systems theory, it is common to distinguish between Robust Stability (RS) and Robust Performance (RP) [Zhou et al., 1996]. For a robustly stable system, uncertainty cannot cause the closed-loop system to become unstable. Similarly , for a system with robust performance uncertainty can only cause modest performance degradation. H_∞ control theory provides powerful methods for the design and analysis of robustness for linear systems.

In MPC, a constrained optimization, such as (2.6), is solved for every time step and the optimal input is applied to the plant. If the initial state $\boldsymbol{x}_k$ is such that there is no sequence $\bar{\boldsymbol{u}}$ that fulfills constraints (2.6c)-(2.6e), thus the optimization problem is denominated *infeasible*. In general the fulfillment of the constraints is desired not only at a given time, but also at all subsequent times. In Chisci et al. [2001] it is shown how to ensure recursive feasibility and also how this can be ensured for bounded disturbances by appropriately restricting the constraints. The approach also ensures robust stability in the case of bounded disturbances. It is interesting to note that in case of non-constant disturbances there is no guarantee that the state will converge to the origin, however it will stays within the neighborhood of the origin. In similar fashion, Pluymers et al. [2005] shows how to ensure robust feasibility and robust stability for systems with parametric model

uncertainty, where the model uncertainty is represented as

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_k \boldsymbol{x}_k + \boldsymbol{B}_k \boldsymbol{u}_k \tag{3.38}$$

$$\boldsymbol{A}_k = \sum_{i=1}^{m} \lambda_{ik} \boldsymbol{A}_i$$

$$\boldsymbol{B}_k = \sum_{i=1}^{m} \lambda_{ik} \boldsymbol{B}_i$$

$$\sum_{i=1}^{m} 1, \qquad \lambda_{ik} \geq 0. \tag{3.39}$$

That is, the true plant must lie inside a polytope whose vertices are given by the 'extreme' models $(\boldsymbol{A}_i, \boldsymbol{B}_i)$. This approach may be adopted for both time-invariant and time-varying models. Kothare et al. [1996] shows an approach to deal with this model uncertainty by using Linear Matrix Inequalities.

The approaches in the papers referenced above are numerically tractable and potentially implementable. However, they address only robust feasibility and robust stability of uncertain, constrained linear systems. No attempt is made to optimize robust performance. Instead, nominal performance is optimized. Several authors have proposed optimizing robust performance by solving specific min-max optimization problems, see Mayne et al. [2000] and references therein. However, these latter approaches are not practicably tractable due to the computational burden, especially for online implementations [Rawlings & Mayne, 2009].

For an introduction to MPC for optimization of robust performance as well as robust MPC for nonlinear systems the reader is referred to Rawlings & Mayne [2009] and references therein.

3.4.2 Industrial approaches to robust MPC

This brief introduction to robust MPC above refers mainly to 'academic' approaches. Industrial MPC controllers seem to deal with this problem in a different way [Qin & Badgwell, 2003].

Feasibility issues are generally addressed by controlling what constraints are violated and by how much, instead of ensuring that constraints are never violated. This can be done by:

- Solving one or more smaller optimizations prior to solving the 'main' MPC problem. These first stage optimizations determine how much the main MPC constraints will have to be relaxed such that a feasible solution can exist.
- Alternatively, the main MPC problem can be augmented with slack variables in the

constraints and additional terms in the cost function to penalize constraint violations.

These approaches to ensure feasibility are discussed in works such as Scokaert & Rawlings [1999]; Hovd & Braatz [2001]; Vada [2000].

From the literature on robustness of linear systems it is easy to obtain an intuitive understanding of the robustness problems with ill-conditioned plants. Consider a Multiple-Input Multiple-Output plant $\boldsymbol{y} = \boldsymbol{M}\boldsymbol{u}$, with $\boldsymbol{y}$ output vector, $\boldsymbol{u}$ input vector. The model $\boldsymbol{M}$ is matrix valued, that is only the steady state response is considered for simplicity of presentation. The condition number $c_n(\boldsymbol{M})$ is defined as

$$c_n(\boldsymbol{M}) = \frac{\overline{\sigma}(\boldsymbol{M})}{\underline{\sigma}(\boldsymbol{M})} \tag{3.40}$$

where $\overline{\sigma}$ and $\underline{\sigma}$ are the largest and the smallest singular values, respectively.

The generalization to dynamic models in the frequency domain is straightforward if we consider $\boldsymbol{M}$, and hence also the singular values, as functions of frequency. [1] If $c_n(\boldsymbol{M}) = 1$, the gain from input to output is independent on the direction of the input, whereas a large $c_n(\boldsymbol{M})$ indicates that the gain is strongly dependent on input direction.

For a reference vector $\boldsymbol{r}$, the input $\boldsymbol{u}_0$ which removes the offset in the output is easily determined[2]

$$\boldsymbol{u}_0 = -\boldsymbol{M}^{-1}(\boldsymbol{y} - \boldsymbol{r}) \tag{3.41}$$

The singular values of $\boldsymbol{M}^{-1}$ are the inverses of the singular values of $\boldsymbol{M}$, which means that a large input will have to be applied in order to remove an offset in a direction where the input has a low gain. It is well known from linear algebra that a small change (error) in an ill-conditioned matrix can lead to a large change (error) in the inverse. Applied to our simple example this means that the controller will attempt to apply a large input in the low gain directions, but due to model error there are some differences between the low gain directions of the 'true' plant and the plant 'model'. As a result, part of this large input will miss the low gain direction of the true plant, and instead be fed into a high gain direction of the true plant. The result can be a large (unwanted) effect on the output. In a feedback system, such model errors can easily degrade performance and also jeopardize stability.

[1] Singular values are scaling dependent, so in order to use the singular values for analysis, the plant $\boldsymbol{M}$ needs to be appropriately scaled. Skogestad & Postlethwaite [2005] recommend scaling the inputs according to their range of actuation and the output according to the maximum acceptable offset in each output.

[2] Here issues regarding invertibility of a dynamic model are neglected, see Skogestad & Postlethwaite [2005] for details.

Consider now (2.6) - (2.7). It is possible to express the predicted future states as

$$\begin{bmatrix} \boldsymbol{x}_{k+1} \\ \vdots \\ \boldsymbol{x}_{k+N_p} \end{bmatrix} = \hat{\boldsymbol{A}}\boldsymbol{x}_k + \hat{\boldsymbol{B}} \begin{bmatrix} \boldsymbol{u}_k \\ \vdots \\ \boldsymbol{u}_{k+N_p-1} \end{bmatrix}. \tag{3.42}$$

Recalling the QP form (2.8), in which the Hessian has the following structure [Maciejowski, 2002]

$$\boldsymbol{H} = \hat{\boldsymbol{B}}^T \hat{\boldsymbol{Q}} \hat{\boldsymbol{B}} + \hat{\boldsymbol{R}} \tag{3.43}$$

where

$$\hat{\boldsymbol{B}} = \begin{bmatrix} \boldsymbol{B} & \boldsymbol{0} & \dots & \dots & \boldsymbol{0} \\ \boldsymbol{AB}+\boldsymbol{B} & \boldsymbol{B} & \boldsymbol{0} & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \boldsymbol{B} & \boldsymbol{0} \\ \sum_{i=0}^{N_p-1} \boldsymbol{A}^i\boldsymbol{B} & \dots & \dots & \boldsymbol{AB}+\boldsymbol{B} & \boldsymbol{B} \end{bmatrix}, \tag{3.44}$$

and

$$\hat{\boldsymbol{Q}} = \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \boldsymbol{0} \\ \boldsymbol{0} & \dots & \boldsymbol{0} & \boldsymbol{Q} \end{bmatrix}, \tag{3.45}$$

$$\hat{\boldsymbol{R}} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \boldsymbol{0} \\ \boldsymbol{0} & \dots & \boldsymbol{0} & \boldsymbol{R} \end{bmatrix}, \tag{3.46}$$

and the Gradient is instead

$$\boldsymbol{G} = \hat{\boldsymbol{B}}^T \hat{\boldsymbol{Q}} \hat{\boldsymbol{A}} \tag{3.47}$$

where

$$\hat{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{A}^2 \\ \vdots \\ \boldsymbol{A}^{N_p} \end{bmatrix}. \tag{3.48}$$

In the case of no active constraints, the optimal input sequence for MPC can easily be found by differentiating the criterion in (2.8a), and solving for the input sequence. This gives

$$\bar{\boldsymbol{u}}_k = \begin{bmatrix} \boldsymbol{u}_k \\ \vdots \\ \boldsymbol{u}_{k+N_p-1} \end{bmatrix} = -\boldsymbol{H}^{-1}\boldsymbol{G}\boldsymbol{x}_k = \left(\hat{\boldsymbol{B}}^T\hat{\boldsymbol{Q}}\hat{\boldsymbol{B}} + \hat{\boldsymbol{R}}\right)^{-1}\hat{\boldsymbol{B}}^T\hat{\boldsymbol{Q}}\hat{\boldsymbol{A}}\boldsymbol{x}_k \tag{3.49}$$

It is clear that if the Hessian matrix $\boldsymbol{H}$ is ill-conditioned, a small error in $\boldsymbol{H}$ (stemming from a small model error) can lead to a large error in $\boldsymbol{H}^{-1}$, and hence the effect of the input sequence may be far from excepted, with strong detrimental effect on control performance. Industrial MPC controllers often address this problem in one of the two following ways:

- Using Singular Value Thresholding (SVT), where the plant model (that is, $\hat{\boldsymbol{B}}$) is decomposed using Singular Value Decomposition (SVD), and small singular values are set to zero, and an approximate model $\tilde{\boldsymbol{B}}$ is assembled after setting the small singular values to zero. The approximate model is then used when solving the MPC optimization problem. Inspecting (3.49), it is clear that singular value thresholding does not actually reduce the ill-conditioning of the Hessian. Instead, the predictions are effectively modified such that the controller 'does not see' future offsets in directions where the inputs have little effect. SVT is used in Honeywell's Robust Multivariable Predictive Control Technology (RMPCT) [Qin & Badgwell, 2003].
- Increasing the input weight $\boldsymbol{R}$ (that is, the weight on *all* inputs) reduces the ill-conditioning of the Hessian, and thus avoids excessive input moves. This technique is used in Aspentech's DMCplus controller [Qin & Badgwell, 2003].

It should be noted that SVT deliberately introduces a model error, whereas increasing the input weight detunes the controller and makes nominal response slower. Thus, both approaches reduce nominal performance. The use of these techniques in industrial controllers therefore reflects the fact that most MPC controllers in industry are implemented on open-loop stable systems. These two 'robustification techniques' are studied and compared in Aoyama et al. [1997].

3.4.3 The state dependent input weight

Considering the brief introduction to the robustness of MPC controllers presented above, it is clear that the use of the state dependent input weight is a refinement of the industrially popular technique of increasing the input weight. Making the input weight state dependent (when properly applied) will, however, only reduce the controller gain when this is required by the operating conditions, and will not reduce performance in 'safe' operating regions. From this point of view, the state dependent input weight introduces gain scheduling into the MPC controller, as claimed in the introduction to this chapter.

The state dependent input weight requires that the control engineer has a sound understanding of the control problem, and implementing it in standard MPC controllers (with a quadratic objective function) is very simple. Also, the weight calculation results in a negligible increase in computational cost[1].

3.5 Conclusions

The industrial 'robustification techniques', described in Section 3.4.2, are motivated as ways of avoiding drastic performance degradation stemming from small model errors. For the AUV example, it seems clear that these model errors stem from using a linearized model of the AUV. The linearized model used in the MPC produces then an ill-conditioned optimization problem. This generates performance degradation, and in some cases the control fails to reach its goal.

A model predictive control framework for the depth control of an underwater vehicle is presented where different MPC approaches are implemented. It is shown how by using an LTV model, and the state dependent input weight it is possible to improve the region of attraction of the controller and have a smooth control action. The nonlinear input weight definition is inspired by the gain-scheduling technique. It helps to better adapt the control response to the initial condition (depth of the AUV). In fact, although the system is nonlinear, the state dependent input weight allows the use of linear models for prediction, giving the best performance with the LTV model. Related to this gain-scheduling effect is the reduction in the condition number of the Hessian that results from the state dependent input weight. As discussed, reducing the ill-conditioning of the Hessian is a common approach to improve the robustness of industrial MPC. This is of particular importance for this problem when the AUV is far from the reference and large inputs would otherwise be demanded.

An alternative approach is to solve the *nonlinear* model equations, which would typically require solving a Sequential Quadratic Programming (SQP) problem [2] (Nocedal & Wright [2000]), instead of solving a single QP problem at each time step, as briefly introduced in Section (2.3.3). Such an approach may be computationally tractable, but the computational cost (and difficulty of coding) is much higher than for the state dependent input weight, which requires solving only a single QP at each time step. In addition, even if all available physical knowledge may be applied when formulating the nonlinear model, all models are in practice uncertain to some extent. Thus, although direct use of the nonlinear model removes inaccuracies stemming from model linearization, the resulting

[1] Actually, for ordinary MPC problems where the optimization problem is solved online, the state dependent input weight probably saves online computing compared to solving QP problems with ill-conditioned Hessians.

[2] That is, as sequence of QP problems that gradually converge to the solution of the nonlinear problem.

MPC controller may nevertheless still experience robustness problems.

In summary, the state dependent input weight is clearly related to approaches in use in industry to enhance robustness of MPC controllers. Although more sophisticated methods exist in the academic literature, there seems to be little need for such methods for the problem studied here. The main advantages of using the state dependent input weight is the low computational cost and easy implementation, compared with min-max robustification techniques, or SQP, for example. With the state dependent approach the controller gain is increased smoothly and automatically when entering the safe operating region. The main disadvantage is the need for the control engineer to know where the safe operating region is, in order to be able to reliably design the state dependent input weight.

Chapter 4

Persistently Exciting Model Predictive Control, a dual control approach

In this chapter, MPC is considered in an adaptive context where the model adjustment requirements are built into the MPC formulation. This is a general question of dealing with the dual adaptive control problem, discussed in Section 4.1.1. The input has the dual purpose of regulating the system via feedback and providing the excitation necessary for the identification and adaptation of the system. It is well known that these two purposes are generally in conflict. Further, it is well known that attempting to optimize a control performance criterion over both the regulation and excitation frameworks is essentially an intractable problem. However, several interesting results have been obtained by approximating the original dual control problem.

Fundamentally, MPC is a non-dynamic or memoryless state feedback control. Because of its use of a model, MPC should be amenable to adaptive implementation and to online tuning of the model. Such an approach requires guaranteeing signal properties, known as 'persistent excitation', to ensure uniform identifiability of the model, often expressed in terms of spectral content or 'sufficient richness' of a periodic input. Thus, an approach to augment the input constraint set of MPC to provide this guarantee is proposed. This requires equipping the controller with its own state to capture the control signal history. The feasibility of periodic signals for this condition is also established, and several computational examples are presented to illustrate the technique and its properties.

The approach taken here is to augment the constrained optimization of MPC to include an additional constraint to ensure that a minimal level of excitation is preserved. The normal statement of the persistence of excitation condition (PEC) is via uniform positive definite matrix bounds on the running summed outer products of the regressor sequence in the system identification. In the context of open-loop, stationary, linear system identification of models with N parameters to be identified, this excitation requirement is reinterpreted as the input signal sufficient richness condition (SRC) that the vectors com-

posed of N delayed values of the input signal themselves satisfy the PEC. In turn and still for stationary scalar systems, this condition can be converted to a requirement that the input signal contains at least N distinct complex frequency components [Goodwin & Sin, 1984], that is, as many complex sinusoids as there are parameters to identify. Although the terms PEC and SRC are used almost interchangeably [Green & Moore, 1986], SRC is reserved here for the PEC property pertaining to vectors composed solely of the manipulable input signal.

For closed-loop system identification the regressor PEC requirement is normally transferred to the reference input signal and requires the stability and/or stationarity of the closed-loop to achieve a signal upper bound. An equivalent condition to ensure the convergence of system identification using pseudo-linear regression is that the input signal to the plant yields a persistently exciting pseudo-regressor sequence, in order for the parameter estimation error algorithm to converge sufficiently rapidly. For open-loop output error schemes applied to minimal systems satisfying the standard positive real condition, Anderson & Johnson Jr [1982] show that input richness conditions alone suffice for the entire regressor to satisfy the excitation condition. Accordingly, for many standard linear system identification schemes, the uniform convergence of the parameter estimator may be ensured provided the plant input signal $\{\boldsymbol{u}_k\}$ satisfies a sufficient richness property. The development in this chapter, denominated Persistently Exciting MPC (PE-MPC), is to include an input sufficient richness constraint directly into the MPC formulation. In the absence of prevailing signal conditions in the MPC-controlled loop yielding excitation, this constraint becomes active to ensure plant identifiability in closed-loop. It is shown that, in this circumstance, periodic inputs are feasible.

Due to the focus of MPC on stabilization and regulation, the controller does not need to inherently produce a plant input signal $\{\boldsymbol{u}_k\}$ that satisfies the SRC. Indeed, the inclusion of input richness into PE-MPC, since it adds another constraint, could diminish regulation performance in line with expectations from dual adaptive control. Different approaches have been proposed earlier to ensure richness. For example, in Sotomayor et al. [2009] it is shown that by incorporating an additive external signal, also known as a dithering signal, it is possible to obtain sufficiently rich closed-loop signals for process identifiability. However, in an MPC framework, an unfortunate choice of dithering signal might cause constraint violation. For example, after the manipulated variable is calculated, if the magnitude of a dithering signal is large enough, some input or output constraint may be violated. One might accommodate the dither by tightening the constraints to conservative values. However, the dither must diminish control performance even when the closed-loop signals are otherwise persistently exciting by dint of, say, reference changes or other transients. In Genceli & Nikolaou [1996] an approach, named MPCI, is presented. In this case, the input is forced to be persistently exciting by imposing a Persistently Exciting (PE) constraint. However, the MPCI has the drawback that its formulation contains the definition of an equality constraint to enforce periodic solutions.

Another approach is Hovd & Bitmead [2004], where an augmented cost function is introduced in an NMPC contest to add the dual effect necessary such that state estimation is improved.

We consider incorporating the input PEC into the formulation of an MPC controller through the inclusion of an additional constraint, guaranteeing a sufficiently rich input signal but with the possibility of this constraint being inactive during normal system transients. In the stationary case, this approach admits, rather than imposes, a periodic input. Since MPC involves the receding horizon computation at time k of a plant-state-dependent control sequence, $\{\boldsymbol{u}_k, \boldsymbol{u}_{k+1}, \ldots, \boldsymbol{u}_{k+N_p-1}\}$, of which only the initial element, $\boldsymbol{u}_k$, is applied. The remaining elements are relegated to the initial feasible solution of the subsequent solution and sufficient richness asserted on the forward-looking N_p-step MPC solution need not occur as result, because of the optimization problem restatement after the application of the first element. Thus, it is necessary to introduce a state into the MPC controller to effectively look backwards to ensure that $\boldsymbol{u}_k$ satisfies the SRC with respect to its immediate predecessor inputs. This is a novelty when compared to the formulation of MPC as a non-dynamic (memoryless) function of the plant state; PE-MPC is now a dynamic plant state feedback control law. By the same token, our formulation of PE-MPC associates this new constraint only with the first manipulated variable $\boldsymbol{u}_k$. This has a direct consequence on the analysis of the constraint, simplifying the solution of the corresponding optimization problem. Finally, we also need to assume for our analysis, as with most early approaches to adaptive control, that other mechanisms are at play to ensure the closed-loop system is stabilized so that an upper bound is available for the input to ensure that the PEC is a feasible constraint for the MPC problem. That is, our methods will need to assume and rely on the effectiveness and feasibility of the regulation side of the MPC controller.

In this chapter, we shall denote by: n the plant system state dimension, m the input dimension, p the output dimension, N_p the control and prediction horizon of the MPC, N the number of parameters to be identified in a plant model, P the input sufficient richness backwards-looking horizon to be defined.

4.1 Background results

In this section, a short general introduction to dual control is given. Then, technical concepts are defined and introduced to arrive at the main results of the chapter. These background results are fundamental to understand how the Persistently Exciting Model Predictive Control formulation is obtained.

4.1.1 Fundamentals of Dual Control theory

In general, control systems are designed to spend as little control effort as possible in order to obtain the required control performance. That is, a controller that achieves its performance specifications while using the minimum amount of energy is wanted. However, when we have to learn from a system, it is necessary to have a certain level of excitation, such that the 'learning procedure' can work properly. To obtain a minimum level of excitation, a certain level of energy has to be used. Thus, it is obvious that control action and learning action are in conflict. The compromise between these two actions leads to the dual control problem.

The dual control problem was first defined and introduced by Feldbaum at the beginning of the Sixties with a series of four seminal papers, [Feldbaum, 1960a,b, 1961a,b]. In his work he discussed the problem of conflict between control quality and information gathering. Feldbaum introduced the dual effect and neutral system notions, discussed more in detail in next section. He defined the dual control system such as a system characterized by having a dual effect.

It is difficult to have a formal definition of dual control, however as mentioned in Brustad [1991], the following definition may be given:

Definition 4.1.1. Dual controller:
A dual controller is a model-based controller where the model is, to some degree, uncertain, and where the controller in addition to its usual control task, spends some energy in reducing the model uncertainties (learning) in such a way that the overall control performance is improved.

This is a general definition which covers a large range of controllers. The scope of this chapter is to describe the dual control problem in a general manner. It is hoped that this introduction to dual control will make easier to appreciate the results in the chapter.

In 2000, IEEE Control System Society listed dual control as one of the 25 most prominent subjects in the last century which had a great impact on control theory development [Li et al., 2005].

In principle, the optimal solution of a dual control problem can be found by dynamic programming, by solving the Bellman functional equation. However, it is well known that is very difficult to find an analytical recursive solution, and the problem is generally considered intractable [Kumar & Varaiya, 1986]. Moreover, the dimensionality of the underling spaces causes several numerical problems that makes this problem practically unsolvable [Bayard & Eslami, 1985], [Bar-Shalom & Tse, 1976].

Dual effect and neutral systems

The principal characteristic of a dual controller is to have a dual effect. Feldbaum described the dual effect as the application of the following two actions,

- directing, and
- investigating.

In Bar-Shalom [1981] both features are described as caution and probing, respectively. In more detail, a dual controller is capable of

- driving the system towards a desired state (directing);
- performing some learning procedure such that system uncertainties are reduced (investigating).

Typically, these two aspects are in conflict, and they are source of difficulties when finding the optimal dual control solution.

It is worth mentioning that not every system can have a dual effect. In fact it is possible to define a neutral system [Bar-Shalom & Tse, 1974].

Definition 4.1.2. Neutral system:
A system that does not have a dual effect is called neutral.

A practical way to understand whether a system is neutral or not is to check that the uncertainty in the system is totally independent from the past control sequence.

In Brustad [1991] an example of neutral system is given. It is shown that a linear Gaussian system does not have dual effect. Consider the problem of estimating the state of the following linear system

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{v}_k \\ \boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{w}_k \end{aligned} \tag{4.1}$$

where $\boldsymbol{v}_k$ and $\boldsymbol{w}_k$ are white Gaussian noise distributed signals, with covariances $\boldsymbol{P}^v$ and $\boldsymbol{P}^w$, respectively. The uncertainty in the system is expressed by the state covariance matrix $\boldsymbol{P}^x$. Using the standard Kalman Filter (see Chapter 5) the covariance is computed by

$$\boldsymbol{K}_k = \boldsymbol{P}^x_{k|k-1}\boldsymbol{C}^T(\boldsymbol{C}\boldsymbol{P}^x_{k|k-1}\boldsymbol{C} + \boldsymbol{P}^w)^{-1} \tag{4.2}$$

$$\boldsymbol{P}^x_{k|k} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C})\boldsymbol{P}^x_{k|k-1} \tag{4.3}$$

$$\boldsymbol{P}^x_{k+1|k} = \boldsymbol{A}\boldsymbol{P}^x_{k|k}\boldsymbol{A}^T + \boldsymbol{P}^v \tag{4.4}$$

Thus, as it can be seen, $\boldsymbol{A}$, $\boldsymbol{C}$, $\boldsymbol{P}^v$, and $\boldsymbol{P}^w$ are all independent of the input $\boldsymbol{u}_k$. Therefore, the state covariance matrix $\boldsymbol{P}^x$ is totally independent from the past input sequence, and thus the system (4.1) is neutral.

