

Continuous-time Model Predictive Control

Quan Truong

March 2007

Submitted in accordance with the requirements for the
degree of Master of Engineering

School of Electrical and Computer Engineering
RMIT University
Melbourne, Australia

Abstract

Model Predictive Control (MPC) refers to a class of algorithms that optimize the future behavior of the plant subject to operational constraints [46]. The merits of the class algorithms include its ability to handle imposed hard constraints on the system and perform on-line optimization.

This thesis investigates design and implementation of continuous time model predictive control using Laguerre polynomials and extends the design approaches proposed in [43] to include intermittent predictive control, as well as to include the case of the nonlinear predictive control.

In the Intermittent Predictive Control, the Laguerre functions are used to describe the control trajectories between two sample points to save the computational time and make the implementation feasible in the situation of the fast sampling of a dynamic system.

In the nonlinear predictive control, the Laguerre polynomials are used to describe the trajectories of the nonlinear control signals so that the receding horizon control principle are applied in the design with respect to the nonlinear system constraints.

In addition, the thesis reviews several Quadratic Programming methods and compares their performances in the implementation of the predictive control. The thesis also presents simulation results of predictive control of the autonomous underwater vehicle and the water tank.

Keywords: Model Predictive Control, Intermittent Model Predictive Control, Nonlinear Model Predictive Control, Quadratic Programming, Cost Functions, Laguerre Functions.

Declaration

I certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program; and, any editorial work, paid or unpaid, carried out by a third party is acknowledged.

Quan Truong

March 2007

Acknowledgements

I would like to thank all those who have assisted and supported me during the course of research. I would especially like to thank my supervisor, Dr. Liuping Wang for providing me with much valuable advices and guidance. Also thankyou to everyone I shared the office with over the past two years for creating such a great environment to work in and for providing some necessary distractions. To my all friends and family who have given me much needed encouragement, thankyou.

Contents

Abstract	i
Declaration	iii
Acknowledgements	v
List of Figures	xvii
List of Tables	xix
Table of Symbols	xxi
1 Introduction	1
1.1 Literature Review	1
1.1.1 Model Predictive Control	1
1.1.2 Nonlinear Model Predictive Control	3
1.1.3 Quadratic Programming	5
1.1.4 Intermittent Continuous-time Model Predictive Control	6
1.1.5 Autonomous Underwater Vehicle	7

1.2	Contributions	8
1.3	Thesis Outline	9
2	Model Predictive Control	11
2.1	Introduction	11
2.2	Design Principle	12
2.2.1	Orthonormal Expansion and Laguerre Functions	12
2.2.2	Prediction	13
2.2.3	Optimal Control Strategy	14
2.2.4	Unconstrained Solution	15
2.2.5	Constrained Solution	15
2.3	Case Study 1: Food Cooking Extruder Process	16
2.3.1	Investigation Procedure	16
2.3.2	Simulation Results	20
2.4	Case Study 2: Copolymerization Process	26
2.4.1	Copolymerization Process	26
2.4.2	Simulation Results	28
2.5	Summary	44
3	Review Quadratic Programming Methods	45
3.1	Introduction	45
3.2	Active Set Method	46
3.3	Primal Dual Interior Point Algorithm	47

3.4	Hildreth's Quadratic Programming Procedure	50
3.5	Shor's r-Algorithm	50
3.6	Case Studies	51
3.7	Summary	59
4	Intermittent Continuous-time MPC	61
4.1	Introduction	61
4.2	Intermittent Continuous-time Model Predictive Control	61
4.3	Design IMPC for Online Simulation	63
4.4	Case Studies: A Coupled Tanks Model	65
4.5	Simulation Studies	66
4.6	Simulation Results	67
4.7	Conclusions	76
5	Nonlinear Model Predictive Control	77
5.1	Introduction	77
5.2	NMPC Using Laguerre Functions	78
5.2.1	Predicted Plant Response	79
5.2.2	Disturbance Rejection	80
5.3	Case Studies for NMPC Using Laguerre Functions	81
5.4	NMPC Control Using Taylor's Series	93
5.4.1	Prediction	93
5.4.2	Optimal Control Strategy	94