Formulation

In Filatov & Unbehauen [2004] a general formulation of the dual control problem is presented. The formal solution using Stochastic Dynamic Programming is given, and the main difficulties on finding the optimal solution are discussed. It is shown how to improve (but not optimize) the performance, using the dual effect, of different adaptive control problems without excessively complicating the original algorithms, such that it is possible to have practical online implementations.

According to Filatov & Unbehauen [2004], let us define the following dual control problem. Consider the state space representation

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\omega}_k; \boldsymbol{\theta}_k) \tag{4.5a}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{v}_k; \boldsymbol{\theta}_k) \tag{4.5b}$$

where $\boldsymbol{x}_k \in \mathcal{R}^n$, $\boldsymbol{u}_k \in \mathcal{R}^m$, $\boldsymbol{y}_k \in \mathcal{R}^p$, $\boldsymbol{\omega}_k$ and $\boldsymbol{v}_k$ are sequences of independent zero-mean random variables, and $\boldsymbol{\theta}_k \in \mathcal{R}^N$ denotes the vector of unknown parameters, with assumed time-varying dynamic

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{p}(\boldsymbol{\theta}_k, \boldsymbol{\epsilon}_k) \tag{4.6}$$

where the function $\boldsymbol{p}(\cdot)$ describes the time-varying nature of the parameter vector $\boldsymbol{\theta}$, and $\boldsymbol{\epsilon}_k$ is a vector of independent random white noise sequence with zero-mean and known probability distribution.

Control inputs and outputs, available at time k, are denoted by

$$\boldsymbol{\Upsilon}_k = \{\boldsymbol{u}_{k-1}, \boldsymbol{u}_{k-2}, \ldots, \boldsymbol{u}_0, \boldsymbol{y}_k, \boldsymbol{y}_{k-1}, \ldots, \boldsymbol{y}_0\} \tag{4.7}$$

for $k = 1, \ldots, N_p - 1$ and $\boldsymbol{\Upsilon}_0 = \{\boldsymbol{y}_0\}$.

The performance index

$$J = \mathrm{E}\left\{ \sum_{k=0}^{N_p-1} g(\boldsymbol{x}_{k+1}, \boldsymbol{u}_k) \right\} \tag{4.8}$$

is used for control optimization purpose, where $g(\cdot)$ is a positive, convex, scalar function. The operator $\mathrm{E}\{\cdot\}$ denotes the expectation, and it is taken with respect to all the random variables $\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{\omega}, \boldsymbol{v}, \boldsymbol{\epsilon}$.

The dual control problem consists of finding the control policy

$$\boldsymbol{u}_k(\boldsymbol{\Upsilon}_k) \in \boldsymbol{\Omega}_k, \qquad k = 0, 1, \ldots, N_p - 1 \tag{4.9}$$

that minimizes the performance index (4.8) for the system (4.5), where the unknown parameter dynamics is described in (4.6). The optimal control sequence has to be contained in $\boldsymbol{\Omega}_k$, that defines the admissible control values domain, in the space $\mathcal{R}^m$.

To find the optimal solution (4.9), stochastic dynamic programming is used [Kall & Wallace, 1994; Birge & Louveax, 1997]. Thus the backward recursion

$$J_{N_p-1}^{CLO}(\Upsilon_{N_p-1}) = \min_{\boldsymbol{u}_{N_p-1}\in\boldsymbol{\Omega}_{N_p-1}} \left[\mathrm{E}\left\{g[\boldsymbol{x}_{N_p}, \boldsymbol{u}_{N_p-1}]|\Upsilon_{N_p-1}\right\}\right], \tag{4.10}$$

$$J_{k}^{CLO}(\Upsilon_{k}) = \min_{\boldsymbol{u}_{k}\in\boldsymbol{\Omega}_{k}} \left[\mathrm{E}\left\{g[\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k}] + J_{k+1}^{CLO}(\Upsilon_{k+1})|\Upsilon_{k}\right\}\right], \tag{4.11}$$

for $k = N_p - 2, N_p - 3, \ldots, 0$ is obtained. The superscript CLO means Closed-Loop Optimal, that is the current control signal is obtained by considering knowledge about dynamic, probability statistics, and input/output measurements, in accordance to Bar-Shalom & Tse [1976] formulation. A simple recursive solution of (4.10) and (4.11) is difficult to find due to both analytical difficulties and numerical issues of the problem dimensionality. Therefore, even for small size, simple problems it is, with few exceptions, practically impossible to find solutions [Bar-Shalom & Tse, 1976; Bayard & Eslami, 1985]. One notable exception is Sternby [1976], who presents a simple example of a dual control problem with an analytical solution. However, the dual control approach is trying to simplify the problem such that a possible numerical solution is found. Thus, the analysis of this formulation may lead to insights that can help identify the main dual properties. This can then be used to define a simpler problem, which is an approximation, but contains the dual effect. The advantage is that the new problem is practically solvable, as in Wittermark [2002], for example. Note that, however, the solution of an approximated dual control problem is typically sub-optimal rather than truly optimal.

An example of an application of dual effect, probing and controlling, is in Allison et al. [1995], where it is shown how to implement a dual control strategy for a wood chip refiner. A more recent application to a coupled multivariable process for paper coating is shown in Ismail et al. [2003], where the authors implement an adaptive dual control law, and a Kalman Filter for estimating the variable process gain. Moreover, the controller is run for a trial period in the mill with successful results. This is an example of how adding the probing effect, in addition to the control effect, can substantially improve the production.

Difficulties in finding the optimal solution of the dual control problem (4.10-4.10) have contributed to discovering several approaches for deriving simpler sub-optimal dual controllers. As mentioned in Wittermark [1995], some approximated dual controllers may be obtained by constraining the variance of parameter estimates, using serial expansion, or even modification, of the performance index (4.8), using ideas from robust control design. Other approximation policies are discussed in Filatov & Unbehauen [2004], where an exhaustive classification of sub-optimal dual controllers is given.

Fundamentally, dual control approximations may be gathered into two distinct categories.

- Implicit Dual Control
- Explicit Dual Control

The first category contains all methods obtained by approximating the original dual control problem (4.10-4.10), or the performance index (4.8). The main characteristic of these controllers is that they provide good performance, but due to a significant computational requirement their application in real time is practically restricted.

The second category contains all ad-hoc controllers, and all methods based on a sort of bi-criterial approach. The bi-criterial approach can be obtained by modifying the performance index (4.8), such as two distinct parts may be defined

$$J = J^c + J^e \tag{4.12}$$

where J^c is the cost function term that contributes to the 'control' part of the resulting controller, while J^e is the cost function term that contributes to the 'excitation' level of the resulting control signal. This particular form of the performance index (or cost function) makes it possible to explicitly enforce the dual effect.

The persistently exciting model predictive control formulation given in this chapter can be classified as an explicit dual controller.

4.1.2 Schur complement

Given a matrix

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{P} & \boldsymbol{Q} \\ \boldsymbol{R} & \boldsymbol{S} \end{bmatrix}, \tag{4.13}$$

with nonsingular trailing principal square sub-matrix $\boldsymbol{S}$, the Schur complement is defined as

$$\boldsymbol{M}/\boldsymbol{S} = \boldsymbol{P} - \boldsymbol{Q}\boldsymbol{S}^{-1}\boldsymbol{R}. \tag{4.14}$$

The notation $\boldsymbol{M}/\boldsymbol{S}$ indicates the Schur complement of $\boldsymbol{S}$ in $\boldsymbol{M}$.

Definition 4.1.3. *Inertia of Hermitian Matrices*
The inertia of an $n \times n$ Hermitian matrix $\boldsymbol{H}$ is the ordered triple

$$In(\boldsymbol{H}) := (p(\boldsymbol{H}), q(\boldsymbol{H}), z(\boldsymbol{H})) \tag{4.15}$$

where $p(\boldsymbol{H})$, $q(\boldsymbol{H})$, and $z(\boldsymbol{H})$ are the numbers of positive, negative, and zero eigenvalues of $\boldsymbol{H}$ including multiplicity, respectively. The rank is $\operatorname{rank}(\boldsymbol{H}) = p(\boldsymbol{H}) + q(\boldsymbol{H})$.

Theorem 4.1.1 (Zhang [2005])**.** *Let $\boldsymbol{H}$ be a Hermitian matrix and let $\boldsymbol{S}$ be a nonsingular principal sub-matrix of $\boldsymbol{H}$. Then*

$$In(\boldsymbol{H}) = In(\boldsymbol{S}) + In(\boldsymbol{H}/\boldsymbol{S}). \tag{4.16}$$

A direct consequence of this theorem is the following.

Theorem 4.1.2. *Let $\boldsymbol{H}$ be a Hermitian matrix partitioned as*

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{H}_{11} & \boldsymbol{H}_{12} \\ \boldsymbol{H}_{12}^{*} & \boldsymbol{H}_{22} \end{bmatrix} \tag{4.17}$$

where $\boldsymbol{H}_{22}$ is square and nonsingular. Then $\boldsymbol{H} > 0$ if and only if both $\boldsymbol{H}_{22} > 0$ and $\boldsymbol{H}/\boldsymbol{H}_{22} > 0$.

4.1.3 Persistence of excitation for adaptive control schemes

In general, for uniform convergence of system identification schemes it is desired that the regressor vectors generated in the time-varying closed-loop must be persistently exciting, i.e. summed outer products of the regression vector, $\boldsymbol{\psi}_k$, should satisfy uniform boundedness conditions.

Definition 4.1.4. *[Goodwin & Sin, Sec 3.4 Goodwin & Sin [1984]] A sequence of vectors $\{\boldsymbol{\psi}_k : k = 1, 2, \dots\}$ in $\mathcal{R}^N$ is persistently exciting if there exist real numbers α and β and integer P such that*

$$0 < \alpha I_N \leq \sum_{k=k_0+1}^{k_0+P} \boldsymbol{\psi}_k^T \boldsymbol{\psi}_k \leq \beta I_N < \infty, \quad \forall k_0.$$

This is not easily verifiable, instead it is more convenient to investigate how the persistence of excitation propagates from input signal to the regressors. This problem has been studied by a number of researchers and follows if the regressor vectors are reachable from the input. For Single-Input, Single-Output (SISO) cases, the authors in Anderson & Johnson Jr [1982]; Mareels [1985]; Dasgupta et al. [1990] have derived conditions on the input that guarantee parameter convergence. For multivariable cases, results have been obtained in Green & Moore [1986]; Moore [1982]; Bai & Sastry [1985].

In Goodwin & Sin [1984] general definitions on persistence of excitation of a signal are given. We choose the following definition from Bai & Sastry [1985] and adopt the convention from Green & Moore [1986] of referring to regressors as satisfying PEC and inputs satisfying the SRC.

Definition 4.1.5. *A sequence of input signals, $\{\boldsymbol{u}_k\}, \in \mathcal{R}^m$ is sufficiently rich of order N, in P steps, if there exist positive integers P, ρ_1 and ρ_0 such that*

$$0 < \rho_0 I_{Nm} \leq \sum_{k=k_0+1}^{k_0+P} \begin{bmatrix} \boldsymbol{u}_{k-1} \\ \boldsymbol{u}_{k-2} \\ \vdots \\ \boldsymbol{u}_{k-N} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{k-1}^T & \boldsymbol{u}_{k-2}^T & \dots & \boldsymbol{u}_{k-N}^T \end{bmatrix} \leq \rho_1 I_{Nm} < \infty, \qquad \forall k_0. \tag{4.18}$$

This definition states that the signal $\boldsymbol{u}_k$ is uniformly strongly persistently exciting in the parlance of Goodwin & Sin [1984]. Moreover, this has an interpretation in the frequency domain. Lemma 3.4.6 of Goodwin & Sin [1984] states that a scalar ($m = 1$) quasi-stationary u_k satisfying this definition must have a two-sided spectrum which is non-zero at at least N frequencies.

We now state a definition and an important theorem of Green & Moore [1986], which permits the derivation of conditions on the input sequence $\{\boldsymbol{u}_k\}$ to ensure persistency of excitation of the regressors. This problem is also very well characterized by Anderson & Johnson Jr [1982].

Consider the minimal state space system

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \tag{4.19a}$$
$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{D}\boldsymbol{u}_k \tag{4.19b}$$

with $\boldsymbol{x}_k \in \mathcal{R}^n$, $\boldsymbol{u}_k \in \mathcal{R}^m$, $\boldsymbol{y}_k \in \mathcal{R}^p$, and the associated proper transfer function

$$\boldsymbol{T} = \boldsymbol{C}(z\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}.$$

Definition 4.1.6. *The system (4.19) is called output reachable if, for any $\bar{\boldsymbol{y}} \in \mathcal{R}^p$ and arbitrary initial state, there exists an input sequence $\{\boldsymbol{u}_i,\ i = 0, \dots, k < \infty\}$ such that its output at time k satisfies $\boldsymbol{y}_k = \bar{\boldsymbol{y}}$.*

Definition 4.1.6 is satisfied if the following Markov parameter matrix

$$\boldsymbol{M}_n = \begin{bmatrix} \boldsymbol{D} & \boldsymbol{C}\boldsymbol{B} & \boldsymbol{C}\boldsymbol{A}\boldsymbol{B} & \dots & \boldsymbol{C}\boldsymbol{A}^{n-1}\boldsymbol{B} \end{bmatrix},$$

associated with the system (4.19) has full row rank p.

Theorem 4.1.3. *[Green & Moore, Green & Moore [1986]] A necessary and sufficient condition for the output of every output reachable system (4.19) of McMillan degree n to be persistently exciting over the interval $[k+1-n,\ k+l]$ independent of initial conditions, is that the vector $\tilde{\boldsymbol{u}}_i = [\boldsymbol{u}_i^T, \boldsymbol{u}_{i-1}^T, \dots, \boldsymbol{u}_{i-n}^T]^T$ is sufficiently rich over the interval $[k+1,\ k+l]$.*

Theorem 4.1.3 simply states that a sufficiently rich (of order n in l steps) input sequence $\tilde{\boldsymbol{u}}_i$ provides a persistently exciting output sequence. Also, any system that is not output reachable cannot have a persistently exciting output sequence. When adaptive estimation is considered, the requirement is that the regressor vectors involving past inputs, past outputs and possible noise estimates are persistently exciting. The methods of Green & Moore [1986] may now be applied by defining the output reachable system to be that which has input $\tilde{\boldsymbol{u}}_k$ and output $\boldsymbol{\psi}_k$, the complete regressor. Then the output being sufficiently rich coincides with the regressors being persistently exciting if the system is output reachable. Thus, similar conditions to Theorem 4.1.3 are needed. For example, consider the multivariable ARMA model

$$\boldsymbol{y}_k + \boldsymbol{A}_1\boldsymbol{y}_{k-1} + \boldsymbol{A}_2\boldsymbol{y}_{k-2} + \cdots + \boldsymbol{A}_\ell\boldsymbol{y}_{k-\ell} = \boldsymbol{B}_1\boldsymbol{u}_{k-1} + \boldsymbol{B}_2\boldsymbol{u}_{k-2} + \cdots + \boldsymbol{B}_q\boldsymbol{u}_{k-q}$$

with transfer matrix

$$\boldsymbol{T}(z) = \boldsymbol{A}^{-1}(z)\boldsymbol{B}(z)$$

where

$$\begin{aligned}\boldsymbol{A}(z) &= \boldsymbol{I}z^\ell + \boldsymbol{A}_1 z^{\ell-1}\cdots + \boldsymbol{A}_\ell,\\ \boldsymbol{B}(z) &= \boldsymbol{B}_1 z^{\ell-1} + \cdots + \boldsymbol{B}_q z^{\ell-q}.\end{aligned}$$

Then, the associated regressor

$$\boldsymbol{\phi}_{\ell q}(k) = \begin{bmatrix}\boldsymbol{y}_{k-1}^T & \boldsymbol{y}_{k-2}^T & \cdots & \boldsymbol{y}_{k-\ell}^T & \boldsymbol{u}_{k-1}^T & \cdots & \boldsymbol{u}_{k-q}^T\end{bmatrix}^T \tag{4.20}$$

must be reachable from the input, and the input itself must be persistently exciting. To check the reachability of (4.20), the following result may be used.

Corollary 4.1.4. *[Green & Moore Green & Moore [1986]] The regressor (4.20) is reachable from $\boldsymbol{u}_k$ if and only if $[\boldsymbol{A}(z)\ \boldsymbol{B}(z)]$ is irreducible (has full row rank for all z) or, equivalently, $\boldsymbol{A}(z)$ and $\boldsymbol{B}(z)$ are left coprime.*

When, ARMAX models are considered, pseudo-linear regressor vectors are used, instead. In this case the persistence of excitation is analyzed by Moore [1982]. Briefly, consider here

$$\boldsymbol{y}_k + \boldsymbol{A}_1\boldsymbol{y}_{k-1} + \cdots + \boldsymbol{A}_\ell\boldsymbol{y}_{k-\ell} = \boldsymbol{B}_1\boldsymbol{u}_{k-1} + \cdots + \boldsymbol{B}_q\boldsymbol{u}_{k-q} + \boldsymbol{\omega}_k + \boldsymbol{C}_1\boldsymbol{\omega}_{k-1} + \cdots + \boldsymbol{C}_r\boldsymbol{\omega}_{k-r}$$

where the associated regressor is

$$\boldsymbol{\phi}_{\ell qr}(k) = \begin{bmatrix}\boldsymbol{y}_{k-1}^T & \cdots & \boldsymbol{y}_{k-\ell}^T & \boldsymbol{u}_{k-1}^T & \cdots & \boldsymbol{u}_{k-q}^T & \hat{\boldsymbol{\omega}}_{k-1}^T & \cdots & \hat{\boldsymbol{\omega}}_{k-r}^T\end{bmatrix}^T \tag{4.21}$$

with $\hat{\boldsymbol{\omega}}_{\boldsymbol{k}}$ noise estimates, generated in terms of parameter estimates

$$\hat{\boldsymbol{\theta}} = \{\hat{\boldsymbol{A}}_i,\ i = 1, \ldots, \ell;\ \hat{\boldsymbol{B}}_i,\ i = 1, \ldots, q;\ \hat{\boldsymbol{C}}_i,\ i = 1, \ldots, r\}$$

via

$$\begin{aligned}\hat{\boldsymbol{\omega}}_k &= \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k, \\ &= \boldsymbol{y}_k - \Big[-\hat{\boldsymbol{A}}_1 \boldsymbol{y}_{k-1} - \cdots - \hat{\boldsymbol{A}}_\ell \boldsymbol{y}_{k-\ell} + \hat{\boldsymbol{B}}_1 \boldsymbol{u}_{k-1} + \cdots + \hat{\boldsymbol{B}}_q \boldsymbol{u}_{k-q} + \hat{\boldsymbol{C}}_1 \hat{\boldsymbol{\omega}}_{k-1} + \ldots \\ &\quad + \hat{\boldsymbol{C}}_r \hat{\boldsymbol{\omega}}_{k-r}\Big].\end{aligned}$$

Similarly to Corollary 4.1.4, we have the following result for ARMAX models.

Corollary 4.1.5 (Green & Moore [1986])**.** *The pseudo-regressor (4.21) is reachable from $\boldsymbol{u}_k$, if and only if both $[\boldsymbol{A}(z),\ \boldsymbol{B}(z)]$, and $[\hat{\boldsymbol{C}}(z),\ \boldsymbol{A}(z)\hat{\boldsymbol{B}}(z) - \hat{\boldsymbol{A}}(z)\boldsymbol{B}(z)]$ are left coprime.*

4.2 Persistently Exciting Model Predictive Control

4.2.1 Derivation of constraints for persistence of excitation

To include the PEC into MPC, we observe that the MPC controller is a receding horizon method. That is, at the current time index k, the approach proceeds by calculating the finite-horizon, open-loop, optimal constrained future control input sequence $\{\boldsymbol{u}_k^k, \boldsymbol{u}_{k+1}^k, \ldots, \boldsymbol{u}_{k+N_p-1}^k\}$ for the horizon value N_p.

Since this depends on the current plant state value, $\boldsymbol{x}_k$, a feedback control law, $\boldsymbol{u}_k^k = \boldsymbol{\ell}(\boldsymbol{x}_k)$, is achieved from the open-loop calculation. Because the optimization (2.16) is time-invariant, this feedback law is also time-invariant if the plant model is fixed. When a PEC is included into the MPC, as the MPC solution looks forward N_p steps but implements only a single term in the solution, it is necessary to insist that the PEC considers the relationship between the applied new control signal, $\boldsymbol{u}_k^k$, and its recent past, $\{\boldsymbol{u}_{k-P-N+2}^{k-P-N+2}, \ldots, \boldsymbol{u}_{k-1}^{k-1}\}$. This does not affect the feedback nature of the control law. But it does force its dependence on more than the current plant state value. That is, PE-MPC requires that the state feedback control law has the memory of a certain number of previous inputs. Hence PE-MPC is a dynamic state feedback controller. We formulate this next.

Using Definition 4.1.5, a constraint suitable for implementation with PE-MPC may be derived. The basic idea of the sufficient richness constraint implementation is to specify an additional constraint at time k on the first solution variable $\boldsymbol{u}_k^k$ alone. Subsequently the superscripts on $\boldsymbol{u}_k$ are omitted.

Define $P \geq N$ as the backwards-looking input excitation horizon and

$$\boldsymbol{\Omega}_k = \sum_{j=0}^{P-1} \begin{bmatrix} \boldsymbol{u}_{k-j} \\ \boldsymbol{u}_{k-1-j} \\ \vdots \\ \boldsymbol{u}_{k-N+1-j} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{k-j} \\ \boldsymbol{u}_{k-1-j} \\ \vdots \\ \boldsymbol{u}_{k-N+1-j} \end{bmatrix}^T$$

then write the richness condition (4.18) as

$$\rho_1 I_{Nm} > \boldsymbol{\Omega}_k > \rho_0 \boldsymbol{I}_{Nm}. \tag{4.22}$$

The focus from here on is solely on the lower bound in (4.22), because the achievement of the upper bound is a requirement of the MPC controller achieving regulation, such as is addressed in many works on MPC using techniques such as terminal state constraints and horizon choice. We assume that this condition has been met and that the inclusion of an additional lower-bounding excitation constraint to be asserted in PE-MPC does not violate this. Our focus is on achieving uniform adaptation of plant models via persistence of excitation. Without assuming adequate bounds on the loop signals, the focus of the adaptation on model consistency is moot.

Thus (4.22) can be written in a more compact form

$$\tilde{\boldsymbol{\Omega}}_k = \sum_{j=0}^{P-1} \boldsymbol{\phi}_{k-j} \boldsymbol{\phi}_{k-j}^T - \rho_0 \boldsymbol{I}_{Nm} > 0, \tag{4.23}$$

where

$$\boldsymbol{\phi}_i = \begin{bmatrix} \boldsymbol{u}_i \\ \boldsymbol{u}_{i-1} \\ \vdots \\ \boldsymbol{u}_{i-N+1} \end{bmatrix}. \tag{4.24}$$

Further, rewritten as

$$\boldsymbol{\phi}_i = \begin{bmatrix} \boldsymbol{u}_i \\ \boldsymbol{\phi}_{i-1}^- \end{bmatrix}$$

where $\boldsymbol{\phi}_{i-1}^-$ is a $m(N-1)\times 1$ dimensional vector obtained from $\boldsymbol{\phi}_{i-1}$ by removing the last m elements, which by time k have already been determined and therefore are not subject to the current optimization. With this new notation, (4.23) becomes

$$\begin{aligned} \tilde{\boldsymbol{\Omega}}_k &= \sum_{j=0}^{P-1} \begin{bmatrix} \boldsymbol{u}_{k-j} \\ \boldsymbol{\phi}_{k-1-j}^- \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{k-j}^T & \boldsymbol{\phi}_{k-1-j}^{-T} \end{bmatrix} - \rho_0 \boldsymbol{I}_{Nm} \\ &= \begin{bmatrix} \boldsymbol{u}_k \boldsymbol{u}_k^T + \sum_{j=1}^{P-1} \boldsymbol{u}_{k-j} \boldsymbol{u}_{k-j}^T - \rho_0 \boldsymbol{I}_m & \boldsymbol{u}_k \boldsymbol{\phi}_{k-1}^{-T} + \sum_{j=1}^{P-1} \boldsymbol{u}_{k-j} \boldsymbol{\phi}_{k-1-j}^{-T} \\ \boldsymbol{\phi}_{k-1}^- \boldsymbol{u}_k^T + \sum_{j=1}^{P-1} \boldsymbol{\phi}_{k-1-j}^- \boldsymbol{u}_{k-j}^T & \sum_{j=0}^{P-1} \boldsymbol{\phi}_{k-1-j}^- \boldsymbol{\phi}_{k-1-j}^{-T} - \rho_0 \boldsymbol{I}_{m(N-1)} \end{bmatrix} > 0. \end{aligned}$$

Now applying Theorem 4.1.2 we obtain that $\tilde{\boldsymbol{\Omega}}_k > 0$ if the previous regressors fulfill

$$\tilde{\boldsymbol{\Omega}}_{k-1}^{-} = \sum_{j=0}^{P-1} \boldsymbol{\phi}_{k-1-j}^{-} \boldsymbol{\phi}_{k-1-j}^{-T} - \rho_0 \boldsymbol{I}_{m(N-1)} > 0 \tag{4.25}$$

and the Schur complement satisfies

$$\begin{aligned} \tilde{\boldsymbol{\Omega}}_k / \tilde{\boldsymbol{\Omega}}_{k-1}^{-} = \boldsymbol{u}_k \boldsymbol{u}_k^T + \sum_{j=1}^{P-1} \boldsymbol{u}_{k-j} \boldsymbol{u}_{k-j}^T - \rho_0 \boldsymbol{I}_m - \left(\boldsymbol{u}_k \boldsymbol{\phi}_{k-1}^{-T} + \sum_{j=1}^{P-1} \boldsymbol{u}_{k-j} \boldsymbol{\phi}_{k-1-j}^{-T} \right) \\ \left(\tilde{\boldsymbol{\Omega}}_{k-1}^{-} \right)^{-1} \left(\boldsymbol{\phi}_{k-1}^{-} \boldsymbol{u}_k^T + \sum_{j=1}^{P-1} \boldsymbol{\phi}_{k-1-j}^{-} \boldsymbol{u}_{k-j}^T \right) > 0. \end{aligned} \tag{4.26}$$

By noticing that $\tilde{\boldsymbol{\Omega}}_{k-1}^{-}$ is the leading principal submatrix of $\boldsymbol{\Omega}_{k-1} - \rho_0 \boldsymbol{I}_{Nm} > 0$, it is concluded that (4.25) is always satisfied if (4.23) holds at time $k-1$. Thus, after some simple manipulation of (4.26), the PE candidate constraint for the MPC is

$$\alpha \boldsymbol{u}_k \boldsymbol{u}_k^T + \boldsymbol{u}_k \boldsymbol{\beta}^T + \boldsymbol{\beta} \boldsymbol{u}_k^T + \boldsymbol{\Gamma} > 0 \tag{4.27}$$

where

$$\alpha = 1 - \boldsymbol{\phi}_{k-1}^{-T} \left(\tilde{\boldsymbol{\Omega}}_{k-1}^{-} \right)^{-1} \boldsymbol{\phi}_{k-1}^{-}, \tag{4.28a}$$

$$\boldsymbol{\beta} = - \sum_{j=1}^{P-1} \boldsymbol{u}_{k-j} \boldsymbol{\phi}_{k-1}^{-T} \left(\tilde{\boldsymbol{\Omega}}_{k-1}^{-} \right)^{-1} \boldsymbol{\phi}_{k-1}^{-}, \tag{4.28b}$$

$$\boldsymbol{\Gamma} = \sum_{j=1}^{P-1} \boldsymbol{u}_{k-j} \boldsymbol{u}_{k-j}^T - \rho_0 \boldsymbol{I}_m - \sum_{j=1}^{P-1} \boldsymbol{u}_{k-j} \boldsymbol{\phi}_{k-1-j}^{-T} \left(\tilde{\boldsymbol{\Omega}}_{k-1}^{-} \right)^{-1} \sum_{j=1}^{P-1} \boldsymbol{\phi}_{k-1-j}^{-} \boldsymbol{u}_{k-j}^T. \tag{4.28c}$$

Note that α is a scalar, $\boldsymbol{\beta}$ is an $m \times 1$ vector, and $\boldsymbol{\Gamma}$ is an $m \times m$ matrix.

Proposition 4.2.1. *The MPC candidate PE constraint (4.27) is non-convex.*

Proof. Relation (4.23) is equivalent to

$$\tilde{\boldsymbol{\Omega}}_k = \boldsymbol{\phi}_k \boldsymbol{\phi}_k^T + \boldsymbol{\Omega}_{k-1} - \boldsymbol{\phi}_{k-P} \boldsymbol{\phi}_{k-P}^T - \rho_0 \boldsymbol{I}_{Nm} > 0. \tag{4.29}$$

Thus, defining

$$\tilde{\boldsymbol{\Omega}}_{k-1}^{+} = \boldsymbol{\Omega}_{k-1} - \boldsymbol{\phi}_{k-P} \boldsymbol{\phi}_{k-P}^T - \rho_0 \boldsymbol{I}_{Nm},$$

$\tilde{\boldsymbol{\Omega}}_k$ may be written as

$$\tilde{\boldsymbol{\Omega}}_k = \boldsymbol{\phi}_k \boldsymbol{\phi}_k^T + \tilde{\boldsymbol{\Omega}}_{k-1}^{+} > 0.$$

From the matrix determinant lemma, we have that, for an invertible matrix $\boldsymbol{M}$, and two column vectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$

$$\det(\boldsymbol{M} + \boldsymbol{v}_1\boldsymbol{v}_2^T) = (1 + \boldsymbol{v}_2^T\boldsymbol{M}^{-1}\boldsymbol{v}_1)\det(\boldsymbol{M}).$$

Applying it to $\tilde{\boldsymbol{\Omega}}_k$ and assuming that $\det(\tilde{\boldsymbol{\Omega}}_{k-1}^+) \neq 0$, we obtain

$$\det(\tilde{\boldsymbol{\Omega}}_k) = \det(\tilde{\boldsymbol{\Omega}}_{k-1}^+ + \boldsymbol{\phi}_k\boldsymbol{\phi}_k^T) = (1 + \boldsymbol{\phi}_k^T(\tilde{\boldsymbol{\Omega}}_{k-1}^+)^{-1}\boldsymbol{\phi}_k)\det(\tilde{\boldsymbol{\Omega}}_{k-1}^+) > 0.$$

Now, let us distinguish three cases:

$\tilde{\boldsymbol{\Omega}}_{k-1}^+ < 0 \implies \det(\tilde{\boldsymbol{\Omega}}_{k-1}^+) < 0$, thus to have $\det(\tilde{\boldsymbol{\Omega}}_k) > 0$, we need $1 + \boldsymbol{\phi}_k^T(\tilde{\boldsymbol{\Omega}}_{k-1}^+)^{-1}\boldsymbol{\phi}_k < 0$ which implies that $\boldsymbol{u}_k$ must be outside the ellipsoid

$$\boldsymbol{\phi}_k^T(\tilde{\boldsymbol{\Omega}}_{k-1}^+)^{-1}\boldsymbol{\phi}_k > 1.$$

$\tilde{\boldsymbol{\Omega}}_{k-1}^+ > 0 \implies \det(\tilde{\boldsymbol{\Omega}}_{k-1}^+) > 0$, thus to have $\det(\tilde{\boldsymbol{\Omega}}_k) > 0$, we need $1 + \boldsymbol{\phi}_k^T(\tilde{\boldsymbol{\Omega}}_{k-1}^+)^{-1}\boldsymbol{\phi}_k > 0$ which is always satisfied. This could happen if $P > N$, and the previous input sequence excitation level is such that, adding any new input to the sequence at time k, will still conserve the minimum excitation requirement.

$\tilde{\boldsymbol{\Omega}}_{k-1}^+$ is indefinite then it will be still non-convex but in a lower dimension.

□

Remarks:

- Note that (4.27) defines the non-convex exterior of an ellipsoid.
- By using the Schur complement the PEC matrix (4.22) dimensions are reduced, from $Nm \times Nm$ to $m \times m$, (4.27), with obvious benefits on its implementation.

4.2.2 Persistently Exciting Model Predictive Control formulation

The Persistently Exciting Model Predictive Control formulation that is proposed here is obtained by augmenting the constraint set of (2.16) with the PE constraint (4.27).

Persistently Exciting Model Predictive Control

$$\min_{\bar{\boldsymbol{u}}_k} J(k, \bar{\boldsymbol{u}}) = F(\hat{\boldsymbol{x}}_{k+N_p}) + \sum_{j=0}^{N_p-1} g(\hat{\boldsymbol{x}}_{k+j}, \boldsymbol{u}_{k+j}^k) \tag{4.30a}$$

$$s.t. \quad \hat{\boldsymbol{x}}_{k+j+1} = f(\hat{\boldsymbol{x}}_{k+j}, \boldsymbol{u}_{k+j}^k, \hat{\boldsymbol{\theta}}), \quad \text{from } \boldsymbol{x}_k, \quad j = 1, \ldots, N_p - 1, \tag{4.30b}$$

$$\boldsymbol{u}_{k+j}^k \in \mathcal{U}, \qquad j = 0, \ldots, N_p - 1, \tag{4.30c}$$

$$\hat{\boldsymbol{x}}_{k+j} \in \mathcal{X}, \qquad j = 0, \ldots, N_p, \tag{4.30d}$$

$$\alpha\boldsymbol{u}_k\boldsymbol{u}_k^T + \boldsymbol{u}_k\boldsymbol{\beta}^T + \boldsymbol{\beta}\boldsymbol{u}_k^T + \boldsymbol{\Gamma} > 0, \tag{4.30e}$$

where α, $\boldsymbol{\beta}$ and $\boldsymbol{\Gamma}$ are defined in (4.28) and are functions of previous inputs only.

Note that the PE constraint (4.30e) is a quadratic function of only the first input $\boldsymbol{u}_k$. The coefficients of this constraint, α, $\boldsymbol{\beta}$, and $\boldsymbol{\Gamma}$, are computed every time step and are time-varying. The constrained optimization problem (4.30) is a standard Nonlinear Model Predictive Control (NMPC) formulation as for example defined in Mayne [2000]. The additional constraint (4.27) or (4.30e) is fundamental to preserving a minimal level of excitation for the identification and adaptation of the system. We next explore some of its properties.

Theorem 4.2.2. *Consider the PE-MPC problem (4.30) at time k with input-only constraints, i.e. $\mathcal{X} = \mathcal{R}^n$. Assume that the solutions at times $k-1$ through $k-P-N+1$ are feasible and that the resulting PE-MPC solutions yield strictly bounded plant state x_k. Then the solution $\boldsymbol{u}_k = \boldsymbol{u}_{k-P}$ at time k is feasible, although not necessarily optimal, and its repeated application for all k is feasible and yields a P-periodic solution for $\{\boldsymbol{u}_k\}$.*

Proof. At time $k-1$ a feasible solution is $\{\boldsymbol{u}_{k-1}^{k-1}, \boldsymbol{u}_k^{k-1}, \ldots, \boldsymbol{u}_{k+N_p-2}^{k-1}\}$. This indicates the input constraint (4.30c) is satisfied. Moreover, (4.30e) and the PEC (4.23), thus the constraint (4.30e) is derived and are both satisfied.

At time k, due to the time-varying nature of the constraint (4.27), it is necessary to guarantee that the solution $\{\boldsymbol{u}_k^k, \boldsymbol{u}_{k+1}^k, \ldots, \boldsymbol{u}_{k+N_p-1}^k\}$ is still feasible.

Recalling that (4.30e) is equivalent to (4.29) and considering that $\boldsymbol{\Omega}_{k-1} - \rho_0 \boldsymbol{I}_{Nm} > 0$, since it is the PEC at time $k-1$, then (4.30e) (or equivalently (4.29)) is satisfied if $\boldsymbol{\phi}_k \boldsymbol{\phi}_k^T - \boldsymbol{\phi}_{k-P} \boldsymbol{\phi}_{k-P}^T \geq 0$.

Recalling (4.24), and that the constraint (4.30e) affects only the first input $\boldsymbol{u}_k$, it is clear that choosing $\boldsymbol{u}_k = \boldsymbol{u}_{k-P}$ in $\boldsymbol{\phi}_k$ results in a feasible solution at time k. Obviously, $\boldsymbol{u}_k = \boldsymbol{u}_{k-P}$ also satisfies the input constraint.

Finally, note that due to the receding horizon property, only the first input $\boldsymbol{u}_k$ of the sequence $\{\boldsymbol{u}_k^k, \boldsymbol{u}_{k+1}^k, \ldots, \boldsymbol{u}_{k+N_p-1}^k\}$ is applied. This, would naturally yield a P-periodic input sequence if exactly this input was repeatedly chosen.

□

Corollary 4.2.3. *Consider the PE-MPC problem of Theorem 4.2.2 with feasible periodic input $u_k = u_{k-P}$ in the scalar ($m = 1$) case, then this input sequence is the sum of at least N complex sinusoids with non-zero amplitudes and different frequencies.*

Proof. Since $\{u_k\}$ is feasible and periodic, the regressor vector $\boldsymbol{\phi}_k$ is also periodic and has associated $N \times N$ matrix $\boldsymbol{\Omega}_k$ from (4.22) fixed and positive definite. The result follows by appealing to the frequency domain interpretation of the SRC (4.18) from Goodwin & Sin [1984].

□

Remarks:

- The dual adaptive control effect is evident in the PE-MPC formulation. While the standard MPC seeks to drive the input $\boldsymbol{u}_k$ to zero, stabilizing the plant, the PE constraint (4.27) tries to bound $\boldsymbol{u}_k$ away from zero. Indeed, the cost of persistence of excitation is seen explicitly in the inclusion of an additional constraint (4.30e).
- The periodic solution $u_k = u_{k-P}$ is feasible, but not necessarily optimal.
- In the case where state constraints are considered, their satisfaction would need to be separately ensured to establish feasibility of the PE-MPC problem.
- In time-invariant situations where the MPC is able to stabilize the system to the strict interior of the state-constraint set, $\mathcal{X}$, and where the requisite excitation level, ρ_0, is correspondingly modest, one would presume that the periodic solution would be the asymptotic limit of the control signal. This has yet to be proven. Although, this is the case in the following example.
- The periodic solution arises directly from the PEC formulation. Thus, there is no need to force it by an equality constraint as done in Genceli & Nikolaou [1996].

4.3 Examples using Finite Impulse Response models

This and subsequent sections give several examples to show how the PE-MPC can inherently produce persistently exciting inputs and tend towards periodic solutions. Some interesting properties of the proposed approach are discussed.

In the first example a plant is regulated by a constrained predictive controller based on an FIR model and an RLS algorithm is used to continually estimate the Markov parameters of the plant. The coefficients of the FIR model used in the MPC are updated every 50 time steps.

The plant is itself described by an FIR model and so is open-loop stable;

$$y_k = \boldsymbol{\phi}_k \boldsymbol{\theta}^T + v_k,$$

where $\{v_k\}$ is a Gaussian measurement noise sequence with covariance $\sigma_v = 6.25 \cdot 10^{-4}$, $N = 3$, the regressor is $\boldsymbol{\phi}_k = \begin{bmatrix} u_k & u_{k-1} & u_{k-2} \end{bmatrix}^T$, and

$$\boldsymbol{\theta} = \begin{bmatrix} 0.03 \\ 0.28 \\ 0.63 \end{bmatrix}$$

are the 'true' system parameters.

In this case, for a scalar input, starting from the FIR based MPC formulation (2.13),

the PE-MPC formulation becomes

$$\min_{\bar{u}} J_k = \frac{1}{2} \sum_{j=0}^{N_p-1} \| y_{k+j+1} \|_Q^2 + \| u_{k+j} \|_R^2, \tag{4.31a}$$

$$s.t. \quad y_{k+j} = \boldsymbol{\phi}_{k+j} \hat{\boldsymbol{\theta}}_k^T, \qquad j = 1, \ldots, N_p, \tag{4.31b}$$

$$u_{\min} \leq u_{k+j} \leq u_{\max}, \quad j = 0, \ldots, N_p - 1, \tag{4.31c}$$

$$\alpha u_k^2 + 2\beta u_k + \gamma > 0. \tag{4.31d}$$

with prediction horizon $N_p = 10$, input constraints $-5 \leq u \leq 5$, cost function output weight $Q = 5$, and input weight $R = 0.3$. The time varying scalars (α, β, γ), from the PE constraint (4.31d), are computed every time step as in (4.28), and the backwards-looking excitation horizon length is chosen as $P = 7$ which exceeds $N = 3$, and the SRC design parameter is selected as $\rho_0 = 2.5$.

Decomposing the quadratic constraint (4.31d) into two linear constraints, it is possible to formulate the FIR model MPC as two Quadratic Programming (QP) problems. Thus, the optimal solution is obtained by choosing the better solution between the two QP problems.

A standard RLS algorithm without the forgetting factor, and initial conditions

$$\hat{\boldsymbol{\theta}}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \boldsymbol{P}_0 = \begin{bmatrix} 1000 & 0 & 0 \\ 0 & 1000 & 0 \\ 0 & 0 & 1000 \end{bmatrix},$$

is used to adapt the model parameters. The estimation algorithm is not reset but the MPC has its model updated every 50 samples.

In the second example, to allow set-point changes, the cost function (4.31a) is modified as follows

$$J_k = \frac{1}{2} \sum_{j=0}^{N_p-1} \| y_{k+j+1} - y_s \|_Q^2 + \| \Delta u_{k+j} \|_R^2, \tag{4.32}$$

where y_s is the set-point and $\Delta u_j = u_j - u_{j-1}$.

Finally in both cases the MPC model is initialized with the following FIR coefficients

$$\boldsymbol{\theta}_0 = \begin{bmatrix} 0.13 \\ 0 \\ 1.26 \end{bmatrix}.$$

4.3.1 Simulation results

The simulations are run in MATLAB 7.9 using the solver *e04nf* from the NAG library to solve the QP problems every time step. For every simulation, in the first $P + 1$ time steps standard FIR MPC (without PE constraint) is used and it is also checked that all the necessary vectors for the PE constraint formulation are properly initialized.

Regulation to zero

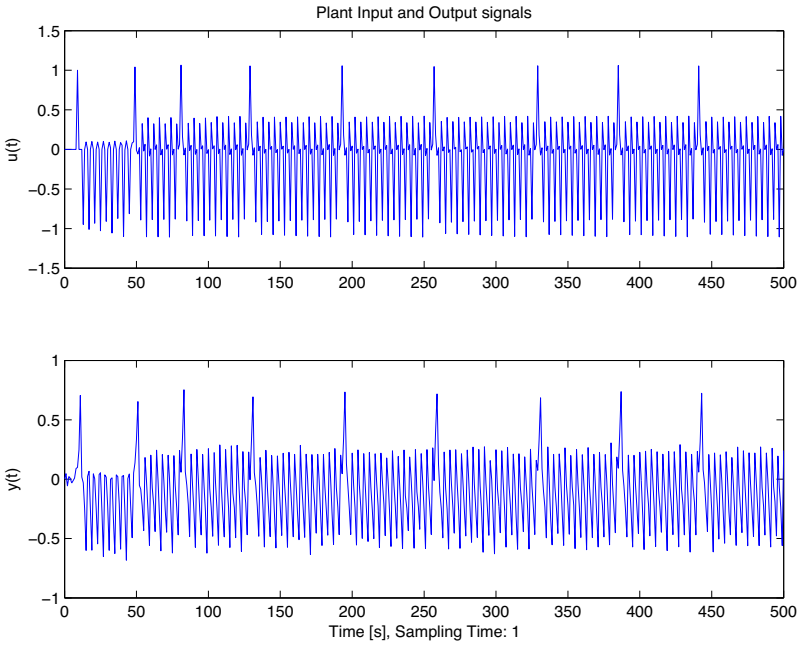


Figure 4.1: Plant input and output signals.

Figure 4.1 shows the plant input and output. Input magnitude constraints are never active and the input signal is effectively driven entirely by the SRC/PEC constraint. In Figure 4.2 the FIR coefficients are shown, the dashed lines represent the real parameter values $\boldsymbol{\theta}_0$, the solid lines are the RLS estimates, and the stars are the occasionally updated values used in the MPC model. For example, the stars at time step 50 are the FIR coefficients $\hat{\boldsymbol{\theta}}$, they are used in the controller for the first 50 steps, then the estimate from RLS is used to update the MPC model, which is again kept constant until the next update occurs (50 steps later).

Finally, Figure 4.3 shows the plant input in the time domain, and more importantly

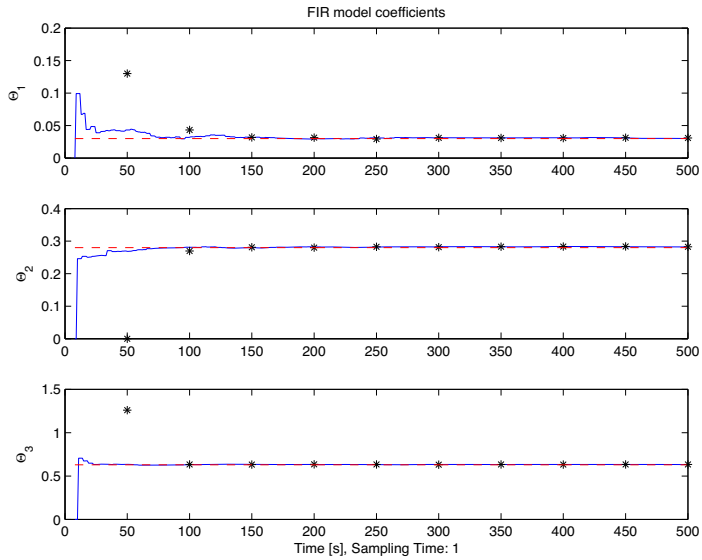


Figure 4.2: FIR parameters: RLS estimates (solid line), 'real' values (dashed line), MPC model parameter (stars).

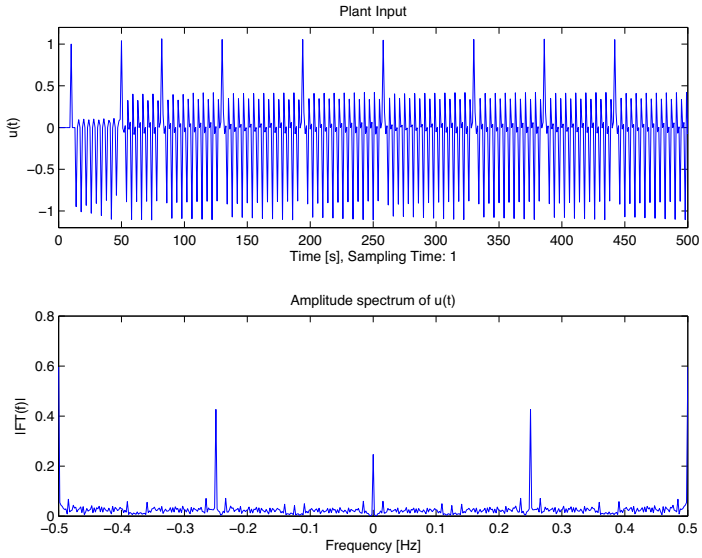


Figure 4.3: Persistently exciting input and its spectrum indicating excitation suited to the fitting of a three-parameter model.

the magnitude of its Fourier transform. The three peaks confirm that the input is persistently exciting of order three, which is the minimum order needed to estimate three parameters. This figure also confirms that u_k is tending to become periodic, as expected.

Variable output set-point

A more complex scenario is shown here, where at steps $k = 75$ and $k = 200$ set-point changes are given. In addition, at step $k = 125$ a plant change is simulated with the new plant coefficients being

$$\boldsymbol{\theta} = \begin{bmatrix} 0.5 \\ 0.2 \\ 0.3 \end{bmatrix}.$$

The simulation parameters are the same as in the previous case with the exception of the cost function output weight $Q = 1$, the weight on the input difference (Δu) $R = 3$, the backwards-looking excitation horizon length $P = 4$, and the SRC design parameter $\rho_0 = 4.9$.

Figure 4.4 shows plant input and output signals. It is evident that the PE-MPC controller is able to follow the output set-point (red-dashed line). The oscillation along the set-point is due to the effect of the PE-constraint, that is expected and needed to produce an SRC signal for estimating the three FIR parameters. As result in Figure 4.5 the estimated parameters converge to the true values, even when at step $k = 125$ a change in the plant occurs. Note that at step $k = 150$ the speed of converge suddenly increases, this is due to the covariance resetting into the RLS estimate algorithm that is performed every time the MPC model is updated. To confirm that the input signal is sufficiently rich, the amplitude spectrum is plotted in Figure 4.6.