5.5	Simulation Results for NMPC Using Taylor’s Series	96
5.6	Conclusion	99
6	Autonomous Underwater Vehicles	101
6.1	Introduction	101
6.2	Mathematical Modelling of AUVs	102
6.2.1	Vehicle Dynamics	102
6.3	Vehicle Kinematic	104
6.4	Forces and Moments Acting on AUVs	104
6.4.1	Hydrostatic Forces	104
6.4.2	Lift Forces	105
6.4.3	Drag Forces	106
6.4.4	Added Mass	106
6.4.5	Euler’s Method	106
6.5	Modelling Methodology	107
6.5.1	Equations of motion	107
6.5.2	External forces and moments acting on AUV	108
6.5.3	State space equations	109
6.6	Case Studies	110
6.6.1	Intermittent Continuous-time Model Predictive Control	110
6.6.2	Nonlinear Model Predictive Control	118
6.7	Conclusions	120

7 Conclusion	121
7.0.1 Conclusions	121
7.0.2 Future Works	122
A Appendix	123
A.1 Chapter 3: Intermittent Continuous-time MPC Design to On-line Simulation	123
A.2 Chapter 6: Autonomous Underwater Vehicles	127
A.2.1 Equations for Coefficients	127
A.2.2 Values of Coefficients	129
Bibliography	131
Index	138

List of Figures

2.1	The food cooking extruder machine	18
2.2	Variety foods made by the extruder	18
2.3	A schematic diagram of the food cooking extruder	18
2.4	Process output (top), control signal (middle) and its derivative (bottom) of the food cooking extruder ($N=3$, $T_p = 100$): solid- $p = 1/15$, dash-dashed- $p = 1/10$, and dash-dotted- $p = 1/20$	20
2.5	Process output (top), control signal (middle) and its derivative (bottom) of the food cooking extruder ($p=1/15$, $T_p = 100$): solid- $N = 5$, dash-dashed- $N = 3$, and dash-dotted- $N = 2$	21
2.6	Process output (top), control signal (middle) and its derivative (bottom) of the food cooking extruder ($N=3$, $p = 1/15$): solid- $T_p = 100$, dash-dashed- $T_p = 50$, and dash-dotted- $T_p = 150$	22
2.7	Food cooking extruder control with disturbances: solid- with disturbance, dash-dashed- without disturbance.	23
2.8	Increasing p to $1/15$ to reduce the effect of disturbance	24
2.9	Continuous-time MPC results with constraints of the food cooking extruder (solid line: constrained control; dash-dashed line: unconstrained control. Process output: top-left: y_1 top-right: y_2 . Control signal: middle-left: u_1 middle-right: u_2 . Derivative of control signal: bottom-left: \dot{u}_1 bottom-right: \dot{u}_2).	25

2.10	Closed-loop response for the derivative of the control inputs with different pole location tuning parameters p (solid- $p = 0.1$, dash-dashed- $p = 0.15$, and dash-dotted- $p = 0.075$).	28
2.11	Closed-loop response for control inputs with different pole location tuning parameters p (solid- $p = 0.1$, dash-dashed- $p = 0.15$, and dash-dotted- $p = 0.075$).	29
2.12	Closed-loop response for outputs with different pole location tuning parameters p (solid- $p = 0.1$, dash-dashed- $p = 0.15$, and dash-dotted- $p = 0.075$).	30
2.13	Closed-loop response for the derivative of the inputs with different prediction horizon parameters T_p (dash-dashed- $T_p = 350$, and dash-dotted- $T_p = 50$).	31
2.14	Closed-loop response for the inputs with different prediction horizon parameters T_p (dash-dashed- $T_p = 350$, and dash-dotted- $T_p = 50$).	32
2.15	Closed-loop response for the outputs with different prediction horizon parameters T_p (dash-dashed- $T_p = 350$, and dash-dotted- $T_p = 50$).	33
2.16	Disturbances are added into the inputs.	34
2.17	Copolymerization process control with disturbances: solid- with disturbance, dash-dashed- without disturbance).	35
2.18	Copolymerization process control with disturbances: solid- with disturbance, dash-dashed- without disturbance).	36
2.19	Copolymerization process control with disturbances: solid- with disturbance, dash-dashed- without disturbance).	37
2.20	Continuous-time MPC results with constraints on the derivative of inputs, where $\tau_i = 0.01$ and $i = 0$	38
2.21	Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0$	39
2.22	Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0$	40