Finally, Figure 4.7 shows an instance of how PE-MPC is divided into two QP problems, and how the PE constraints are positioned with respect to the bound constraints on the input. It is possible to note the dual effect introduced by the PE constraints. While the standard MPC would try to drive the input to zero, the new constraint bounds the input away from zero. This produces a small performance degradation, but it is necessary to obtain the dual control feature such that the controller is able to produce sufficient excitation to learn from the plant (estimate its parameters). The PE-MPC optimal solution is chosen between the two QP optimal solutions, and when one is not feasible it is chosen from the remaining one. Note that at no point both QPs are infeasible. If both QPs are infeasible at the same time, this would indicate a too high value of the SRC design parameter ρ_0.

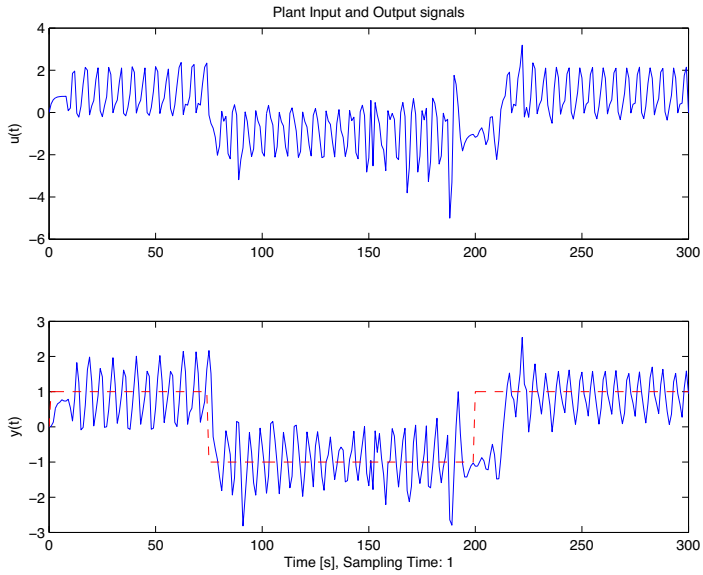


Figure 4.4: Plant input and output signals. The dashed line indicates the set-point.

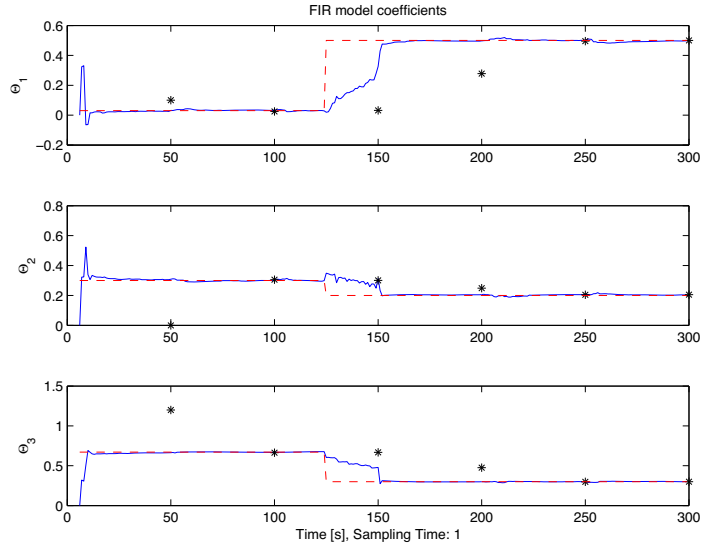


Figure 4.5: FIR parameters: RLS estimates (solid line), 'real' values (dashed line), MPC model parameter (stars).

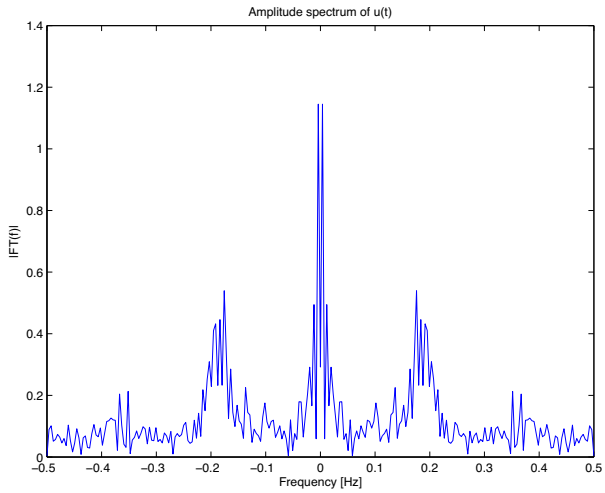


Figure 4.6: Spectrum of the input indicating excitation suited to the fitting of a three-parameter model.

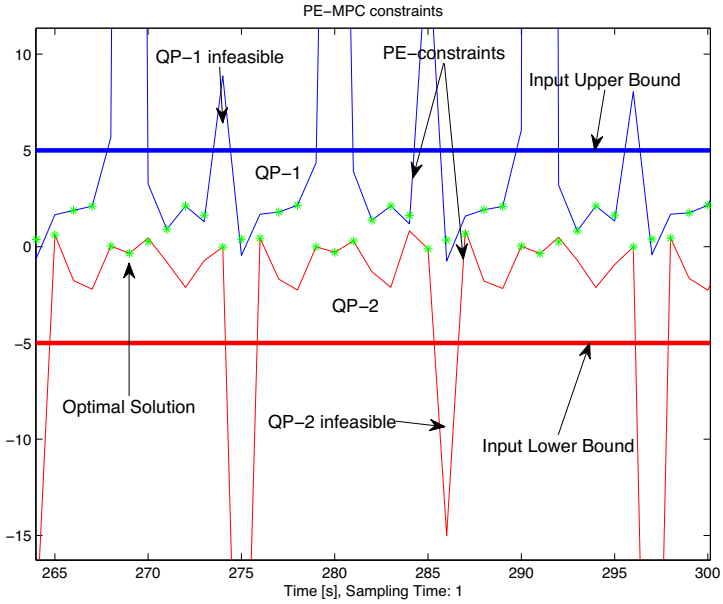


Figure 4.7: Representation of PE-MPC constraints and optimal solution for a short simulation interval.

Next, it is shown how the choice of certain design parameters such as the backwards-looking excitation horizon P and the SRC design parameter ρ_0 affects the excitation level and thus the speed of convergence of the FIR estimates. Figures 4.8 and 4.9 are obtained with a zero set-point, $\rho_0 = 2.5$, $P = 3$, and are used as base for comparison. Note that at $k = 55$ a plant change occurs. When a longer excitation horizon $P = 5$ is used, the input changes shape (Figure 4.10) and this has a direct effect by slowing the estimate speed of convergence as shown in Figure 4.11. When instead a smaller $\rho_0 = 1.8$ is chosen (with $P = 3$), the input keeps almost the same shape (Figure 4.12), but with a reduced excitation level the FIR parameter estimates converge slowly (Figure 4.13).

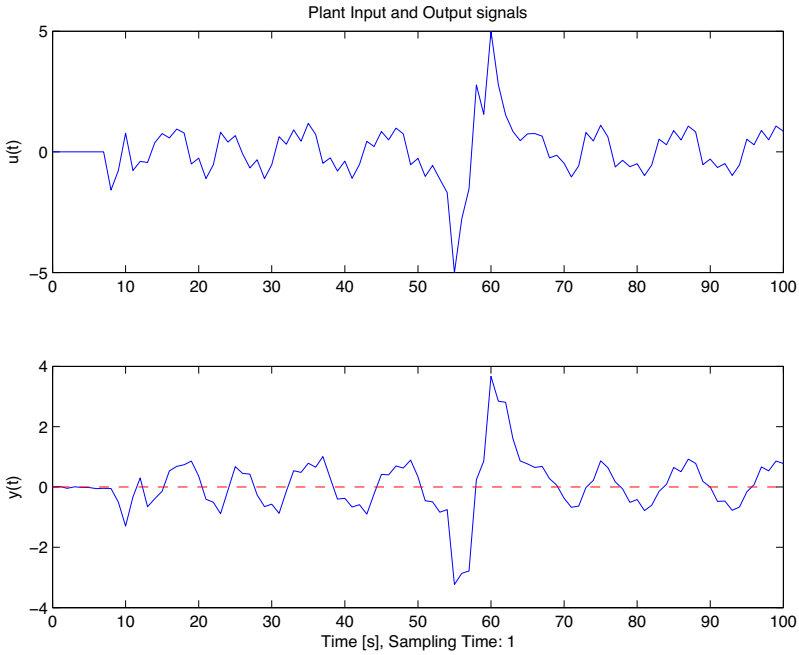


Figure 4.8: Plant input and output signals, base example for comparison. The dashed line indicates the set-point.

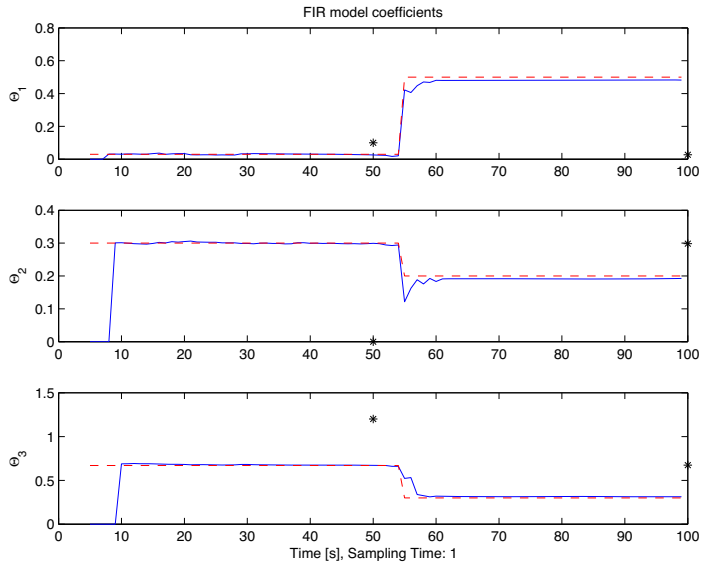


Figure 4.9: FIR parameters, base example for comparison: RLS estimates (solid line), 'real' values (dashed line), MPC model parameter (stars).

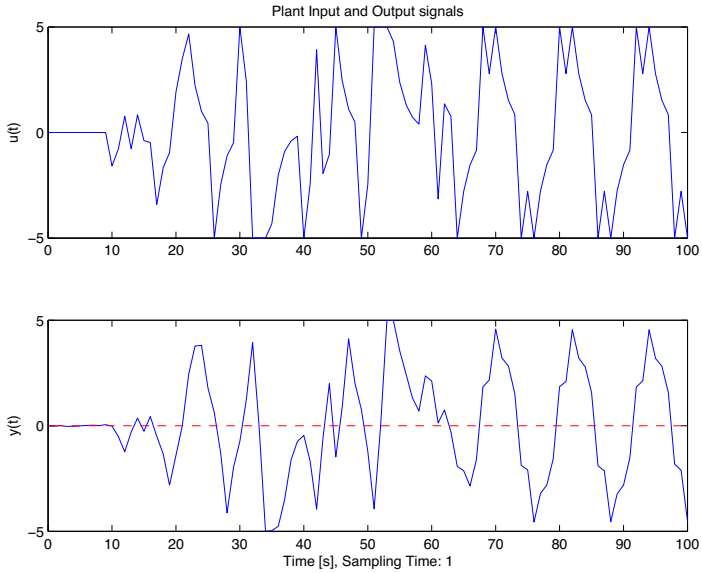


Figure 4.10: Plant input and output signals when using longer excitation horizon P. The dashed line indicates the set-point.

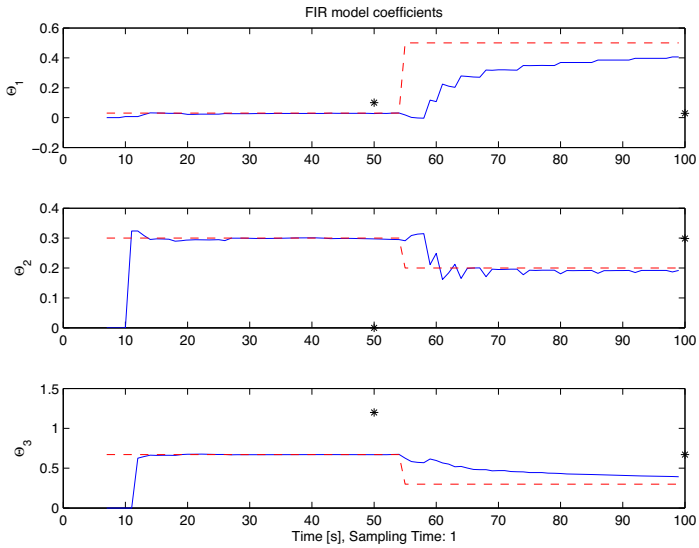


Figure 4.11: FIR parameters when using longer excitation horizon P: RLS estimates (solid line), 'real' values (dashed line), MPC model parameter (stars).

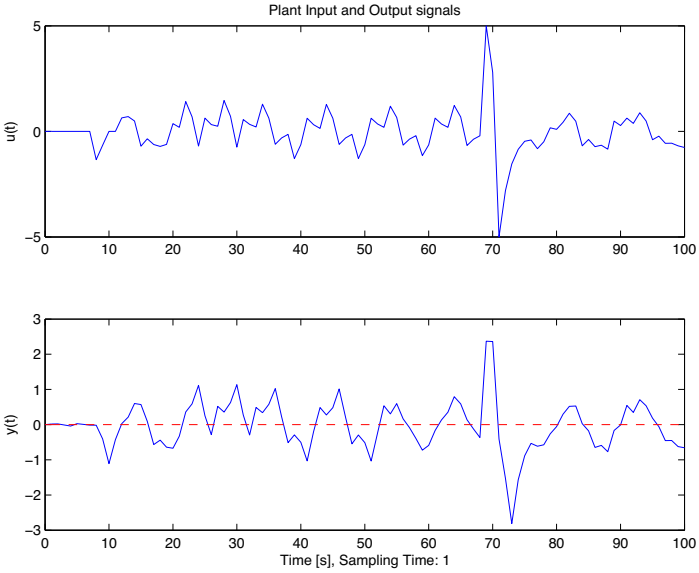


Figure 4.12: Plant input and output signals when using smaller SRC design parameter ρ_0. The dashed line indicates the set-point.

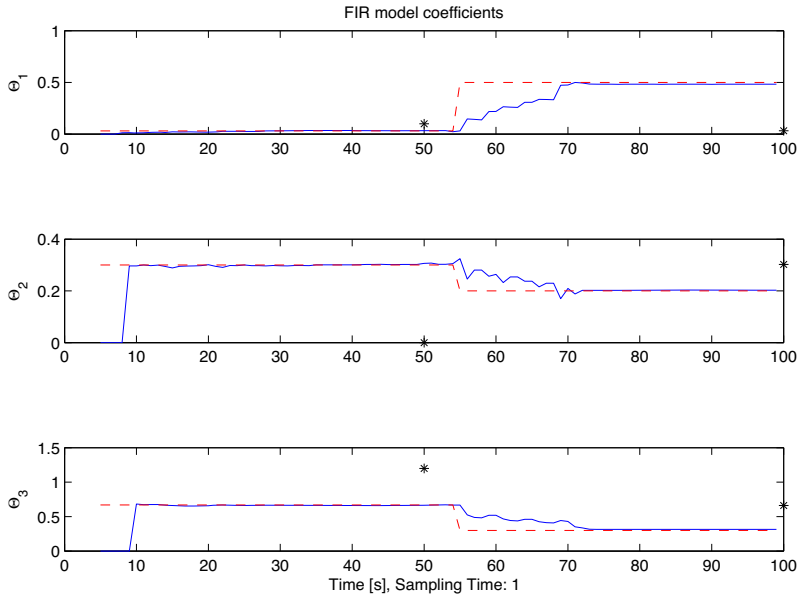


Figure 4.13: FIR parameters when using smaller SRC design parameter ρ_0: RLS estimates (solid line), 'real' values (dashed line), MPC model parameter (stars).

4.4 Examples using state space models

This section describes two examples of an PE-MPC controller based on state space models. The plant is itself described by the following state space model

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k \\ y_k &= x_k + v_k, \end{aligned}$$

with $A = 1$ and $B = 0.5$, and where $\{w_k\}$, and $\{v_k\}$ are Gaussian process and measurement noise sequences with zero means and covariances $\sigma_w = 1{\cdot}10^{-4}$ and $\sigma_v = 6.25{\cdot}10^{-4}$, respectively.

The PE-MPC formulation is

$$\min_{\bar{u}} J_k = \frac{1}{2} \sum_{j=0}^{N_p-1} \parallel y_s - y_{k+j+1} \parallel_Q^2 + \parallel \Delta u_{k+j} \parallel_R^2, \tag{4.33a}$$

$$s.t. \quad x_{k+1+j} = \hat{A}x_{k+j} + \hat{B}u_{k+j}, \qquad j = 1, \ldots, N_p, \tag{4.33b}$$

$$u_{\min} \leq u_{k+j} \leq u_{\max}, \quad j = 0, \ldots, N_p - 1, \tag{4.33c}$$

$$\alpha u_k^2 + 2\beta u_k + \gamma > 0. \tag{4.33d}$$

with $\Delta u_j = u_j - u_{j-1}$, prediction horizon $N_p = 10$, input constraints $-0.1 \leq u \leq 0.1$, cost function output weight $Q = 0.1$, and weight on the input changes $R = 1000$. The time varying scalars (α, β, γ), from the PE constraint (4.33d), are computed every time step as in (4.28), and the backwards-looking excitation horizon length is chosen as $P = 18$ which exceeds $N = 2$, finally the SRC design parameter is selected as $\rho_0 = 0.01$.

An RLS algorithm is implemented to estimate the parameter of the following ARMA model

$$y_k = ay_{k-1} + bu_{k-1}$$

where

$$\hat{\boldsymbol{\theta}}_k = \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} -\hat{A} \\ \hat{B} \end{bmatrix}, \tag{4.34}$$

and the associated regressor is

$$\boldsymbol{\phi}(k) = \begin{bmatrix} y_{k-1} & u_{k-1} \end{bmatrix}^T. \tag{4.35}$$

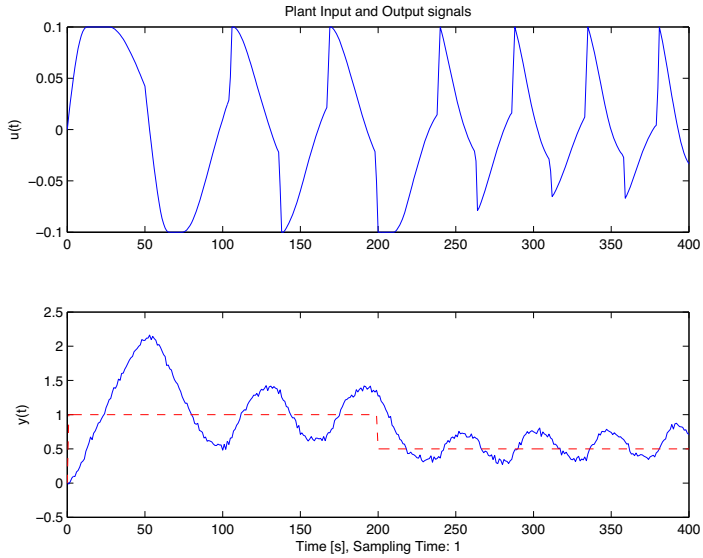
It is important to note that the regressor contains past inputs and outputs, thus Corollary 4.1.4 must be verified. In this case this is done trivially, thus a persistently exciting output may be obtained by a sufficiently rich input.

4.4.1 Simulation results

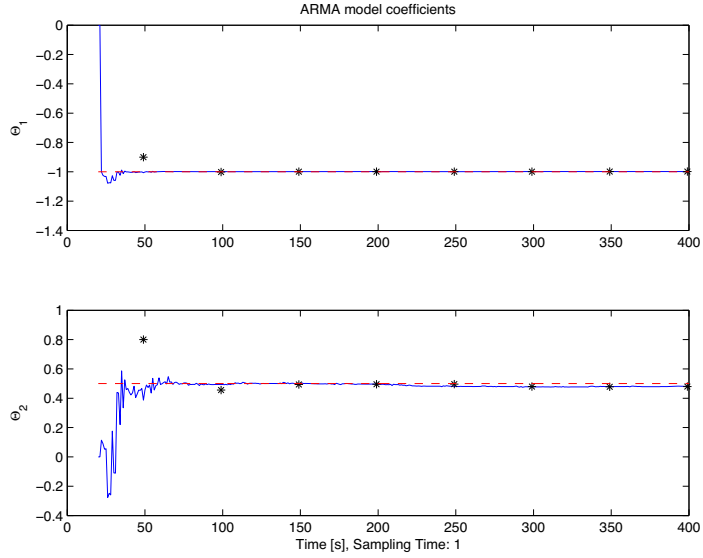
The first simulation (Figure 4.14(a)) shows that starting from zero initial condition, the output is driven to $y_s = 1$ and then at $k = 200$ a set-point change occurs. The model parameter convergence and MPC model parameter updates are shown in Figure 4.14(b).

It is important to note in Figure 4.15 that due to the difference between the zero initial condition and the output set-point $y_s = 1$, both PE constraints are not active. In fact only one QP problem (green color in Figure 4.15(b)) is solved every time step for the first part. Then when the actual plant output gets close to the set-point the excitation level diminishes making the PE constraints active and yielding two distinct QP problems (blue and red Figure 4.15(b)). This is interesting behavior because it shows that when, for some external reason, the input is already sufficiently rich the PE constraints are not active and thus only one QP problem (corresponding to a standard MPC implementation) has to be solved.

While in the first example full state feedback was assumed, in the next example a standard linear Kalman Filter is used for state estimation. The steady state Kalman gain is re-calculated every time the model is updated (each 50 steps). Figures 4.16(a) and 4.16(b) show input, output and model parameters, respectively. This example illustrates that starting from zero steady state, once the PE-MPC is activated, it drives the plant away from the steady state, due to the lack of excitation. At step $k = 200$ a sinusoidal set-point is applied which makes both PE constraints become inactive as given in Figure 4.17. Finally, Figure 4.18(a) shows that input and output have two peaks respectively on their amplitude spectrum, which confirms the minimum frequency content for estimating $N = 2$ parameters. Figure 4.18(b) compares the Kalman state estimate with the real state.

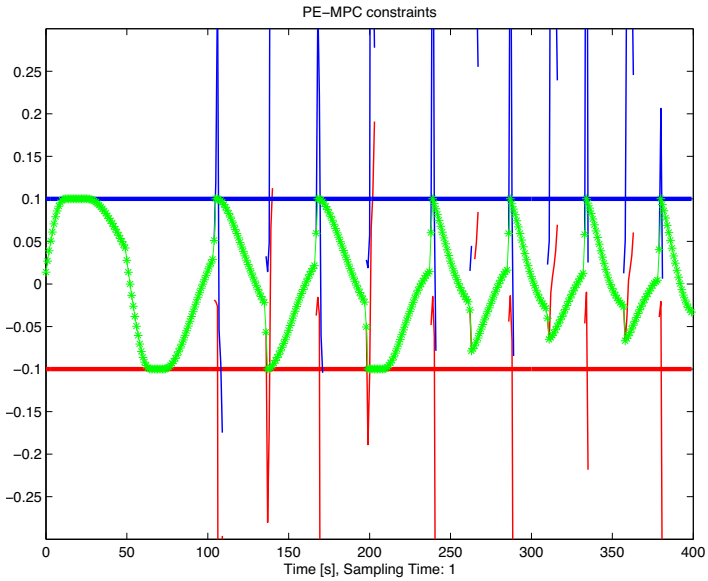


(a) Plant input and output signals. The dashed line indicates the set-point.

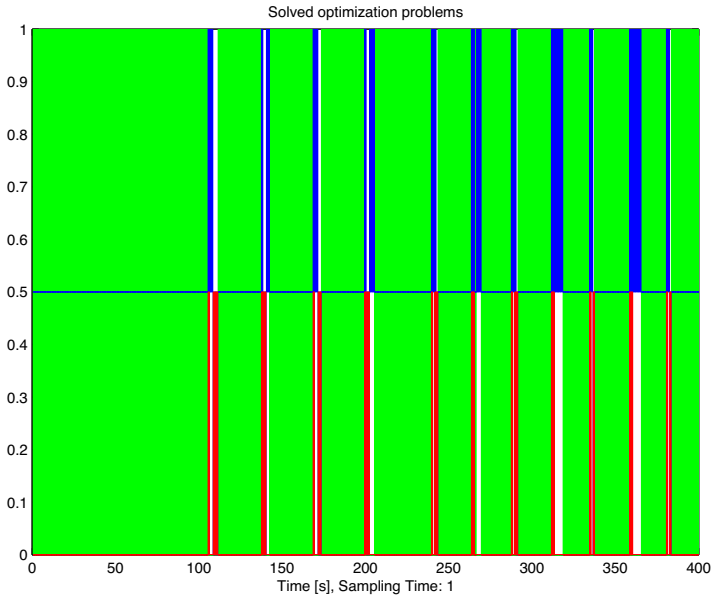


(b) ARMA parameters: RLS estimates (solid line), 'real' values (dashed line), MPC model parameter (stars).

Figure 4.14: Input, output and parameters for state space case.

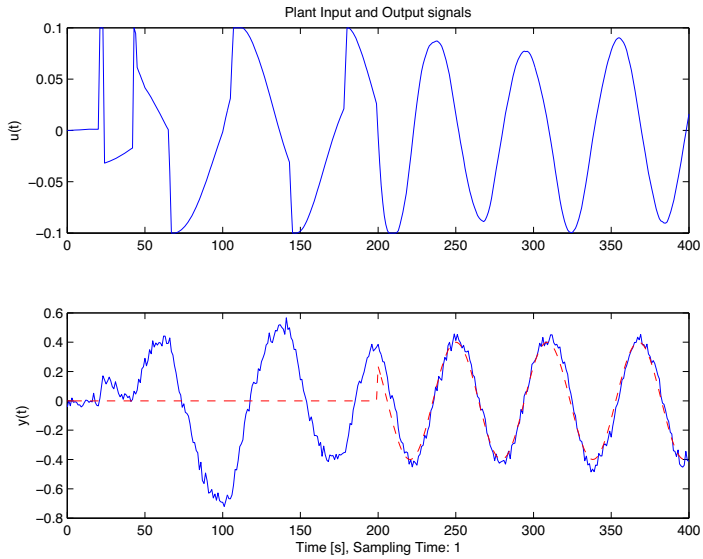


(a) Representation of PE-MPC constraints and optimal solution.

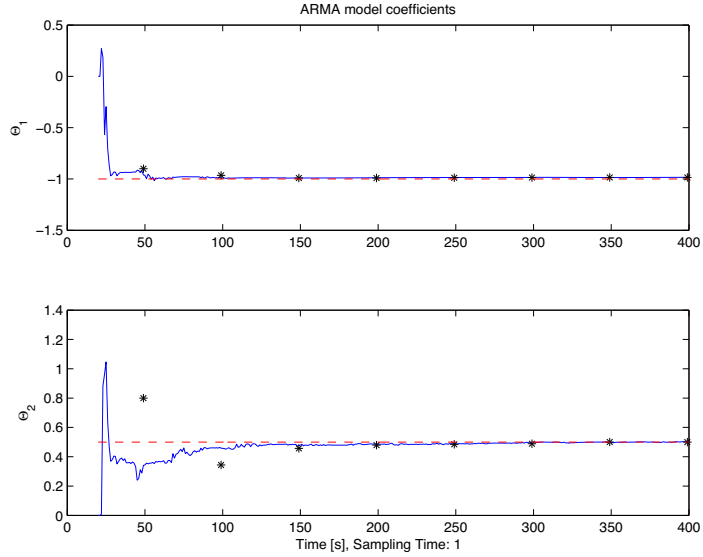


(b) Solved QP problems: QP with no PE constraints (green), QP with upper PE constraint (blue), QP with lower PE constraint (red). Note that the color indicates which QP is solved.

Figure 4.15: PE-MPC constraints and solved QP problems for state space case.

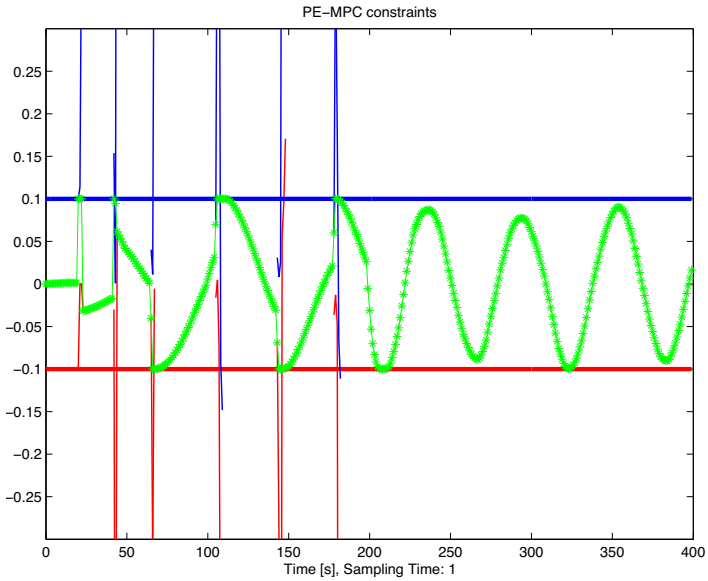


(a) Plant input and output signals. The dashed line indicates the set-point.

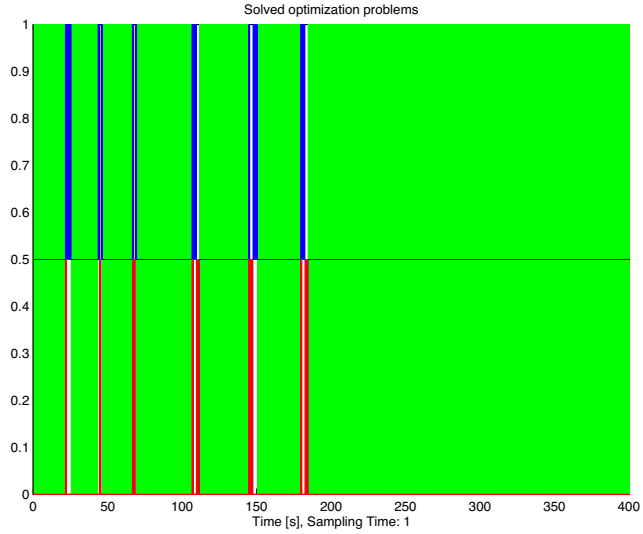


(b) ARMA parameters: RLS estimates (solid line), 'real' values (dashed line), MPC model parameter (stars).

Figure 4.16: Input, output and parameters for state space case with Kalman Filter state estimate.

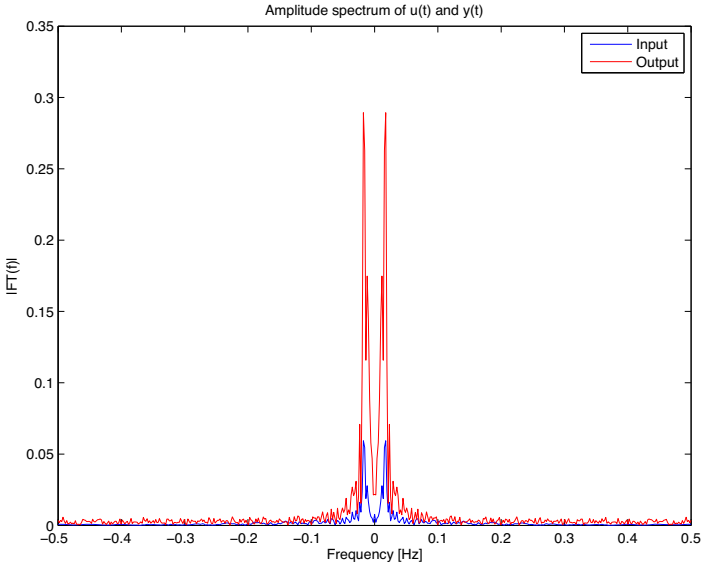


(a) Representation of PE-MPC constraints and optimal solution.

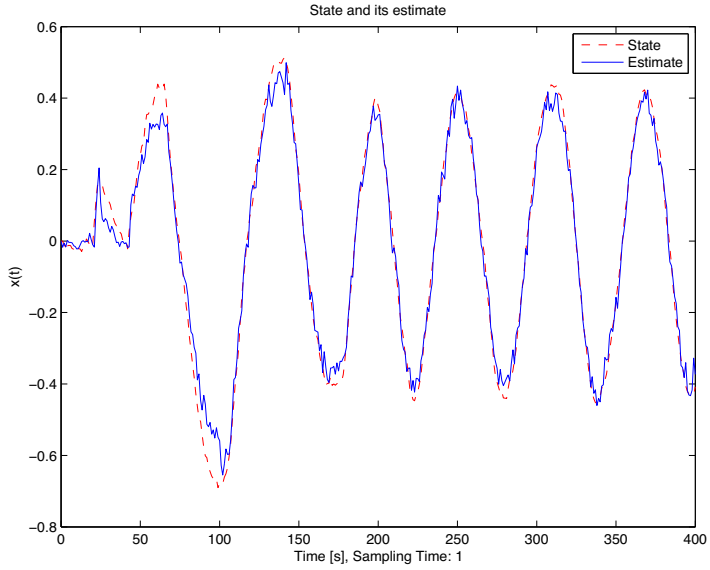


(b) Solved QP problems: QP with no PE constraints (green), QP with upper PE constraint (blue), QP with lower PE constraint (red). Note that the color indicates which QP is solved.

Figure 4.17: PE-MPC constraints and solved QP problems for state space case with Kalman Filter state estimate.



(a) Input and output spectrum.



(b) State and its estimate.

Figure 4.18: Input and output spectrum, and state and its estimate for state space case with Kalman Filter state estimate.

4.5 Conclusions

A simple modification of the model predictive control formulation is given which yields persistently exciting regressor vectors suited to online model adaptation or tuning. Its effectiveness is shown by several computational examples. The approach chosen is similar to the one in Genceli & Nikolaou [1996], but with the difference that the PE constraint is imposed only on the first MPC manipulated variable and via a backwards-looking additional constraint. This alters the MPC controller from memoryless full-state feedback to dynamic full-state feedback. This novel framework allowed us to show formally and by examples that a periodic control signal is feasible and arises autonomously, avoiding the necessity of explicitly forcing this by adding an equality constraint. Several simulation examples, for FIR and state space models, with full state and also output feedback are given. In all examples it is possible to note that the particular structure of the PE constraint with scalar inputs gives PE-MPCs that are expressed as two QP problems. In general, the PE-MPC formulation given is valid for MIMO systems and for general adaptive schemes, yielding a non-convex optimization problem. In this case general non-convex solvers may be required or, alternatively, the non-convex SRC constraint (4.30e) might be broken down into the union of a number of convex constraints. Efficient numerical implementation for MIMO systems will be a topic for future research.

Chapter 5

State estimation issues

5.1 The state estimation problem

The state estimation problem can generally be seen as a probabilistic inference problem, which means how to estimate hidden variables (state) using a noisy set of observations (measurement) in an optimal way. In probability theory, this solution is obtained using the recursive Bayesian estimation algorithm. For linear Gaussian systems, it can be shown that the closed-form of the optimal recursive solution is given by the well known Kalman Filter.

For real world applications, which are neither linear nor Gaussian, the Bayesian approach is intractable. Thus, a sub-optimal approximated solution has to be found. For instance, in the case of a nonlinear system with non-Gaussian probability density function (pdf), the Extended Kalman Filter (EKF) algorithm approximates a nonlinear system with its truncated Taylor series expansion around the current estimate. Here, the non-Gaussian pdf is approximated with its first two moments, mean and covariance. A more elaborate approach is to use a Sequential Monte-Carlo method, where instead of approximating the system or the probability distribution, the integrals in the Bayesian solution are approximated by finite sums. Although a famous Monte-Carlo filter is the Particle Filter, in this chapter the focus is on Kalman filtering for two simple reasons. Firstly, Kalman algorithms are easy to understand and to implement. Secondly, they do not have high computational complexity, making them suitable to be coupled with a model predictive controller. This is a benefit for real-time implementation. Since all computation must be done within a maximum time interval, having a quick estimation algorithm allows the use of the remaining time to be used for solving the control optimization problem.

The focus is mainly on Kalman filtering for nonlinear systems. In detail, the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF) are described and their performances are compared in two specific cases.

5.1.1 Definitions of Observability

When state estimation is considered, it is very important to understand the concept of the observability of a system. Observability is used to determine whether the internal state of a system can be inferred by knowledge about its external output. A formal definition of observability is the following [Kailtah, 1980].

Definition 5.1.1. Observability:
A system is said to be observable if for any possible sequence of state and control vectors, the current state can be reconstructed in finite time as a function of only the past output sequence.

Although Definition 5.1.1 is very clear, it has the drawback that is not straightforward to check. Therefore, a more practical mathematical tool or condition is needed to determine whether or not it is possible to estimate the state of a system.

Linear systems

Consider the SISO discrete linear system

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}u_k \\ y_k &= \boldsymbol{C}\boldsymbol{x}_k \end{aligned} \tag{5.1}$$

where $\boldsymbol{x}_k \in \mathcal{R}^n$, $u_k \in \mathcal{R}$, $y_k \in \mathcal{R}$, and n is a positive scalar.

Given the linear system (5.1), define the following observability matrix

$$\mathcal{O} = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{C}\boldsymbol{A} \\ \boldsymbol{C}\boldsymbol{A}^2 \\ \vdots \\ \boldsymbol{C}\boldsymbol{A}^{n-1} \end{bmatrix}. \tag{5.2}$$

As shown in Kailtah [1980], or any other reference on linear system theory, using (5.2) the following definition is given.

Definition 5.1.2. Observable system:
A time-invariant linear system in the state space representation (5.1) is observable if the rank of the observability matrix (5.2) is equal to n, where n is the state dimension.

Definition 5.1.2 is straightforward to check, and this may be considered the first step to do when state estimation has to be implemented for a linear system.

Nonlinear systems

The determination of observability for nonlinear systems is more complicated than the one for linear systems. The principal reason is that, for nonlinear systems, the observability depends on the system input itself. In Hermann & Krener [1977] the property of observability for nonlinear systems is discussed extensively. While for linear systems the observability is a unique and global property, for nonlinear systems this is not. In fact, there are four different forms of observability.

Given the general continuous time nonlinear system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \tag{5.3a}$$

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t)) \tag{5.3b}$$

where $\boldsymbol{x} \in \mathcal{R}^n$, $\boldsymbol{u} \in \mathcal{R}^m$, and $\boldsymbol{y} \in \mathcal{R}^p$, the system may be

$$\begin{array}{ccc} \text{Locally observable} & \Rightarrow & \text{Observable} \\ \Downarrow & & \Downarrow \\ \text{Locally weakly observable} & \Rightarrow & \text{Weakly observable} \end{array}$$

where the arrows show their relationships.

Rather than stating all four observability definitions (they can be easily found in Hermann & Krener [1977]) in this thesis the focus is only on the 'locally weakly observability' property. This is mainly due to the fact that the locally weakly observability attribute is the only one which is possible to test in practice.

Thus, for the system (5.3), let us define an open set $\mathcal{U}$ contained in $\mathcal{R}^n$, and then define the concept of '$\mathcal{U}$-Indistinguishability'.

Definition 5.1.3 (Hermann & Krener [1977])**. $\mathcal{U}$-Indistinguishability:**
A pair of points $\boldsymbol{x}^0$ and $\boldsymbol{x}^1$, contained in $\mathcal{U}$ are called $\mathcal{U}$-Indistinguishable if, for $i = 0, 1$, we have that solutions $\boldsymbol{x}^i(t)$ of (5.3a) with respect to initial conditions $\boldsymbol{x}^i(0) = \boldsymbol{x}^i$ are identical, for every admissible control $\boldsymbol{u}(t)$ defined in the interval $[0, T]$.

Note that we denote all points $\boldsymbol{x}^1 \in \mathcal{U}$ that are $\mathcal{U}$-Indistinguishable from $\boldsymbol{x}^0$ by $I(\boldsymbol{x}^0, \mathcal{U})$.

Definition 5.1.4 (Hermann & Krener [1977])**.** *The system (5.3) is* **locally weakly observable at $\boldsymbol{x}_0$** *if there exists an open neighborhood $\mathcal{U}$ of $\boldsymbol{x}^0$, such that for every open neighborhood $\mathcal{V}$ of $\boldsymbol{x}^0$ contained in $\mathcal{U}$*

$$I(\boldsymbol{x}^0, \mathcal{V}) = \boldsymbol{x}^0 \tag{5.4}$$

Moreover

Definition 5.1.5 (Hermann & Krener [1977])**.** *If $\mathcal{U}$ coincides with $\mathcal{R}^n$, and Definition 5.1.4 is valid for every $\boldsymbol{x} \in \mathcal{R}^n$, we say that the system (5.3) is* **locally weakly observable**.

Locally weakly observability can be easily tested using a rank condition [Hermann & Krener, 1977; Besançon, 1999] similar to the one for linear systems.

For system (5.3) the following observability matrix

$$\mathcal{O}(\boldsymbol{x}) = \begin{bmatrix} dh(\boldsymbol{x}) \\ d(L_f h(\boldsymbol{x})) \\ d(L_f^2 h(\boldsymbol{x})) \\ \vdots \\ d(L_f^{n-1} h(\boldsymbol{x})) \end{bmatrix} \tag{5.5}$$

is defined, where the differential of h is defined as

$$dh = \frac{\partial h}{\partial \boldsymbol{x}} = \left[\frac{\partial h}{\partial x_1}, \ldots, \frac{\partial h}{\partial x_1} \right], \tag{5.6}$$

and for a constant input $\bar{\boldsymbol{u}}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}, \bar{\boldsymbol{u}}) \tag{5.7}$$

defines the Lie derivative [Khalil, 2002] with the following property

$$L_f^i h(\boldsymbol{x}) = L_f L_f^{i-1} h(\boldsymbol{x}) = \frac{\partial (L_f^{i-1})}{\partial \boldsymbol{x}} f(\boldsymbol{x}, \bar{\boldsymbol{u}}) \tag{5.8}$$

for $i = 1, \ldots, n-1$.

Note that (5.5) is state dependent and has dimension $n \times n$. The following are important theorems for testing the locally weakly observability.

Theorem 5.1.1 (Hermann & Krener [1977])**.** *If the system (5.3) satisfies the rank condition on (5.5) at $\boldsymbol{x}^0$ then (5.3) is locally weakly observable at $\boldsymbol{x}^0$.*

In practice, the rank condition states that if, for a given $\boldsymbol{x}^0$, (5.5) has full rank n then the system (5.3) is locally weakly observable at $\boldsymbol{x}^0$. Note that the converse is almost always true due to the following theorem.

Theorem 5.1.2 (Hermann & Krener [1977])**.** *If the system (5.3) is weakly controllable, then it is weakly observable if and only if it is locally weakly observable if and only if the observability rank condition is satisfied.*

5.2 A solution: Kalman filtering

It is well known that the Kalman Filter (KF) [Kalman, 1960] is the optimal state estimator for unconstrained, linear systems subject to a normally distributed process and measurement noise. In this part of the thesis, general filter formulations are presented, more details and formulations are available in the literature. As a starting point several references are given during this discussion.

Given the following system

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}_k \boldsymbol{x}_k + \boldsymbol{B}_k \boldsymbol{u}_k + \boldsymbol{G}_k \boldsymbol{w}_k \tag{5.9}$$

$$\boldsymbol{y}_k = \boldsymbol{C}_k \boldsymbol{x}_k + \boldsymbol{v}_k \tag{5.10}$$

where $\boldsymbol{x}_k \in \mathcal{R}^n$, $\boldsymbol{u}_k \in \mathcal{R}^m$, $\boldsymbol{y}_k \in \mathcal{R}^p$, and n, m, p are positive scalars, $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are mutually independent sequences of zero mean white Gaussian noise with joint covariance

$$\mathrm{E}\begin{bmatrix} \boldsymbol{w}_k \boldsymbol{w}_k^T & \boldsymbol{w}_k \boldsymbol{v}_k^T \\ \boldsymbol{v}_k \boldsymbol{w}_k^T & \boldsymbol{v}_k \boldsymbol{v}_k^T \end{bmatrix} = \begin{bmatrix} \boldsymbol{P}_k^w & 0 \\ 0 & \boldsymbol{P}_k^v \end{bmatrix}. \tag{5.11}$$

Given, also, the initial conditions

$$\hat{\boldsymbol{x}}_0 = \mathrm{E}\left[\boldsymbol{x}_0\right] \tag{5.12a}$$

$$\boldsymbol{P}_0^x = \mathrm{E}\left[(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^T\right] \tag{5.12b}$$

the Kalman Filter is obtained by minimizing the mean-square error between the state and its estimate. Generally, the KF algorithm is formulated in two steps, known as prediction and filtering. When the prediction step is running the filter computes the a priori state estimate and its covariance

$$\hat{\boldsymbol{x}}_{k+1|k} = \boldsymbol{A}_k \hat{\boldsymbol{x}}_{k|k} + \boldsymbol{B}_k u_k, \tag{5.13}$$

$$\boldsymbol{P}_{k+1|k}^x = \boldsymbol{A}_k \boldsymbol{P}_{k|k}^x \boldsymbol{A}_k^T + \boldsymbol{G}_k \boldsymbol{P}_k^w \boldsymbol{G}_k^T. \tag{5.14}$$

When the new measurement (5.10) is available, the a priori statistics, (5.13) and (5.14), are updated by the filtering step

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}^x \boldsymbol{C}_k^T \left(\boldsymbol{C}_k \boldsymbol{P}_{k|k-1}^x \boldsymbol{C}_k^T + \boldsymbol{P}_k^v\right)^{-1}, \tag{5.15}$$

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k \left(\boldsymbol{y}_k - \boldsymbol{C}_k \hat{\boldsymbol{x}}_{k|k-1}\right), \tag{5.16}$$

$$\boldsymbol{P}_{k|k}^x = \left[\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{C}_k\right] \boldsymbol{P}_{k|k-1}^x, \tag{5.17}$$

where $\boldsymbol{K}_k$ is the Kalman gain.

Kalman Filter and its several extensions are well established, and successfully applied to solve estimation problems. Among them the Extended Kalman Filter and the Unscented Kalman Filter will be presented for the state estimation of nonlinear systems.

For a more detailed Kalman Filter presentation, including the case where the cross-covariance terms $\boldsymbol{v}_k \boldsymbol{w}_k^T$ and $\boldsymbol{w}_k \boldsymbol{v}_k^T$ are not zero, the reader is referred to Simon [2006].

5.2.1 Extended Kalman Filter

The Extended Kalman Filter (EKF) is perhaps the most famous extension of the KF for nonlinear systems. There are several versions available in the literature, see for example Simon [2006].

Consider the following nonlinear system

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) \tag{5.18a}$$
$$\boldsymbol{y}_k = h(\boldsymbol{x}_k, \boldsymbol{v}_k) \tag{5.18b}$$

where $\boldsymbol{x}_k \in \mathcal{R}^n$, $\boldsymbol{u}_k \in \mathcal{R}^m$, $\boldsymbol{y}_k \in \mathcal{R}^p$, n, m, p are positive scalars, $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are normally distributed Gaussian noise sequences with joint covariance (5.11).

The EKF gives an approximation of the optimal state estimate, and the nonlinearities of the system (5.18) are approximated by a linearized version of the nonlinear model around the last state estimate. For this approximation to be valid, it is very important that the linearization is a good representation of the nonlinear model in all the uncertainty domain associated with the state estimate.

Given initial conditions

$$\hat{\boldsymbol{x}}_0 = \mathrm{E}[\boldsymbol{x}_0] \tag{5.19a}$$
$$\boldsymbol{P}_0^x = \mathrm{E}[(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^T] \tag{5.19b}$$

for each sample $k = 1, ..., \infty$, the EKF algorithm is formulated by the following steps.