2.23	Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0, 1$	41
2.24	Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0, 1$	42
2.25	Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0, 1$	43
3.1	Quadratic Programming methods used where $\tau_i = 0, i = 0$: Hildreth QP Procedure, Active Set method, and Primal Dual Interior Point method.	53
3.2	Quadratic Programming method used where $\tau_i = 0, i = 0$: Shor's r-Algorithm	54
3.3	Quadratic Programming methods where $\tau_i = 0, i = 0, 1, 2$: Active Set method and Hildreth's QP method are used	57
3.4	Quadratic Programming method where $\tau_i = 0, i = 0, 1, 2$: Interior Point Primal Dual method is used	58
4.1	Intermittent MPC.	64
4.2	Intermittent MPC on-line.	64
4.3	Water tank apparatus.	65
4.4	The water level control with slow and fast response speed. Parameters p change from $\frac{1}{30}, \frac{2}{30}$ to $\frac{10}{30}$	68
4.5	The water level control with slow and fast response speed. Parameters N change from 1, 3 to 9.	69
4.6	Continuous IMPC results with different number of ahead prediction. The derivative of control signal is predicted with 1, 3, and 9 steps ahead.	70
4.7	Continuous time IMPC results with different open loop time intervals. The open loop time intervals are 0.5, 1, 2, and 4 seconds. The sampling time interval is 0.02 second.	71

4.8	Continuous time IMPC results with different open loop time intervals. The open loop time interval is 4 seconds. The sampling time intervals are 0.005, 0.01, 0.02, and 0.1 seconds.	72
4.9	The continuous time IMPC with and without step input disturbances.	73
4.10	The water level control with higher observer gain. The parameters p change from $\frac{1}{15}$ to $\frac{10}{15}$.	74
4.11	Continuous time IMPC with constraints. Sampling time interval = 0.02 seconds and open-loop time interval is 2 seconds.	75
5.1	Constrained control using linear QP and nonlinear QP for linear output system ($p = 5$)	84
5.2	Constant disturbance at the inputs at $p = 6$	85
5.3	Constant disturbance at the inputs at $p = 15$	86
5.4	Constant disturbance at outputs at $p = 6$	87
5.5	Constant disturbance at outputs at $p = 15$	88
5.6	NMPC using Laguerre functions with constraints	90
5.7	Constrained control for SISO under unknown constant disturbance θ	92
5.8	NMPC using Taylor expansion: Input constrained problem.	97
5.9	NMPC using Taylor expansion: Output constrained problem.	98
6.1	Body fixed coordinates and earth fixed coordinates	102
6.2	(a)Rudder angle attack and (b) Stern angle attack	106
6.3	Influences of rudder and stern angles on the AUV's trajectory	112
6.4	Continuous time IMPC: Changing parameters N, T_p, p to tune the response speeds.	113

6.5	Continuous time IMPC: Changing sampling interval h	114
6.6	Continuous time IMPC: Investigating disturbances with higher observer gain.	115
6.7	Continuous time IMPC: Investigating disturbances with lower observer gain.	116
6.8	Continuous time Intermittent MPC with and without con- straints.	117
6.9	Nonlinear MPC to control the AUV	119

List of Tables

Table of Symbols

l	Number of system inputs in the orthogonal function
l_i	Orthogonal function
m	Number of system outputs
n	Number of system states
p	Pole location tuning parameters of the Model Predictive Control method or the search direction
q	Number of element of setpoint or input of the AUV
r	The setpoint of the MPC method
t_i	Current time
$u(t), u(k)$	System inputs in continuous and discrete time
u_h	High limit of the control signals.
u_l	Low limit of the control signals.
\dot{u}_h	High limit of the derivative of the control signals.
\dot{u}_l	Low limit of the derivative of the control signals.
$x(t), x(k)$	System states in continuous and discrete time
$y(t), y(k)$	System outputs in continuous and discrete time
λ	Lagrange multiplier.
η	Solution of the cost function
τ	Future time instant.
A, B, C, D	State-space system matrices
I	Unit matrix.
J	Cost function
L	Laguerre matrix
N	Tuning parameter of the Model Predictive Control method
T_p	Prediction horizon tuning parameter of the Model Predictive Control method