- Calculate the following Jacobians around the last filtered state estimate $\hat{\boldsymbol{x}}_{k|k}$

$$\boldsymbol{A}_k = \left.\frac{\partial f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k|k}} \qquad \boldsymbol{G}_k = \left.\frac{\partial f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})}{\partial \boldsymbol{w}}\right|_{\boldsymbol{w}=\bar{\boldsymbol{w}}} \tag{5.20}$$

  where $\bar{\boldsymbol{w}}$ is the mean value of the process noise sequence $\boldsymbol{w}_k$.
- Apply the prediction step of the Kalman Filter

$$\hat{\boldsymbol{x}}_{k+1|k} = f(\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{u}_k, \bar{\boldsymbol{w}}) \tag{5.21}$$
$$\boldsymbol{P}_{k+1|k}^x = \boldsymbol{A}_k \boldsymbol{P}_{k|k}^x \boldsymbol{A}_k^T + \boldsymbol{G}_k \boldsymbol{P}_k^w \boldsymbol{G}_k^T \tag{5.22}$$

- Calculate the following Jacobians around the predicted state estimate $\hat{\boldsymbol{x}}_{k+1|k}$

$$\boldsymbol{C}_k = \left.\frac{\partial h(\boldsymbol{x}, \boldsymbol{v})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k+1|k}} \qquad \boldsymbol{L}_k = \left.\frac{\partial h(\boldsymbol{x}, \boldsymbol{v})}{\partial \boldsymbol{v}}\right|_{\boldsymbol{v}=\bar{\boldsymbol{v}}} \tag{5.23}$$

  where $\bar{\boldsymbol{v}}$ is the mean value of the measurement noise sequence $\boldsymbol{v}_k$.

- Apply the filtering step to the linearized observation dynamics

$$\boldsymbol{K}_k = \boldsymbol{P}^x_{k|k-1}\boldsymbol{C}_k^T \left(\boldsymbol{C}_k\boldsymbol{P}^x_{k|k-1}\boldsymbol{C}_k^T + \boldsymbol{L}_k\boldsymbol{P}^v_k\boldsymbol{L}_k^T\right)^{-1}, \tag{5.24}$$

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\left(\boldsymbol{y}_k - h(\hat{\boldsymbol{x}}_{k|k-1}, \bar{\boldsymbol{v}})\right), \tag{5.25}$$

$$\boldsymbol{P}^x_{k|k} = \left[\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C}_k\right]\boldsymbol{P}^x_{k|k-1}, \tag{5.26}$$

  where $\boldsymbol{K}_k$ is the Kalman gain.

The EKF is not an optimal filter, but rather it is implemented based on approximations. This implies that the matrix $\boldsymbol{P}^x$ is not the true covariance but an approximation. While the Kalman Filter for linear systems is an optimal state estimator, which always provides state estimate convergence, if the observability condition (5.1.2) is met, the EKF may diverge if the approximation due the the consecutive linearization is not sufficiently accurate.

5.2.2 Unscented Kalman Filter

A more recent approach than the EKF, to the state estimation problem, is the Unscented Kalman Filter. The UKF uses the Unscented Transformation, which is based on the idea that is easier to approximate a probability distribution than an arbitrary nonlinear function or transformation [Julier & Uhlmann, 1996]. This approximation is done using a finite set of points, called sigma points. An important feature of the UKF, with respect to the EKF, is that no Jacobians need to be computed. This is relevant especially in the case of strong nonlinearities, as the introduction of linearization errors is avoided. In general both filters have similar computational complexity Wan & Van Der Merwe [2000], and both implementations are straightforward.

In the following a UKF formulation is presented, more precisely the one used in Wan & Van Der Merwe [2001].

Given the nonlinear system (5.18), define $L = n + dim(\boldsymbol{w}) + dim(\boldsymbol{v})$, where $dim(\cdot)$ indicates the dimension of $(\cdot)$. Define the following scalar weights W_i

$$W_0^{(m)} = \frac{\lambda}{(L+\lambda)} \tag{5.27a}$$

$$W_0^{(c)} = \frac{\lambda}{(L+\lambda)} + (1 - \alpha^2 + \beta) \tag{5.27b}$$

$$W_i^{(m)} = W_i^{(c)} = \frac{1}{2(L+\lambda)}, \qquad i = 1, ...2L \tag{5.27c}$$

with $\lambda = \alpha^2(L + \kappa) - L$. Note that these weights may be positive, or negative, but to provide an unbiased estimate, $\sum_i^{2L+1} W_i^{(m)} = 1$ must be satisfied. The design parameters α and κ control the spread of sigma points, β is related to the distribution of the random

variable $\boldsymbol{x}$. In most cases typical values are $\beta = 2$, and $\kappa = 0$ or $\kappa = 3 - n$, leaving only the parameter α as design parameter. Considering that $1 \cdot 10^{-5} \leq \alpha \leq 1$, the tuning of the UKF becomes simpler. For finer tuning and a more comprehensive description of the UKF parameters see Wan & Van Der Merwe [2001]. As it is the case for the EKF, the initial covariance matrices can be also used for performance tuning.

Given initial conditions

$$\hat{\boldsymbol{x}}_0 = \mathrm{E}[\boldsymbol{x}_0] \tag{5.28a}$$

$$\boldsymbol{P}_0^x = \mathrm{E}[(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^T], \tag{5.28b}$$

the modified nonlinear dynamics matrix $\mathbf{F}\left(\boldsymbol{\mathcal{X}}_j^x, \boldsymbol{u}_j, \boldsymbol{\mathcal{X}}_j^w\right)$

$$\left[f\left(\boldsymbol{\mathcal{X}}_j^{x(0)}, \boldsymbol{u}_j, \boldsymbol{\mathcal{X}}_j^{w(0)}\right) \quad f\left(\boldsymbol{\mathcal{X}}_j^{x(1)}, \boldsymbol{u}_j, \boldsymbol{\mathcal{X}}_j^{w(1)}\right) \quad \ldots \quad f\left(\boldsymbol{\mathcal{X}}_j^{x(2L+1)}, \boldsymbol{u}_j, \boldsymbol{\mathcal{X}}_j^{w(2L+1)}\right),\right] \tag{5.29}$$

and the observation mapping $\mathbf{H}\left(\boldsymbol{\mathcal{X}}_j^x, \boldsymbol{\mathcal{X}}_j^v\right)$

$$\left[h\left(\boldsymbol{\mathcal{X}}_j^{x(0)}, \boldsymbol{u}_j, \boldsymbol{\mathcal{X}}_j^{w(0)}\right) \quad h\left(\boldsymbol{\mathcal{X}}_j^{x(1)}, \boldsymbol{u}_j, \boldsymbol{\mathcal{X}}_j^{w(1)}\right) \quad \ldots \quad h\left(\boldsymbol{\mathcal{X}}_j^{x(2L+1)}, \boldsymbol{u}_j, \boldsymbol{\mathcal{X}}_j^{w(2L+1)}\right),\right] \tag{5.30}$$

where $f(\cdot)$ and $h(\cdot)$ are defined in (5.18b); the subscript j indicates the sampling time index; the superscript (i) for $i = (0, 1, \ldots, 2L + 1)$ and the sigma points $\boldsymbol{\mathcal{X}}_j$ are defined in (5.32-5.33).

For each sample $k = 1, ..., \infty$, the UKF algorithm is formulated as follow.

- Augment the state vector and the covariance matrix

$$\hat{\boldsymbol{x}}_k^a = \mathrm{E}[\boldsymbol{x}_k^a] = [\boldsymbol{x}_k^T \; \mathbf{0}_w^T \; \mathbf{0}_v^T]^T \tag{5.31a}$$

$$\boldsymbol{P}_k^a = \begin{bmatrix} \boldsymbol{P}_k^x & 0 & 0 \\ 0 & \boldsymbol{P}_k^w & 0 \\ 0 & 0 & \boldsymbol{P}_k^v \end{bmatrix} \tag{5.31b}$$

- Calculate the sigma point matrix $\boldsymbol{\mathcal{X}}_{k-1}^a$, where its i-th column vector $\boldsymbol{\mathcal{X}}_{k-1}^{a(i)}$ is defined as

$$\boldsymbol{\mathcal{X}}_{k-1}^{a(0)} = \hat{\boldsymbol{x}}_{k-1} \tag{5.32a}$$

$$\boldsymbol{\mathcal{X}}_{k-1}^{a(i)} = \hat{\boldsymbol{x}}_{k-1} + \gamma \left(\sqrt{\boldsymbol{P}_{k-1}^a}\right)_i \qquad i = 1, \ldots, L \tag{5.32b}$$

$$\boldsymbol{\mathcal{X}}_{k-1}^{a(i)} = \hat{\boldsymbol{x}}_{k-1} - \gamma \left(\sqrt{\boldsymbol{P}_{k-1}^a}\right)_{i-L} \qquad i = L+1, \ldots, 2L \tag{5.32c}$$

where $\gamma = \sqrt{L+\lambda}$, and $\left(\sqrt{\boldsymbol{P}^a_{k-1}}\right)_i$ is the i-th column vector of the square root of the covariance matrix $\boldsymbol{P}^a_{k-1}$. Note that $\boldsymbol{\mathcal{X}}^a_{k-1}$ has dimension $L \times (2L+1)$ and is partitioned as

$$\boldsymbol{\mathcal{X}}^a_{k-1} = \begin{bmatrix} \boldsymbol{\mathcal{X}}^x_{k-1} \\ \boldsymbol{\mathcal{X}}^w_{k-1} \\ \boldsymbol{\mathcal{X}}^v_{k-1} \end{bmatrix} \tag{5.33}$$

with respect to row vectors.

- Propagate the sigma points through the nonlinear dynamics matrix defined in (5.29), properly modified to allow the correct sigma point propagation, and compute the predicted state estimate, where the index i is used to select the appropriate sigma point column

$$\boldsymbol{\mathcal{X}}^x_{k|k-1} = \mathbf{F}\left(\boldsymbol{\mathcal{X}}^x_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{\mathcal{X}}^w_{k-1}\right) \tag{5.34}$$

$$\hat{\boldsymbol{x}}^-_k = \sum_{i=0}^{2L} W^{(m)}_i \boldsymbol{\mathcal{X}}^{x(i)}_{k|k-1}. \tag{5.35}$$

  Note that the *a priori* state estimate $\hat{\boldsymbol{x}}^-_k$ is found as a weighted sum of the propagated sigma points.

- Compute the predicted covariance, instantiate the prediction points through the observation mapping defined in (5.30). This is properly modified to allow the correct sigma point propagation, compute the predicted state estimate, and calculate the predicted measurement

$$\boldsymbol{P}^{x-}_k = \sum_{i=0}^{2L} W^{(c)}_i \left[\boldsymbol{\mathcal{X}}^{x(i)}_{k|k-1} - \hat{\boldsymbol{x}}^-_k\right]\left[\boldsymbol{\mathcal{X}}^{x(i)}_{k|k-1} - \hat{\boldsymbol{x}}^-_k\right]^T \tag{5.36}$$

$$\boldsymbol{\mathcal{Y}}_{k|k-1} = \mathbf{H}\left(\boldsymbol{\mathcal{X}}^x_{k|k-1}, \boldsymbol{\mathcal{X}}^v_{k|k-1}\right) \tag{5.37}$$

$$\hat{\boldsymbol{y}}^-_k = \sum_{i=0}^{2L} W^{(m)}_i \boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1}. \tag{5.38}$$

- Obtain the innovation covariance and the cross covariance matrices

$$\boldsymbol{P}_{\tilde{y}_k\tilde{y}_k} = \sum_{i=0}^{2L} W^{(c)}_i \left[\boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} - \hat{\boldsymbol{y}}^-_k\right]\left[\boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} - \hat{\boldsymbol{y}}^-_k\right]^T \tag{5.39}$$

$$\boldsymbol{P}_{y_k x_k} = \sum_{i=0}^{2L} W^{(c)}_i \left[\boldsymbol{\mathcal{X}}^{(i)}_{k|k-1} - \hat{\boldsymbol{x}}^-_k\right]\left[\boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} - \hat{\boldsymbol{y}}^-_k\right]^T. \tag{5.40}$$

- Perform the measurement update

$$\boldsymbol{\mathcal{K}}_k = \boldsymbol{P}_{y_k x_k} \boldsymbol{P}^{-1}_{\tilde{y}_k \tilde{y}_k}, \tag{5.41}$$

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}^-_k + \boldsymbol{\mathcal{K}}_k \left(\boldsymbol{y}_k - \hat{\boldsymbol{y}}^-_k \right), \tag{5.42}$$

$$\boldsymbol{P}^x_k = \boldsymbol{P}^{x-}_k - \boldsymbol{\mathcal{K}}_k \boldsymbol{P}_{\tilde{y}_k \tilde{y}_k} \boldsymbol{\mathcal{K}}^T_k. \tag{5.43}$$

Note that no linearization procedure is required. Moreover, a square root of matrix, $\sqrt{\boldsymbol{P}^a_{k-1}}$, has to be calculated at every time step, thus a numerical stable and efficient algorithm must be used, for example the Cholesky factorization. This is the most computationally demanding operation. A more efficient implementation is the square-root UKF [Van Der Merwe & Wan, 2001] which uses a recursive form of the Cholesky factorization.

The algorithm implementation is straightforward, since only simple operations need to be performed, e.g. weighted sums. Other UKF formulations are available in the literature, see for instance Wan & Van Der Merwe [2001] where an algorithm that uses a reduced number of sigma points is shown. For example, this is well suited when process and measurement noise realizations are assumed to be additive.

In general, UKF gives an accuracy of the second-order for the state estimate. This is the same as the standard EKF algorithm approach. However, for Gaussian distribution, a particular choice of the UKF parameter β, i.e. $\beta = 2$, gives an accuracy of the fourth-order term in the Taylor series expansion of the covariance, as discussed in the appendix of Julier & Uhlmann [2004].

Some successful applications of the UKF are described in Kandepu et al. [2008], where several comparisons with the EKF are discussed, for standard examples used in the literature, such as Van der Pol oscillator, and also for a solid oxide fuel cell combined gas turbine hybrid system. Van Der Merwe et al. [2004] give an interesting discussion of a family of Sigma-point based filters and compares their performance with the EKF, for an unmanned aerial vehicle where GPS measurements are integrated with a inertial measurement unit. An interesting discussion on UKF estimation and noise modeling is included in Kolås [2008]. Finally, in Spivey et al. [2010] the UKF is applied to an industrial process fouling and its performance is compared with Moving Horizon Estimation and EKF for a set of experimental data.

In next sections, the advantages of UKF compared to EKF are illustrated in two examples. Firstly, estimation for a microalgae photobioreactor is given. The results presented are particularly interesting since the performance of both estimators are compared with respect to real data obtained from laboratory experiments. Secondly, a well constructed example shows how both filters perform when a locally weakly unobservable nonlinear system is considered. This is shown in an NMPC framework.

5.3 Unscented and Extended Kalman filtering for a photobioreactor

Frequently, in the chemical and biochemical process industry, there is the necessity to monitor and control chemical reactions. This is usually done by using sensors that give measurements. Typically, measurements of reactant and product concentrations, operating temperatures, pressures, and other parameters are obtained. In general, a measurement has to be reliable, i.e. it has to be available and accurate. However, there are several reasons why the required measurements may not be reliable. Some of these reasons are the impracticability of building an appropriate sensor due to lack of technology, the difficulty of the positioning the sensor, the associated cost. In such cases, an attempt to use estimation techniques may be necessary.

Dochain [2003] presents an interesting overview of available results on state and parameter estimation in chemical and biochemical processes. A comparison of several traditional state and parameter estimation approaches is given, discussing the pros and cons of different cases, and describing how the most common implementation problems are solved (see Dochain [2003] and references therein).

In this section, an application of the UKF to a photobioreactor for microalgae production is shown. Microalgae have many applications such as the production of high value compounds (source of long-chain polyunsaturated fatty acids, vitamins, and pigments), in energy production (e.g. microalgae hydrogen, biofuel, methane) or in environmental remediation (especially carbon dioxide fixation and greenhouse gas emissions reduction). However, the photobioreactor microalgae process needs complex and costly hardware, especially for biomass measurement. There are also problems with the practicability of finding reliable online sensors that are able to measure the state variables (Shimizu [1996]). Thus, state and parameter estimation seems to be a critical issue and is studied in the case of a culture of the microalgae Porphyridium purpureum.

The intention of this work is to present the advantages of UKF in terms of performance and implementation ease, compared to the EKF. The work of Becerra-Celis et al. [2008] is considered as starting point. Therein it is shown how to implement an EKF for state estimation in a photobioreactor. Due to the operational data availability for biomass measurement, the results are thus validated. Numerical simulations in batch mode, and real-life experiments in continuous mode are given in order to highlight the performance of the proposed estimator. For the state estimation work on the photobioreactor presented in this chapter, both the model and the experimental data used for validating the result are taken from Becerra-Celis et al. [2008]

5.3.1 Photobioreactor for microalgae production

Strain and growth conditions

The photobioreactor is used to produce the red microalgae *Porphyridium purpureum* SAG 1830-1A obtained from the Sammlung von Algenkulture Pflanzenphysiologister Institut Universität Göttingen, Germany. The strain is grown and maintained on Hemerick medium (Hemerick [1973]). The pH of the Hemerick medium is adjusted to 7.0 before autoclaving it for 20 minutes at 121 $^{\circ}\mathrm{C}$. Cultures are maintained at 25 $^{\circ}\mathrm{C}$ in 500 ml flask containing 400 ml culture under continuous light intensity of 70 $\mu Em^{-2}s^{-1}$ and aerated with air containing 1% (v/v) CO_2 at 100 rpm on an orbital shaker. During the exponential growth phase, within an interval of two weeks, 200 ml of culture are transferred to a new flask containing fresh medium.

Culture conditions and measurements

Figure 5.1 illustrates the photobioreactor diagram where the growth of cultures is performed. The bubble column photobioreactor has a working height of 0.4 m and a diameter of 0.1 m. The total culture volume is 2.5 l, and the cylindrical reactor, made of glass, has an illuminated area of 0.1096 m^2. To agitate the culture an air mixture with 2% (v/v) CO_2 is continuously supplied at a flow rate of 2.5 $V.V.H$ (gas volume per liquid culture volume per hour). 0.22 μm Millipore filters, appropriate valves and flowmeters are used to filter and to control the air flow rate entering the photobioreactor. Four OSRAM white fluorescent tubes (L30W/72) and three OSRAM pink fluorescent tubes (L30W/77) are arranged around the bubble column to provide the external light source. The incident light intensity on the reactor surface is measured at ten different locations with flat surface quantum sensors (LI-COR LI-190SA). The average light intensity is computed by the weighted average of all measurements. The optimal value of irradiance on surface for the reactor is found to be 120 $\mu Em^{-2}s^{-1}$. A transparent jacket connected to a thermostat unit enables the temperature control, regulated to 25 $^{\circ}\mathrm{C}$. Other sensors are a pH sensor (Radiometer Analytical) and a dissolved oxygen sensor (Ingold type 170). A sampling port is applied to the top of the column, from where samples for off line analysis are collected after 6, 8, and 12 hours. The number of cells is counted using an optical microscope ZEISS Axioplan-2 on Malassez cells. The total inorganic carbon (T.I.C.) in the culture medium is calculated by gas phase chromatography. This method, proposed by Marty et al. [1995], is used to measure low inorganic carbon concentrations down to $(10^{-6} mol\ l^{-1})$ within an accuracy of 10 %.

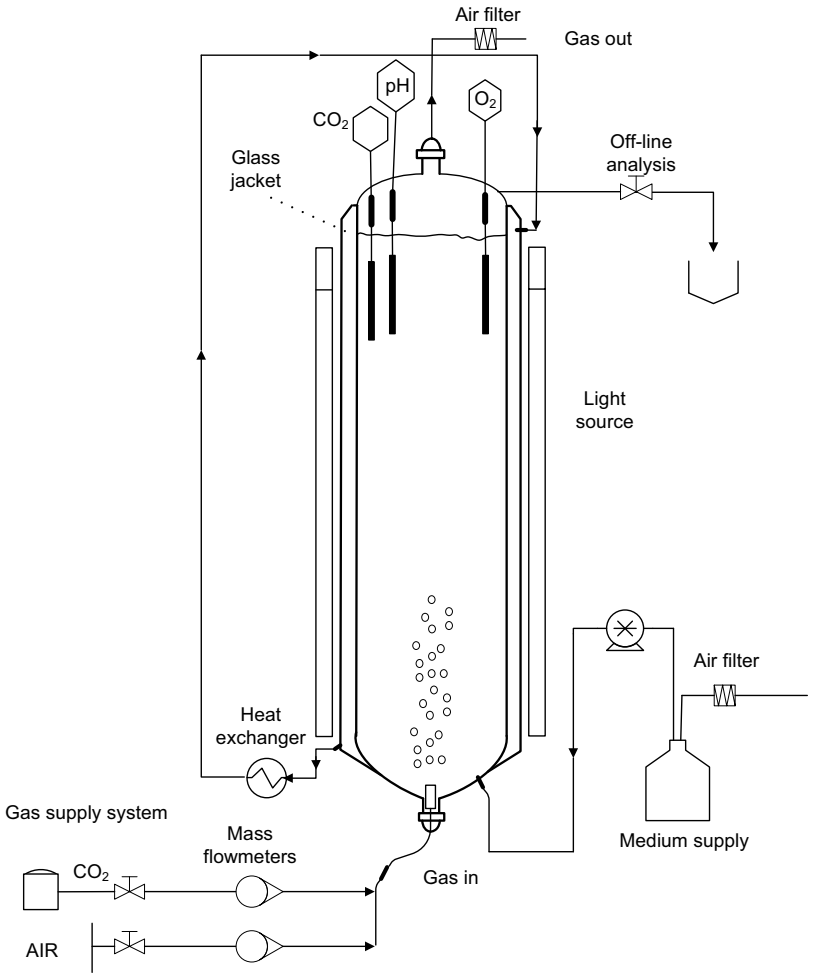


Figure 5.1: Photobioreactor diagram.

Mathematical model

The bioprocess model presented in Baquerisse et al. [1999] is used. It consists of two sub models, one describing the growth kinetics, and the other representing the gas-liquid mass transfer in the photobioreactor. This results in two differential equations describing the state of the reactor

$$\frac{dX}{dt} = \frac{F_{in}}{V}X_{in} + \mu X - \frac{F_{out}}{V}X \tag{5.44a}$$

$$\frac{d[TIC]}{dt} = \frac{F_{in}}{V}[TIC]_{in} - \frac{F_{out}}{V}[TIC]_{out} - \mu\frac{X}{Y_{X/S}}mX + k_La\left([CO_2^*] - [CO_2]\right) \tag{5.44b}$$

where X is the biomass, and $[TIC]$ is the inorganic carbon concentration. The subscripts $[\cdot]_{in}$ and $[\cdot]_{out}$ indicate quantities flowing into, and out from the reactor, respectively. V is the culture volume, and F is the medium flow rate. The mass conversion yield is defined by $Y_{X/S}$, m is the maintenance coefficient, and k_La is the gas-liquid transfer coefficient. The carbon dioxide concentration for the fresh medium is defined as

$$[CO_2^*] = \frac{PCO_2}{\mathcal{H}} \tag{5.45}$$

where PCO_2 is the partial pressure of carbon dioxide, and $\mathcal{H}$ is the Henry's constant for Hemerick medium. Moreover, the carbon dioxide concentration in the medium is given by:

$$[CO_2] = \frac{[TIC]}{\left[1 + \frac{K_1}{[H^+]} + \frac{K_1K_2}{[H^+]^2}\right]} \tag{5.46}$$

where K_1, K_2 are dissociation equilibrium constants, and $[H^+]$ is defined as

$$[H^+] = 10^{-pH} \tag{5.47}$$

representing the hydrogen ions concentration in the culture media.

In addition, a light transfer model is considered, which describes the evolution of incident and outgoing light intensity

$$E = \frac{(I_{in} - I_{out})A_r}{VX}, \tag{5.48}$$

$$I_{out} = C_1 I_{in} X^{C_2}, \tag{5.49}$$

where E is the light "energy" accessible per cell, I_{out} is the outgoing light intensity, I_{in} is the ingoing light intensity. C_1, C_2 are constants depending on the reactor geometry, and A_r is its area.

The light intensity and the total carbon concentration influence the specific growth rate

$$\mu = \mu_{max} \frac{E}{E_{opt}} e^{\left(1-\frac{E}{E_{opt}}\right)} \frac{[TIC]}{[TIC]_{opt}} e^{\left(1-\frac{[TIC]}{[TIC]_{opt}}\right)} \tag{5.50}$$

where μ_{max}, E_{opt}, and $[TIC]_{opt}$ are model parameters identified from the batch data experiments. Finally, the substrate limitation effect, which is the tendency of cells to distribute in layers, is taken into account in (5.50). An example of substrate limitation is when cells accumulate close to the light source.

Batch and continuous operating conditions

The photobioreactor can work in two different operating conditions, batch mode and continuous mode. In batch mode:

$$F_{in} = F_{out} = 0; \qquad [TIC]_{in} = 0; \qquad X_{in} = 0. \tag{5.51}$$

In continuous mode, instead:

$$F_{in} = F_{out} \neq 0. \tag{5.52}$$

Model parameters

The model parameters used in this work are the ones identified in Becerra-Celis et al. [2008]. For more details on the system identification procedure the reader is referred to their work. Tables 5.1 and 5.2 contain the parameters for the microalgae and the total inorganic carbon dynamics, respectively.

Table 5.1: Model parameters for *Porphyridium purpureum* at 25 °C.

Parameter	Unit	Value
μ_{max}	h^{-1}	0.0337
E_{opt}	$\mu E s^{-1}(10^9 cell)^{-1}$	1.20
$[TIC]_{opt}$	$mmole^{-1}$	12.93
C_1		0.28
C_2		-0.55

Table 5.2: Model parameters for [TIC] dynamics.

Parameter	Unit	Value
K_1		$1.02 \cdot 10^{-6}$
K_2		$8.32 \cdot 10^{-10}$
$k_L a$	h^{-1}	41.40
m	$h^{-1} mmole(10^9 cell)^{-1}$	0.004
$Y_{X/S}$	10^9 *cell per mole* TIC	198.1
$\mathcal{H}$	$atm\ l\ mole^{-1}$	34.03

5.3.2 Biomass estimation

Using an appropriate numerical integration routine, the photobioreactor model (5.44) can be written in the following form

$$\boldsymbol{\xi}_{k+1} = f(\boldsymbol{\xi}_k, u_k, \boldsymbol{w}_k^{\xi}; \boldsymbol{\eta}_k) \tag{5.53a}$$
$$\boldsymbol{\eta}_{k+1} = \boldsymbol{\eta}_k + \boldsymbol{w}_k^{\eta} \tag{5.53b}$$
$$y_k = [0\ 1]\boldsymbol{\xi}_k + v_k \tag{5.53c}$$

where the state vector is

$$\boldsymbol{\xi}_k = \begin{bmatrix} X_k \\ [TIC]_k \end{bmatrix} \tag{5.54}$$

$u_k = F_{in}$ is the input, $y_k = [TIC]$ is the measurement, $\boldsymbol{v}_k^{\xi}$ is the process noise, v_k is the measurement noise, of appropriate dimensions, respectively. Here, (5.53b) is the parameter equation, where the parameter dynamics is modeled as a random walk driven by a white noise process $\boldsymbol{w}_k^{\eta}$. A relatively small covariance is associated to $\boldsymbol{w}_k^{\eta}$ to consider the slowly varying nature of the parameter vector. Finally, when the measurement of PCO_2, pH, and I_{in} are available, they are used for updating model parameters.

It may happen that parameters used in the model (5.53) are uncertain or inaccurate. This would decrease the estimator model accuracy. One method to obtain sufficiently good estimate is to make the estimator algorithm robust with respect to the parameter variations. Another method is to try estimating the uncertain parameters for updating the model, and thereby obtaining better accuracy. However, joint parameter and state estimation may lead to observability problems. Therefore one has to be careful about choosing a subset of parameter to estimate, so that the system observability is not penalized. A general framework to introduce the parameter estimation is to extend the state with the uncertain parameters vector and then estimate the augmented state, as shown next for the UKF case.

Modified Unscented Kalman Filter algorithm

To use the UKF for joint state and parameter estimation, the algorithm (5.28-5.43) must be modified as follow.

Define the new state vector

$$\hat{\boldsymbol{x}}_k = \begin{bmatrix} \hat{\boldsymbol{\xi}}_k \\ \hat{\boldsymbol{\eta}}_k \end{bmatrix} \tag{5.55}$$

which has as elements, the state and parameter estimates, respectively. The vector

$$\boldsymbol{w}_k = \begin{bmatrix} \boldsymbol{w}_k^{\xi} \\ \boldsymbol{w}_k^{\eta} \end{bmatrix} \tag{5.56}$$

contains the process noises in the evolution of $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$. Analogously, define the augmented covariance matrix

$$\boldsymbol{P}_k = \begin{bmatrix} \boldsymbol{P}_k^{x} & \boldsymbol{P}_k^{x,w} \\ \boldsymbol{P}_k^{w,x} & \boldsymbol{P}_k^{w} \end{bmatrix} \tag{5.57}$$

where $\boldsymbol{P}_k^x$ consists of the state and parameter error covariances, while $\boldsymbol{P}_k^w$ includes the process noise covariance associated to state and parameters. In addition, the off diagonal entries are cross covariance terms represented by the notation $\boldsymbol{P}_k^{\cdot,\cdot}$. Obviously, all elements of $\hat{\boldsymbol{x}}_k$ and $\boldsymbol{P}_k$ are of appropriate dimensions.

Given the initial conditions

$$\hat{\boldsymbol{x}}_k = \begin{bmatrix} \hat{\boldsymbol{\xi}}_k \\ \hat{\boldsymbol{\eta}}_k \end{bmatrix} \qquad\qquad \boldsymbol{P}_0 = \begin{bmatrix} \boldsymbol{P}_0^{x} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{P}_0^{w} \end{bmatrix} \tag{5.58}$$

where the new system state $\hat{\boldsymbol{x}}$ has dimension n, it is possible to apply the UKF algorithm (5.28-5.43) and jointly estimate the state and parameter vectors.

Simulation results with experimental data

Due to modifications just introduced, the UKF described in Section 5.3.2 can be implemented. The main objective is to estimate the biomass X in the photobioreactor of Section 5.3.1. Focusing on the two different working conditions defined in Section 5.3.1, it is observed how in batch mode the UKF has excellent performance, which also is the case for the EKF designed in Becerra-Celis et al. [2008]. This is due to the fact that the model parameters are identified in batch mode, and the measurements have a constant sampling time. A more complex scenario appears for continuous cultures. The model parameters are still the ones from the batch experiments, and the experimental data are collected at variable instant intervals. Due to the variable time steps, Becerra-Celis et al. [2008] implement a continuous discrete version of the EKF. In the present work, this problem is

tackled in two steps. Firstly, a zero order hold is applied to the measurements, secondly the standard UKF algorithm is properly modified. In more detail, the discrete UKF algorithm with an augmented state to consider parameter estimation and process noise is implemented. The sigma points are recomputed, before propagating them through the observation model. This modification gives the possibility to use the discrete algorithm with the irregular measurement sampling time of the continuous culture case. Furthermore, the parameter μ_{max} in (5.50) is chosen to be estimated. Finally, despite the fact that a zero order hold is used to permit a discrete UKF implementation, the UKF accuracy and speed of convergence are improved with respect to those in the EKF.

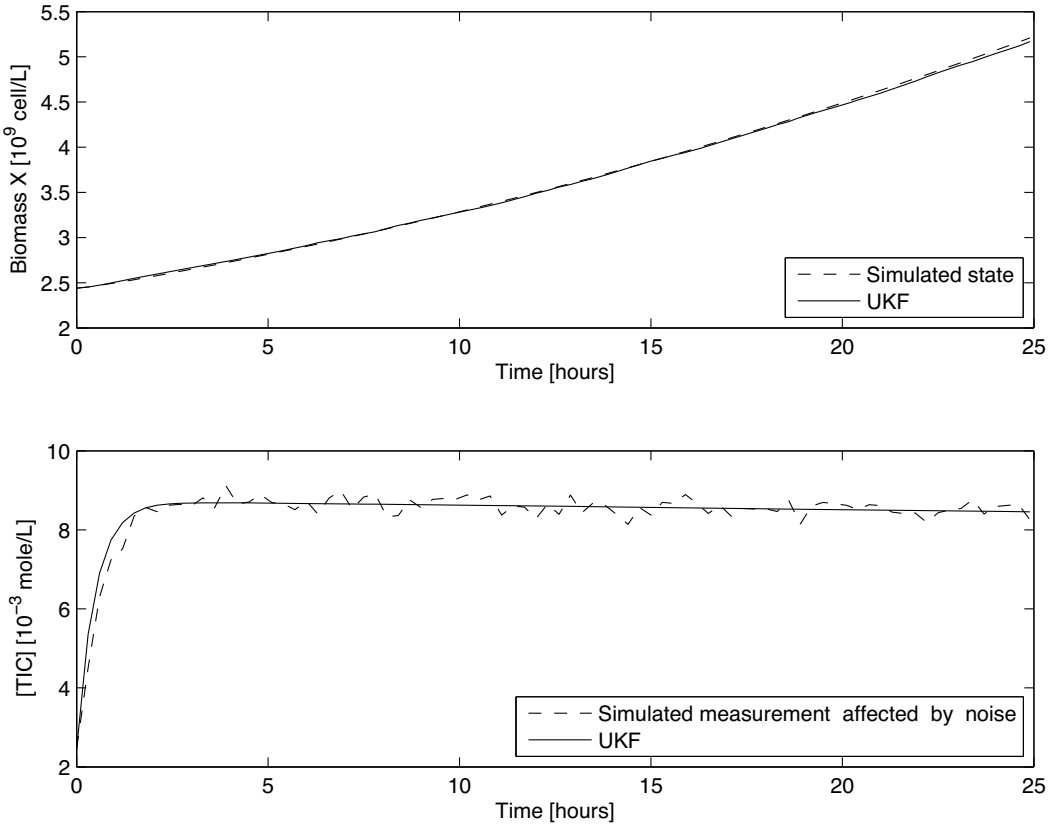
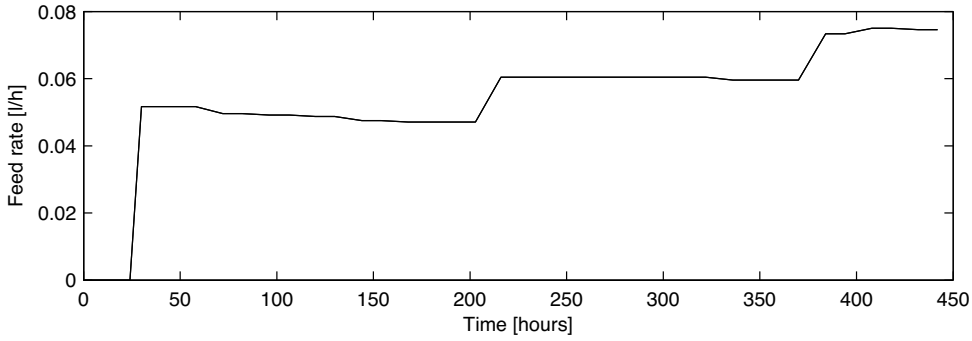
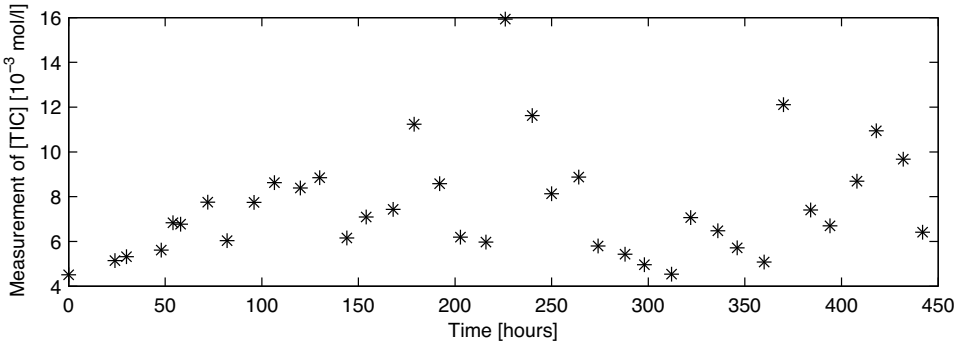


Figure 5.2: UKF estimation for simulated batch mode.

Figure 5.2 illustrates the convergence of the UKF in simulated batch mode, for which conditions (5.51) hold. In this case the nonlinear model (5.44a-5.44b) is discretized at sampling time $T_s = 0.5\ h$ and used to simulate the state of the process, starting from



(a) Culture medium feeding profile



(b) Measurement of [TIC]

Figure 5.3: Experimental data: input and output of the photobioreactor collected in continuous mode.

initial conditions $X_0 = 2.44 \cdot 10^9 cell/l$, $[TIC]_0 = 2.55\ 10^{-3} mole/l$. After that, $[TIC]$ is corrupted by additive Gaussian white noise with standard deviation $\sigma = 0.2\ 10^{-3} mole/l$, and used as measurement for the UKF. Thus, the state is estimated successfully with excellent noise rejection in TIC. The results obtained for continuous cultures are even more interesting. Initial conditions are $X_0 = 1.8 \cdot 10^9 cell/l$, $[TIC]_0 = 4.51 \cdot 10^{-3} mole/l$, and in addition real experiment data, presented in Figure 5.3, are used as input to the filters. The EKF designed in Becerra-Celis et al. [2008], the UKF with only state estimation, and the UKF with joint state and parameter estimation are simulated. The results obtained are shown in Figure 5.4 and discussed here. In Figure 5.4(a) it is noticeable how both UKF implementations have faster speeds of convergence than the EKF. In Figure 5.4(b) the state estimation error of the three different approaches are compared, and it is evident how the UKFs give smaller estimation errors. Note that it was possible to compute

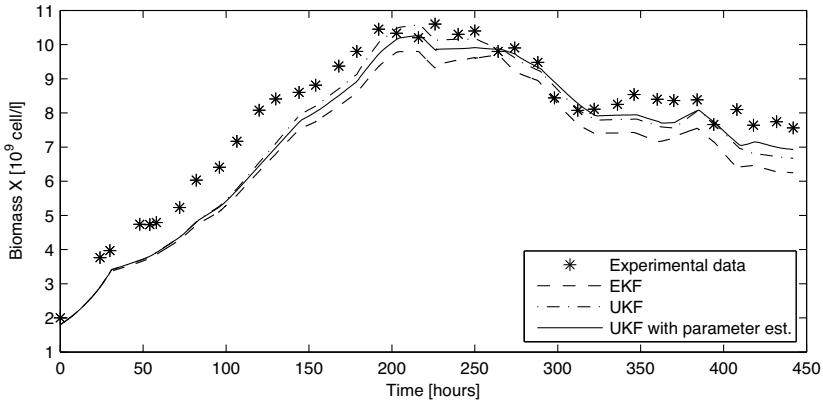
the state estimation error because experimental data for the biomass, shown with a $star$ in Figure 5.4(a), was available. Moreover the mean squared error (MSE) between the biomass and its estimate is computed. Table 5.3 shows the MSE index, which is obtained averaging the MSE along the entire simulation period. From both figures it is noticeable

Table 5.3: Mean Squared Error Index

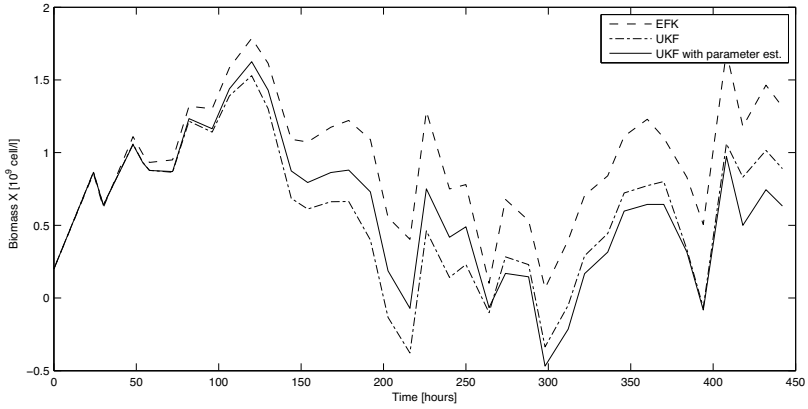
EKF	UKF	UKF with par. est.
13.60	6.12	6.12

how the UKF performs better than the EKF, and how the introduction of parameter estimation in the UKF improves the accuracy of the estimation in the final part (after 300 hours), although it slightly reduces the speed of convergence.

Since the parameters are identified from batch experiments, as shown in Becerra-Celis et al. [2008], adding parameter estimation may be useful when the photobioreactor is run in continuous mode. Figure 5.5 shows the evolution of the μ_{max} estimation, the estimated value is compared with the identified value and the UKF error covariance is also shown.

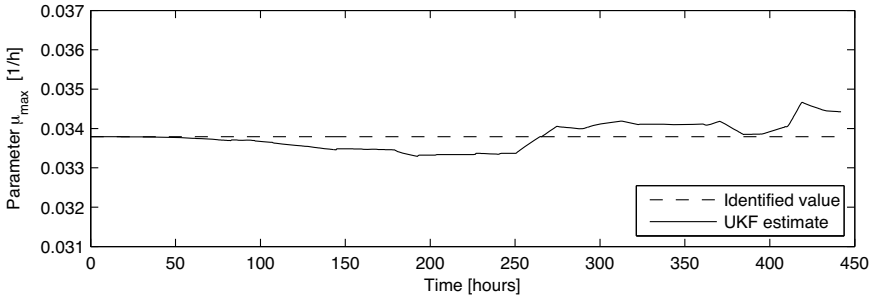


(a) Biomass estimate

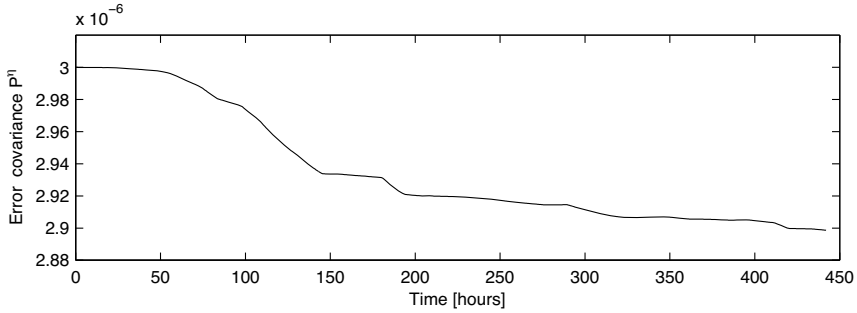


(b) State Estimation Error

Figure 5.4: Biomass estimation comparison for continuous cultures.



(a) UKF estimate and identified value of μ_{max}



(b) UKF error covariance of μ_{max}

Figure 5.5: UKF parameter estimate and its covariance for continuous culture.

5.4 State estimation in Nonlinear Model Predictive Control, UKF advantages

As discussed in Chapter 2, nonlinear model predictive control has proved to be a suitable technique for controlling nonlinear systems, since the simplicity of including constraints in its formulation makes it very attractive for a large class of applications. A limitation for the application of NMPC is that at every time step the state of the system is needed for prediction. However, it is not always possible to measure all the states, thus a filter or observer may be used. For nonlinear systems a Separation Theorem does not exist, so that even if the state feedback controller and the observer are both stable, there is no guarantee of closed-loop nominal stability. Findeisen et al. [2003] give an interesting overview on both state and output feedback NMPC, and Kolås et al. [2008] focus also on high order state estimators and noise model design.

State estimation introduces an extra computational load which can be relevant in the case of systems with relatively fast dynamics. In this case accurate estimation methods with low computational cost are desired, for example the Extended Kalman Filter. Clearly, the EKF does not perform well with all nonlinear systems, but its straightforwardness is the main reason of its popularity.

In this section, a type of locally weakly unobservable system is studied. For this type of system, we find that the EKF drifts because the system is unobservable at the desired operation point. Instead, it is shown how the UKF is used for state estimation in this type of nonlinear systems, and that it provides a stable state estimate, despite the fact that the system is locally unobservable.

5.4.1 An example of locally weakly unobservable system

As seen in Section 5.1.1, the fundamental requirement for an observer to work properly is to be associated with an observable system. The following example is constructed such that the the observability rank condition derived in Hermann & Krener [1977] is not satisfied. Thus, the state estimator might have problems once it reaches locally weakly unobservable regions.

Consider the scalar nonlinear system

$$\begin{aligned} \dot{x}(t) &= (0.1x(t) + 1)u(t) \\ y(t) &= x^3(t) \end{aligned} \tag{5.59}$$

and apply the rank condition described in Section 5.1.1. Clearly for this system the observability matrix (5.5) is

$$\mathcal{O}(x) = \begin{bmatrix} 3x^2 \end{bmatrix} \tag{5.60}$$

with rank equal to $n = 1$ for $x \neq 0$, and rank null for $x = 0$. Therefore, if we apply theorems 5.1.1 and 5.1.2 we can conclude that (5.59) is locally weakly observable for $x \neq 0$. At $x = 0$ the system is not locally weakly observable. Thus, in an NMPC context we implement both EKF and UKF state estimators. Note that the control goal is to drive the system to the origin.

Since a discrete time framework is required, we discretize (5.59) using the Euler approximation, and a sampling time of 1 second.

$$\begin{aligned} x_{k+1} &= x_k + (0.1x_k + 1)u_k + w_k \\ y_k &= x_k^3 + v_k \end{aligned} \tag{5.61}$$

where x is the state, u is the input, y is the measurement, w is the excitation state noise, v is the measurement noise, and the subscript k is the sampling time index. The noise sequences are both assumed to be Gaussian white noise.

Simulation-based results will show that while the EKF fails to estimate the state, the UKF is able to give a stable state estimate. This is due to the implicit structure of the UKF algorithm, in fact no linearization is used. Whereas, when the EKF algorithm is implemented, the linearization at $x = 0$ is the origin of EKF failure. In addition the pdf approximation obtained with the sigma points is more accurate than the EKF one.

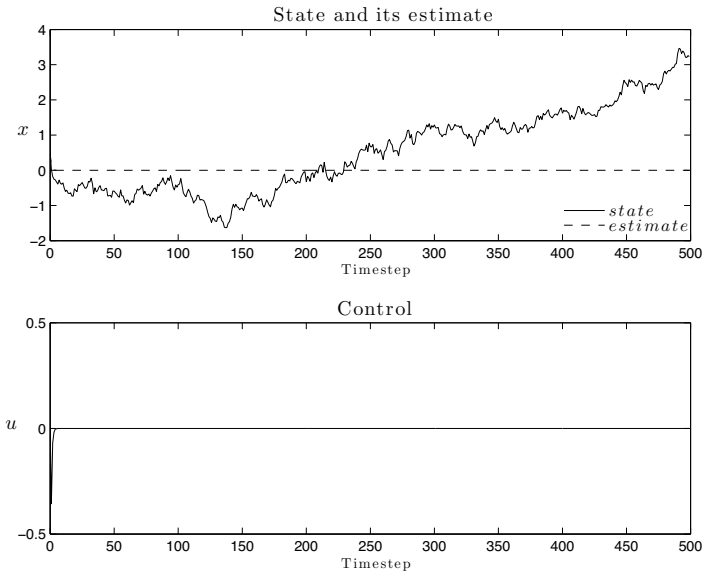
Simulation-based results

The NMPC framework (2.16) is applied with the quadratic stage cost function $J(x, u) = qx^2 + ru^2$, where the corresponding state weight is $q = 2$, and the input weight is $r = 1$. The prediction horizon length is $N_p = 10$. The process and measurement noises have additive white Gaussian distributions with zero means and variances $\sigma_\omega^2 = 0.01$, $\sigma_v^2 = 0.1$, respectively.

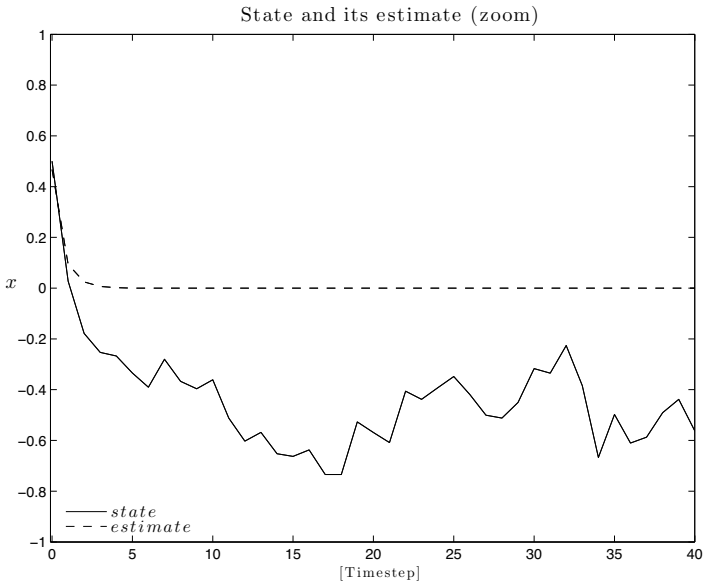
For state estimation, the EKF (5.19-5.26) first, and then with the UKF (5.28-5.43) are implemented. All the simulations are started from an initial condition $x_0 = 0.5$, with initial state variance $P_0 = 0.02$. The UKF tuning parameters are $n = 1$, $\alpha = 1$, $\beta = 2$, and $\kappa = 2$.