Abbreviations and Acronyms

AUV	Autonomous Underwater Vehicle
DMC	Dynamic Matrix Control
FSR	Finite Step Response
GPC	Generalized Predictive Control
IMPC	Intermittent Model Predictive Control
IPPP	Intermittent Predictive Pole Placement
LQR	Linear Quadratic Regulator
KF	Kalman Filter
KKT	Karush Kuhn Kucker
MIMO	Multi-Input Multi-Output
MPC	Model Predictive Control
NLP	Nonlinear Programming
NMPC	Nonlinear Model Predictive Control
OL	Open-Loop
OLIFO	Open-Loop Intermittent Feedback Optimal
PDIP	Primal Dual Interior Point
PID	Proportional Integral and Derivative
QP	Quadratic Programming
SI	Sampling Time Interval
SISO	Single-Input Single-Output

Chapter 1

Introduction

1.1 Literature Review

This thesis is related to several fields of control engineering. In broad terms the fields are: model predictive control; quadratic programming, intermittent model predictive control, nonlinear model predictive control, and autonomous underwater vehicle. The literature review reflects the work completed in recent years in each of these respective areas.

1.1.1 Model Predictive Control

Model predictive control, also known as moving horizon control or receding horizon control [5, 16, 21], is an optimal control based algorithm. The algorithm uses the control input signals by repeatedly solving online optimal control problem to optimize the future plant output to the reference points. At each sampling time, the optimization aims at minimizing the cost functions subject to constraints over a future horizon. The control sequence is then converted into the feedback sequence and the process is repeated when the new plant output is available. Several famous papers can be found at [12, 43, 45, 50, 55, 60, 84, 86, 87]. These papers provide a comprehensive introduction and survey to the theories and technologies associated to the model predictive control.

The model predictive control began in chemical, refinery and petrochemical process industries. This class of algorithm was developed towards the end

Chapter 1. Introduction

of the 1970's and continued on to the 1990's. During this time, it received high recognition and wide applications. The MPC reached its fullest potential was due to the design procedure and the ease to handle the process constraints [20, 35].

In the earlier formulation of the MPC, the step response model was applied to model the plant. The plants could be described as the process delay, gain, and response time. The popular implementations of the MPC were Dynamic Matrix Control (DMC) [13, 55] in 1979 and Generalized Predictive Control (GPC) [19, 33] in 1987. The objective of a DMC controller was to drive the output close to the set point. The finite step response (FSR) was used to model the process plant. This resulted a smaller computed input moves and a less aggressive output response. This algorithm was considered to be less effective in handling multi-variable plants. Other implementations of the MPC are model algorithmic control (MAC or sometimes known as IDCOM) [56], multi-step multi-variable adaptive control [21] and predictive function control [20].

The first MPC technique had been proposed in the discrete time MPC. Some of the famous discrete MPC methods can be found at [12, 52, 56, 61, 96]. The discrete time MPC and continuous time MPC apply the least squares solutions to calculate their prediction. The future plant in the discrete time case is calculated through direct iteration, while that is calculated through convolution in the continuous time case. For that reason, it is more complicated to develop the MPC technique in the continuous time because the computational costs are more expensive. However, the continuous time model is more appropriate in making use of the physical systems. It is also able to handle both linear and nonlinear problems. Therefore, it is more economic benefits of the continuous time MPC model when applied to the process industry.

One of the first continuous time MPC, known as Continuous-Time Generalised Predictive Control (CGPC), was proposed by Demirciogly and Gawthrop [33]. The prediction was obtained through a Taylor series expansion. The CGPC has similar properties to those of the discrete time Generalised Predictive Control. The recent works of the continuous time MPC methods included [9, 33, 43, 59, 69].

This thesis focuses on the continuous time model predictive control algorithm using orthonormal basic function. One of the advantages of this algorithm is that the control trajectory of this model is approximated by using a set of Laguerre functions. According to the works of Wiener Lee [102], the accuracy of the approximation depends on the choices of the scaling factors,

1.1 Literature Review

which affect to the response speed of the closed loop system. Therefore, the system performances can be easily *tuned* by the academics or engineers.

The MPC algorithm uses the state space approach, which contributes many important issues to the MPC model. A velocity form model is introduced into the MPC model. The formulation leads to the natural embedding of an integral action and a simplified form for an implementation of the MPC on the real plant. Moreover, this MPC model is easier to imposed both soft constraints and hard constraints. This aspect is very important to the process industry where the limits on the operation are inevitably present.

It is well known that most of the applications in the process industry operate under the constraints, which has impacted on the quality of the process inputs, plant outputs, and direct costs. Therefore, the most important feature of the MPC is the ability to incorporate the various process constraints on-line at each step without violating them. The Quadratic Programming (QP) methods are adopted into the MPC for solving the quadratic cost function. Some examples of papers, which discussed the combination of the QP methods and the MPC, present in the papers [10, 45, 46, 55].

This thesis focuses on the model predictive control algorithms using a velocity form model. It also makes use of the state space model so that it can handle multivariable control easily. The related papers of the state space form of the model predictive controller can be found at [32, 49].

1.1.2 Nonlinear Model Predictive Control

The linear model predictive control had been developed in 1970s. The limitation of the linear MPC is that it based on the linear system and is not able to handle the nonlinear system. The nonlinear model predictive control had received its recognition in the 1990's [16, 17]. The advantages of the NMPC are:

- using the nonlinear model for prediction.
- handling linear and nonlinear constraints (at inputs, outputs and states).
- optimizing nonlinear process online.
- based on the linear MPC.

Chapter 1. Introduction

In the literature, there are a number of the NMPC that have been proposed. In this section, the NMPC algorithms based on the state space model are focused. The idea of this algorithm is to utilize the well known state space model and the linear MPC technologies that currently in use. They are summarized as follows:

- **NMPC based on Dynamic Matrix Control DMC:** the nonlinear quadratic DMC proposed by Garcia [11]. This algorithm is used to compute the effect of the past manipulated variables on the predicted output. The nonlinear model is linearized at each sampling time so that it can be used to compute the manipulated variable values. The algorithm is improved by Peterson et al [92, 93]. The disturbance is added into the system and the effects of the nonlinearities in the prediction horizon are taken into account. The algorithm is extended by Gattu et al [28, 29]. The state estimation using Kalman filter gain is incorporating into the system at each sampling time. The new algorithm takes the advantages of the state space models and nonlinear input-output models. The algorithm inherits the properties of the filter theory such as disturbance rejection and offset free tracking. The quadratic DMC algorithm has been successfully tested on the Tennessee Eastman Challenge process [39, 40].
- **NMPC based on Newton type:** Another method of the NMPC proposed is based on Newton-type, which is found at [62, 99]. The method is to linearize the nonlinear state space model at the previous sampling time. The impulse or step response models are determined from the state space model which obtained by linearizing the nonlinear model. The new input sequence is computed. This algorithm is extended by introducing the trust region concepts to deal with the ill-condition.
- **NMPC based on Nonlinear programming techniques:** The third NMPC algorithm proposed is based on nonlinear programming (NLP) techniques for solving the optimization. The method is to use a simultaneous solution and an optimization strategy. The method also incorporates with the Successive Quadratic Programming method for solving the NLP problems [8, 57, 82].

There are other NMPC algorithms which are not based on state space model. They are NMPC used recursive state and parameter estimation, such as Extended Kalman filter [15, 40], Neural Network Models [37, 90], Fuzzy and multiple linear models [7, 91, 98], or Polynomial NARX models [30].

1.1.3 Quadratic Programming

Quadratic programming has been an important topic in mathematic, physic, and sciences. The first programming method was linear programming which seeks to optimize the linear functions given linear constraints in the 1940s [27]. In recent years, the quadratic programming has received many interests because of its key role in obtaining the optimal solution of the constrained control problem. The ability to solve the quadratic program efficiently is the most important. However, the large number of quadratic programming methods had been studied extensively in the literature [25, 64, 66, 77, 78, 83, 85, 88]. The purpose of this thesis is to review the literature and to select the best known methods for the MPC.

There are four types of methods devised for solving the constrained control problem. They are primal methods, dual methods, Lagrange methods, and penalty and barrier methods. Each quadratic programming method has its own advantages. In this thesis, four methods are discussed. They are Active Set method, Primal Dual Interior Point algorithm, Hildreth's Quadratic Programming Procedure, and Shor's r-Algorithm.