Figure 5.6 shows how the EKF fails. It is possible to note how once both the state and its estimate reach zero, the estimator is not able anymore to reconstruct the actual state (Figure 5.6(b)). This is due to the linearization problem. At zero the Kalman gain becomes zero and the filter is not able to correct the estimate with the future measurements. The controller uses the estimate, yielding a zero control signal as soon as the state estimate is null. As a result the controller is not able to regulate the state anymore and the state drifts.

In Figure 5.7 the UKF is used instead. It is possible to observe how the filter is able to estimate the state, and in the meantime there is no drift in the state estimation. Having a correct state estimate allows the NMPC to regulate the state, achieving its goal.

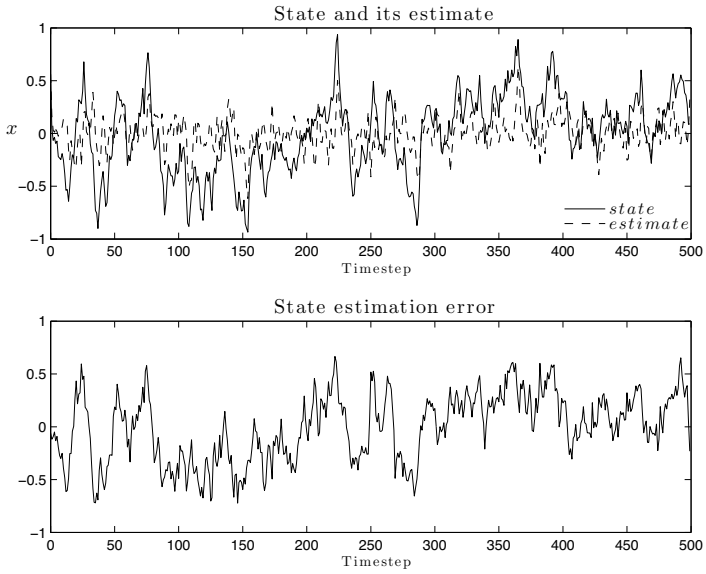


(a) System state, state estimate, control input

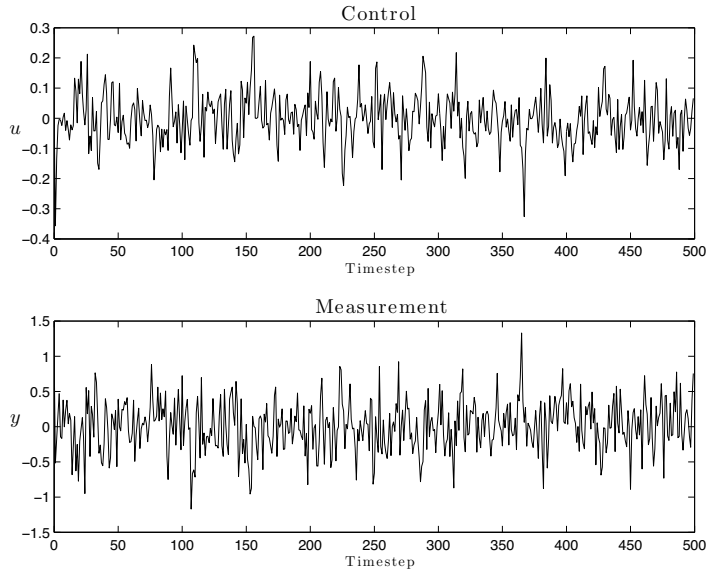


(b) State and its estimate: first 40 time steps

Figure 5.6: Simulation-based results: EKF as state estimator.



(a) System state and its estimate, state estimation error



(b) Control input and system measurement

Figure 5.7: Simulation-based results: UKF as state estimator.

5.5 Conclusions

In this chapter Kalman filtering for state estimation in a model predictive control framework is presented. Two examples of state estimation for non nonlinear systems are studied, and the EKF and UKF performances are compared.

In the first example, which can be considered as an extension of Becerra-Celis et al. [2008], it is shown how to improve the estimation performance by using UKF to implement a biomass estimator for a microalgae photobioreactor. In both batch and continuous mode, the approach presented produces faster estimate convergence and better estimate accuracy. The possibility to easily introduce joint parameter and state estimation, the absence of linearization, the comparable computational complexity make the UKF an attractive estimator for nonlinear systems. The results are validated by comparison with operational data.

In the second example, an output feedback Nonlinear Model Predictive Control is implemented for a regulation problem with a particular example from the family of locally weakly unobservable nonlinear systems. For these types of systems the Extended Kalman Filter may fail to give a correct state estimate, leading to a drift in the state estimation. Using a simple but effective state space representation of a particular nonlinear model, a set of simulations were carried out to show how the UKF gives a stable state estimate, despite operating at locally weakly unobservable operating points. Other estimation algorithms could be used, for instance Monte-Carlo-based methods, but the choice of a Kalman-based filter gives the advantage of lower overall computational complexity.

Chapter 6

Conclusions and recommendations for further work

In this chapter, first general conclusions, and then directions for future work are presented.

6.1 Conclusions

The thesis has focused on MPC performance improvement, and extending the range of applicability for MPC controllers. This is motivated by the success of MPC as an advanced control technique. MPC has a complex structure:

- internal model for plant behavior prediction;
- cost function for control quality/index definition;
- constraints for representing limitations in the plant, actuators, etc.;
- capacity of handling model/plant mismatch and unknown disturbances introducing feedback action by a receding horizon strategy.

All these elements give flexibility and adaptability, but at the same time introduce difficulties for theoretical analysis and possible extensions. This neither discourages researchers nor industry, as matter of fact, it is possible to find a vast body of literature and a large number of applications of MPC. The results presented in this thesis can be summarized in three points.

The state dependent input weight in the MPC cost function

This approach introduces smooth gain scheduling functionality into the MPC formulation, and simultaneously improves the conditioning of the Hessian, thus yielding a more robust optimization formulation. In Chapter 3 this is combined with various representations of the internal model in the MPC. The best performance and largest operating region is achieved with the state dependent input weight and an LTV model.

The novel PE-MPC formulation

This new formulation is obtained by extending standard MPC to allow for on-line model adaptation or tuning. It is shown how to extend the constraints set to include the persistence of excitation condition, necessary to obtain unbiased parameter estimation. The particular method of imposing the PE constraint introduces a new constraint only on the first manipulated variable, and the introduction of a backwards-looking horizon alters the MPC controller from memoryless full-state feedback to dynamic full-state feedback. This novel framework allows us to show formally and by examples with FIR and state space models that a periodic control signal arises autonomously, avoiding the necessity of forcing this condition by an equality constraint. In general, the formulation given is valid for Multi Input Multi Output systems and general adaptive schemes, yielding a non-convex optimization problem.

State estimation using UKF

The Unscented Kalman Filter was proposed relatively recently compared to the well known Extended Kalman Filter. They can both be used for state estimation of nonlinear systems. In UKF there is no need to use linearization procedures, eliminating then linearization errors. A set of well specified points, called sigma points, is used instead to approximate the statistics of the state. This resembles particle filtering, however, the Unscented Transformation allows to choose a relatively small set of 'particles' with obvious benefits on computational complexity. In fact, the UKF computational burden is comparable to the EKF one.

Two examples are used to illustrate and compare the performance of the two filters. In the first example a photobioreactor model is introduced, and operational data is also used to compare the state estimation error. In the second example a simulation-based study for a locally weakly unobservable nonlinear system is carried out. In both cases, UKF outperforms EKF, especially in the second case where the EKF fails to estimate once the system reaches a locally weakly unobservable operating point.

6.2 Further work

This thesis addresses some issues in the areas of MPC, dual control, adaptive control, as well as state estimation.

Each of these are by themselves large and important problem areas, with many relevant research issues. In the following, some remaining topics that are natural extensions and continuations of the work in this thesis will be proposed.

Cost function with state dependent input weight

This approach could be extended to other applications, where for instance there is a need to increase the region where the controller is able to perform its duties. Verification of the results in experiments and/or industrial applications would be of interest.

Persistently Exciting Model Predictive Control

Multi Input Multi Output system implementations yield non-convex optimization problem. A general non-convex optimizer may be used, but it would be interesting to exploit the PE-constraint structure to obtain efficient optimization implementation. The presented Single Input Single Output examples show efficient implementations of the optimization problems. However, extension to problems with multiple inputs is not necessarily straightforward. Comparison of different parameter estimation algorithms would also be of interest.

State Estimation issues

For the photobioreactor example, controller implementation coupled with UKF/EKF is the subsequent step. This will necessarily be left to our collaborators at Supélec.

For the observability issue, the study of larger scale systems would be the natural subsequent step. Furthermore, a more theoretical/analytical analysis, aimed at characterizing the locally weakly unobservable systems for which the UKF can work, could be performed.

Dual Control

It is well known that dual control is in general an intractable problem, but with continued improvements in both systems theory and computational power, the area should be revisited at regular intervals to re-assess what can realistically be achieved.

References

Allison, B. J., Ciarniello, J. E., Tessier, P. J. C., & Dumont, G. A. (1995). Dual adaptive control of chip refiner motor load. *Automatica*, 31(8), 1169–1184.

Anderson, B. D. O. & Johnson Jr, C. R. (1982). Exponential convergence of adaptive identification and control algorithms. *Automatica*, 18(1), 1–13.

Aoyama, A., Bell, G., & Walsh, S. P. (1997). Implementation issues in quadratic model predictive control. In *American Control Conference,* Albuquerque, New Mexico.

Bai, E. W. & Sastry, S. S. (1985). Persistency of excitation, sufficient richness and parameter convergence in discrete time adaptive control. *System and Control Letters*, 6, 153–163.

Baquerisse, D., Nouals, S., Isambert, A., Ferreira dos Santos, P., & Durand, G. (1999). Modelling of a continuous pilot photobioreactor for microalgae production. *Journal of Biotechnology*, 70(1-3), 335 – 342.

Bar-Shalom, Y. (1981). Stocastic dynamic programming: caution and probing. *IEEE Transactions on Automatic Control*, 26, 1184–1194.

Bar-Shalom, Y. & Tse, E. (1974). Dual effect, certainty equivalence and separation in stochastic control. *IEEE Transactions on Automatic Control*, AC-19, 494–500.

Bar-Shalom, Y. & Tse, E. (1976). Concept and methods in stochastic control. In C. T. Leondes (Ed.), *Control and Dynamic Systems*, volume 12 (pp. 99–172). New York: Academic Press.

Bayard, D. & Eslami, M. (1985). Implicit dual control for general stochastic systems. *Optimal Control Applications and Methods*, 6(3), 265–279.

Becerra-Celis, G., Tebbani, S., Joannis-Cassan, C., Isambert, A., & Boucher, P. (2008). Estimation of microalgal photobioreactor production based on total inorganic carbon in the medium. In *17th IFAC World Congress,* Seoul, South Korea.

Bemporad, A. (2006). Model predictive control design: New trends and tools. In *45th IEEE Conference on Decision and Control,* San Diego, California.

Bemporad, A., Morari, M., Dua, V., & Pistikopoulos, E. N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1), 3–20.

Bemporad, A., Morari, M., & Ricker, N. (2004). Model predictive control toolbox for matlab - user's guide. `http://www.mathworks.com/access/helpdesk/help/toolbox/mpc/`.

Besançon, G. (1999). *A viewpoint on observability and observer design for nonlinear systems*, volume 244 of *Lecture Notes in Control and Information Sciences: New Directions in nonlinear observer design*, (pp. 3–22). Springer Berlin/ Heidelberg.

Birge, J. R. & Louveax, F. (1997). *Introduction to Stochastic Dynamic Programming*. New York, New York: Springer-Verlag.

Boyd, S. & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press, UK.

Brustad, G. (1991). *Dual Control A Monte Carlo Approach*. PhD thesis, Norwegian Institute of Technology (NTH).

Camacho, E. & Bordons, C. (2004). *Model Predictive Control*. Springer-Verlag, 2nd edition.

Camacho, E. & Bordons, C. (2007). *Nonlinear Model Predictive Control: An Introductory Review*, volume 358 of *Lecture Notes in Control and Information Sciences*, assessment and future directions of nonlinear model predictive control 1, (pp. 1–16). Springer Berlin/ Heidelberg.

Chisci, L., Rossiter, J., & Zappa, G. (2001). Systems with persistent disturbances: predictive control with restricted constraints. *Automatica*, 37, 1019–1028.

Clarke, D., Mohtadi, C., & Tuffs, P. (1987a). Generalized predictive control-I. The basic algorithm. *Automatica*, 23(1), 137–148.

Clarke, D., Mohtadi, C., & Tuffs, P. (1987b). Generalized predictive control-II. Extensions and interpretations. *Automatica*, 23(2), 149–160.

Cutler, C. R. & Ramaker, B. L. (1979). Dynamic matrix control - A computer control algorithm. In *AIChE Annual Meeting,* Houston, Texas.

Cutler, C. R. & Ramaker, B. L. (1980). Dynamic matrix control - A computer control algorithm. In *Joint Automatic Control Conference,* San Francisco, California.

Dasgupta, S., Anderson, B. D. O., & Tsoi, A. C. (1990). Input conditions for continuous-time adaptive systems problems. *IEEE Transactions on Automatic Control*, 35(1), 78–82.

Dochain, D. (2003). State and parameter estimation in chemical and biochemical processes: a tutorial. *Journal of process control*, 13, 801 – 818.

Feldbaum, A. (1960a). Dual control theory (I). *Automation and Remote Control*, 21(9), 1240–1249.

Feldbaum, A. (1960b). Dual control theory (II). *Automation and Remote Control*, 21(11), 1453–1464.

Feldbaum, A. (1961a). Dual control theory (III). *Automation and Remote Control*, 22(1), 3–16.

Feldbaum, A. (1961b). Dual control theory (IV). *Automation and Remote Control*, 22(2), 129–143.

Filatov, N. M. & Unbehauen, H. (2004). *Adaptive Dual Control*. Springer.

Findeisen, R. & Allgöwer, F. (2002). An introduction to nonlinear model predictive control. In *21st Benelux Meeting on System and Control,* Veldhoven, The Netherlands.

Findeisen, R., Imsland, L., Allgöwer, F., & B.A., F. (2003). State and output feedback nonlinear model predictive control: An overwiev. *European Journal of Control*, 9(2-3),

921–936.

Fossen, T. I. (1994). *Guidance and Control of Ocean Vehicles*. Wiley Publishing.

García, C., Prett, D., & M., M. (1989). Model predictive control: Theory and practice - A survey. *Automatica*, 25(3), 335–348.

Genceli, H. & Nikolaou, M. (1996). New approach to constrained predictive control with simultaneous model identification. *AIChE Journal*, 42(10), 2857–2868.

Goodwin, G. C. & Sin, K. S. (1984). *Adaptive Filtering Prediction and Control*. Prentice Hall.

Green, M. & Moore, J. B. (1986). Persistence of excitation in linear systems. *System and Control Letters*, 7, 351–360.

Hemerick, J. (1973). *Culture methods and growth measurements*, (pp. 250–260). Cambridge University Press, UK.

Hermann, R. & Krener, A. (1977). Nonlinear controllability and observability. *IEEE transactions on Automatic Control*, 22(5), 728–740.

Hovd, M. & Bitmead, R. R. (2004). Interaction between control and state estimation in nonlinear MPC. In *IFAC Symposium on Dynamics and Control of Process Systems.* Boston, Massachusetts.

Hovd, M. & Braatz, R. D. (2001). Handling state and output constraints in MPC using time-dependent weights. In *American Control Conference,* Arlington, Virginia.

Ismail, A., Dumont, G. A., & Backstrom, J. (2003). Dual adaptive control of paper coating. *IEEE transactions on control systems technology*, 11(3), 289–309.

Julier, S. & Uhlmann, J. (1996). *A General Method for Approximating Nonlinear Transformations of Probability Distributions*. Technical report, Robotics Research Group, Department of Engineering Science, University of Oxford.

Julier, S. & Uhlmann, J. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3), 401–422.

Kailtah, T. (1980). *Linear systems*. Englewood Cliffs, New Jersey: Prentice Hall.

Kall, P. & Wallace, S. W. (1994). *Stochastic Programming*. New York, New York: John Wiley & Sons.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*, 82(Series D), 35–45.

Kandepu, R., Foss, B. A., & Imsland, L. (2008). Applying the unscented Kalman filter for nonlinear state estimation. *Journal of process control*, 18, 753–768.

Khalil, H. K. (2002). *Nonlinear systems*. Prentice Hall, 3rd edition.

Kolås, S. (2008). *Estimation in nonlinear constrained systems with severe disturbances*. PhD thesis, Norwegian University of Science and Technlogy (NTNU).

Kolås, S., Foss, B. A., & Schei, T. S. (2008). State estimation IS the real challeng in nmpc. In *NMPC 2008, International Workshop on Assessment and Future Directions of Nonlinear Model Predictive Control.* Pavia, Italy.

Kothare, M. V., Balakrishnan, V., & Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10), 1361–1379.

Kumar, P. & Varaiya, P. (1986). *Stochastic Systems: estimation, identification and adaptive control*. Upper Saddle River, New Jersey: Prentice Hall.

Kvasnica, M. (2009). *Real-time Model Predictive Control via Multi-Parametric Programming: Theory and Tools.* Saarbruecken, Germany: VDM Verlag.

Li, D., Qian, F., & Peilin, F. (2005). Research on dual control. *ACTA Automatica Sinica*, 31(1), 32–42.

Maciejowski, J. M. (2002). *Predictive control with constraints*. Essex, England: Prentice Hall.

Marafioti, G., Bitmead, R. R., & Hovd, M. (2008a). Model predictive control with state dependent input weight: an application to underwater vehicles. In *17th IFAC World Congress* Seoul, South Korea.

Marafioti, G., Bitmead, R. R., & Hovd, M. (2010a). Persistently exciting model predictive control. *International Journal of Adaptive Control and Signal Processing*, (Submitted).

Marafioti, G., Bitmead, R. R., & Hovd, M. (2010b). Persistently exciting model predictive control using FIR models. In *International Conference Cybernetics and Informatics* Vyšná Boca, Slovak Republic.

Marafioti, G., Olaru, S., & Hovd, M. (2008b). State estimation in nonlinear model predictive control, unscented Kalman filter advantages. In *NMPC 2008, International Workshop on Assessment and Future Directions of Nonlinear Model Predictive Control.* Pavia, Italy.

Marafioti, G., Olaru, S., & Hovd, M. (2009a). *State Estimation in Nonlinear Model Predictive Control, Unscented Kalman Filter Advantages*, volume 384 of *Nonlinear Model Predictive Control Towards New Challenging Applications Series: Lecture Notes in Control and Information Sciences*. Springer Berlin/ Heidelberg.

Marafioti, G., Tebbani, S., Beauvois, D., Becerra, G., Isambert, A., & Hovd, M. (2009b). Unscented Kalman filter state and parameter estimation in a photobioreactor for microalgae production. In *ADCHEM 2009, International Symposium on Advanced Control of Chemical Processes.* Instanbul, Turkey.

Mareels, I. (1985). Sufficiency of excitation. *System and Control Letters*, 5, 159–163.

Marty, A., Cornet, J., Djelveh, G., Larroche, C., & Gros, G. (1995). A gas phase chromatography method for determination of low dissolved CO_2 concentration and/or CO_2 solubility in microbial culture media. *Biotechnology Technique*, 9(11), 787–792.

Mayne, D., Rawlings, J., Rao, C., & Scokaert, P. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36(36), 789–814.

Mayne, D. Q. (2000). *Nonlinear model predictive control: challenges and opportunities, in Nonlinear Model Predictive Control*, (pp. 23–44). Birkhauser Verlag.

Moore, J. B. (1982). Persistence of excitation in extended least squares. *IEEE Transactions on Automatic Control*, 28, 69–68.

Naidu, D. S. (2003). *Optimal Control Systems*. CRC Press.

Nocedal, J. & Wright, S. (2000). *Numerical Optimization*. Springer, 2nd edition.

Pluymers, B., Rossiter, J. A., Suykens, J. A. K., & DeMoor, B. (2005). A simple algorithm for robust MPC. In *16th IFAC World Congress* Prague, Czech Republic.

Propoi, A. (1963). Use of LP methods for synthesizing sampled-data automatic systems. *Automation and Remote Control*, 24, 837–844.

Qin, S. & Badgwell, T. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7), 733–764.

Rawlings, J. & Mayne, D. Q. (2009). *Model Predictive Control: Theory and Design*. Nob Hill Publishing, LLC.

Rawlings, J. & Muske, K. R. (1993). The stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, 38(10), 1512–1516.

Richalet, J., Raul, A., Testud, J. L., & Papon, J. (1978). Model predictive heuristic control: Applications to industrial processes. *Automatica*, 14, 413–428.

Rossiter, J. (2003). *Model-based predictive control: a practical approach*. New York, New York: CRC Press.

Rugh, W. & Shamma, J. (2000). Research on gain scheduling. *Automatica*, 36, 1401–1425.

Santos, A. & Bitmead, R. R. (1995). Nonlinear control for an autonomous underwater vehicle (AUV) preserving linear design capabilities. In *34th IEEE Conference on Decision and Control* New Orleans, Louisiana.

Scokaert, P. O. M. & Rawlings, J. B. (1999). Feasibility issues in linear model predictive control. *AIChE Journal*, 45(8), 1649–1659.

Shimizu, K. (1996). A tutorial review on bioprocess systems engineering. *Computers and Chemical Engineering*, 20, 915–941.

Simon, D. (2006). *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. John Wiley & Sons.

Skogestad, S. & Postlethwaite, I. (2005). *Multivariable Feedback Control. Analysis and Desing*. John Wiley & Sons.

Sotomayor, O. A. Z., Odloak, D., & Moro, L. F. L. (2009). Closed-loop model re-identification of processes under MPC with zone control. *Control Engineering Practice*, 17(5), 551–563.

Spivey, B. J., Hedengren, J. D., & Edgar, T. F. (2010). Constrained nonlinear estimation for industrial process fouling. *Industrial & Engineering Chemistry Research*, 49, 7824–7831.

Sternby, J. (1976). A simple dual control problem with an analytical solution. *IEEE transactions on Automatic Control*, 21(6), 840–844.

Sutton, G. & Bitmead, R. (1998). Experiences with model predictive control applied to a nonlinear constrained submarine. In *37th IEEE Conference on Decision and Control* Tampa, Florida.

Tøndel, P., Johansen, T. A., & Bemporad, A. (2003). An algorithm for multiparametric quadratic programming and explicit MPC solutions. *Automatica*, 39(3), 489–497.

Vada, J. (2000). *Prioritized Infeasibility Handling in Linear Model Predictive Control: Optimality and Efficienzy*. PhD thesis, Norwegian University of Science and Technlogy (NTNU).

Vada, J., Slupphaug, O., & A., F. B. (1999). Infeasibility handling in linear MPC subject to prioritized constraints. In *14th IFAC World Congress* Beijing, China.

Van Der Merwe, R. & Wan, E. A. (2001). The square-root unscented Kalman filter for state and parameter estimation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing.* Salt Lake City, Utah.

Van Der Merwe, R., Wan, E. A., & Julier, S. (2004). Sigma-point Kalman filters for nonlinear estimation and sensor-fusion: Applications to integrated navigation. In *AIAA Guidance, Navigation & Control Conference (GNC)* Providence, Rhode Island.

Wan, E. A. & Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. In *Adaptive Systems for Signal Processing, Communications, and Control*

Symposium (pp. 153–158). Lake Louise, Alta. Canada.

Wan, E. A. & Van Der Merwe, R. (2001). *Kalman Filtering and Neural Networks*, chapter 7, (pp. 221–280). Wiley Publishing.

Wittermark, B. (1995). Adaptive dual control methods: An overview. In *5th IFAC symposium on Adaptive Systems in Control and Signal Processing* Budapest, Hungary.

Wittermark, B. (2002). Adaptive dual control. In Control Systems, Robotics and Automation, Encyclopedia of Life Support Systems (EOLSS), Developed under the auspices of the UNESCO. [http://www.eolss.net].

Zhang, F. E. (2005). *The Schur Complement and Its Applications*. Springer.

Zhou, K., Doyle, J. C., & Glover, K. (1996). *Robust and Optimal Control*. Upper Saddle River, New Jersey: Prentice Hall.