Active Set Methods

The active set QP method was suggested by Fletcher [77] in 1971. The method proves to be effective for solving QP problem if there are few constraints at the solution. This method can handle the equality constrained problem by introducing the Lagrange multiplier to the objective function. However, it is more difficult to obtain the feasible solution in case of inequality constrained problem. Even though the active set is not known in advance, it can be constructed during iteration. The objective function could be minimized with all active set constraints treated as equality, where as the inequality constraints are ignored.

Primal-Dual Interior Point Algorithm

The Primal Dual Interior Point algorithm was discussed by Karmarkar [63] in 1984. In the past decades, the algorithms have become one of the most important and useful algorithm. The algorithm generates the points that lie in the neighborhood of the central trajectory. In this algorithm, both primal and dual algorithms are used to obtain the optimal solution. The dual method is used to compute the estimation of the optimal primal variable; whereas the primal method is used to test for convergence. The primal-dual methods find the solution to the primal-dual linear programming pair by applying the modified Newton's method to the Karush Kuhn Kucker (KKT) conditions. The solutions of the primal dual programming, which satisfied the KKT conditions. The literature about this algorithm can be found at

Chapter 1. Introduction

[74, 85, 88, 76, 10, 27].

Hildreth's Quadratic Programming Procedure

This method is a part of the primal dual method. However, the method calculates the Lagrange multiplier directly rather than using the Mehrotra's predictor-corrector search direction. For that reason, it is easy for computer programming. The Hildreth's Quadratic Programming algorithm converges, and given a sufficient number of iteration, the algorithm than gives a reliable result. Due to its simplicity, the convergence rate is slow for the nonnegative Lagrange multiplier [14, 45].

Shor's r-Algorithm The Shor's r-Algorithm has developed for minimizing the smooth convex functions under unconstrained problems. It shows the advantage in the ability to handle both linear and nonlinear objective functions that subject to either linear or nonlinear constrained problems. However, the algorithm takes larger number of iteration to obtain the feasible solutions because of having serious problems with the design of stopping criterion and the choice of the initial step size [25, 64, 65].

1.1.4 Intermittent Continuous-time Model Predictive Control

The continuous time MPC methods successfully predict the future plant. The advantages of these methods are allowed to tune the scaling factors, poles and other parameters so that the responses of the system can reach to the future plant as quickly as possible. Therefore, the continuous time MPC has been driven by many applications in the process industry. However, the processes in those applications have such a long time constant (up to hours) and slow sample interval (few seconds). For the processes that have short time constant and slow sampling interval, the prediction of the continuous time MPC is not assured.

The new approach, which based on the ideas of Model Predictive Control and Intermittent Control, was proposed so that it would be able to work well under these strict conditions. This approach is called an Intermittent Model Predictive Control (IMPC). The IMPC approach is laid between the extremes of purely discrete time and continuous time approaches. In the intermittent control, the continuous time open loop control signal is re-adjusted intermittently. The recent works on this method have been proposed by Gawthrop [70, 71, 73] and Ronco [22, 23, 24].

According to Gawthrop ([70, 73]), the optimization of the intermittent con-

1.1 Literature Review

control is done by using an intermittent feedback approach whereby the open loop trajectory is implemented between intermittent closing of the loop by taking a state measurement and recomputing the open loop control. The intermittent control is applied for the predictive pole placement control to form a method called intermittent predictive pole placement control (IPPP). This method successfully drives the inverted pendulum system which is complicated and unstable.

Another method proposed by Ronco ([22, 23, 24]) is called Open-Loop Intermittent Feedback Optimal Control (OLIFO) where the control method is developed based on the practical continuous time generalised predictive controller. During simulation, the optimised results are not available until the optimisation is completed at time t_i . Thus, the feedback is intermittently introduced for every open loop time interval Δ_{ol} . In other words, the system is updated for every Δ_{ol} . The control signals are calculated based on 2 sections. The first section is in the range that the optimisation is completed ($t > \Delta_{ol}$). In the second section ($t \leq \Delta_{ol}$), no information is available. Therefore, the control signals are repeated the optimised control signal in the previous section which is the section of time t_{i-1} . This method has some drawbacks if there are disturbances in the system. In addition, the ways of constructing control signals produce the fluctuations in the control inputs themselves and also on the process outputs. This approach is also successfully applied to control six muscles human arm model, two links robot arm, and ball and beam system [22, 31, 33].

1.1.5 Autonomous Underwater Vehicle

Autonomous Underwater Vehicles (AUVs) have been developed for over three decades to explore the ocean. They are very useful since not requiring a tethered cable and travelling far from homebase without restriction. Moreover, the AUV can work independent of human control. For these reasons, they have been used in variety of long range oceanography survey application, autonomous docking and shallow water mine countermeasure [34, 68, 100]

Tracking the AUV in the ocean is a challenge while it is influenced by nonlinear forces and moments. The equations of motion of the AUV has been established by Abokowitz of the MIT University in 1969 [58]. The equations comprised six degree of freedom of the AUV. From that, many researches have been done by the Naval Postgraduate School to improve the performance of the vehicle. The original equations of Abokowitz has been linearized so that they could be used for tracking the vehicle in one direction

Chapter 1. Introduction

by using Least Mean Square Estimation method [54].

1.2 Contributions

The main contributions of this thesis are summarized as follows:

- **Intermittent Model Predictive Control**

The new algorithm Intermittent Model Predictive Control design using Laguerre functions was derived based on the ideas of Continuous Time Model Predictive Control method and Intermittent Control. This algorithm is more efficient than the traditional model predictive control while dealing with the applications that have the short sampling interval. The algorithm retains the key success features of the continuous time MPC such as easy to tune, reject disturbance and constrained control. It also overcomes the drawbacks of the fluctuations in the control inputs and process outputs.

The Intermittent Model Predictive Control is programmed for simulation purpose and also for estimation on-line by using Embedded function in Matlab Simulink.

- **Nonlinear Model Predictive Control**

Nonlinear model predictive control design is used where the applications of nonlinear optimization methods are needed. In this thesis, two new control methods were derived to handle the multiple inputs and multiple outputs nonlinear systems. The first nonlinear MPC design uses the Taylor series expansion and the second one uses the Lie's function.

- **Reviewing Quadratic Programming Methods**

The original work of MPC design using Laguerre functions is improved on the constrained control. The future instant time is added to the constrained set and enlarge the set. The idea is to make the prediction of the MPC under the constrained problem to operate accurately.

The important feature of the MPC algorithms is the ability to handle imposed hard constraints on the system. The Quadratic Programming (QP) methods are capable of optimizing a class of quadratic forms within linear inequality constraints. However, in case of complicated process dynamics and fast sampling, it is a challenge to obtain a reliable solution. This thesis presents simple, efficient, and widely

1.3 Thesis Outline

used QP methods. Depending on specific type of constraints, the appropriate QP method will be suggested.

The formula of the constraints on the inputs, outputs and states are constructed for the continuous time model predictive control, intermittent model predictive control and nonlinear predictive control. The size of the constraints depends on the future time instants, which is used to limit the response.

The use of Laguerre polynomials for parameterising the variables in the quadratic programming problem was proposed for the Intermittent MPC. A significant reduction in the number of parameters offered a reduction in the computational cost of locating the optimal set of future control variables.

- **Autonomous Underwater Vehicles**

This thesis revisited the AUV's equation in six degree of freedom. The equation would be reconstructed into a state space model with multiple inputs and multiple outputs. The Intermittent Model Predictive control was applied to control the directions of the vehicle. The model was also kept as high nonlinear. The nonlinear model predictive control was used to control the turn and the dive direction of the vehicle.

The thesis comprises many industrial applications and case studies. It shows how to formulate those systems to the state space models or nonlinear models for the use of the model predictive control.

The following publication was also produced as a result of this research.

Q. Truong, L. Wang, and P. Gawthrop, *Intermittent Model Predictive Control of an Autonomous Underwater Vehicle*, 9th International Conference on Control, Automation, Robotics and Vision (ICCARV), 2006.

1.3 Thesis Outline

The thesis consists of seven chapters. The basic overall structure is the development of model predictive control algorithms by presenting each of the components individually before showing how they work in combination.

Chapter 2 revisits the concept of the model predictive control using Laguerre functions. The key idea and properties of the Laguerre functions

Chapter 1. Introduction

are briefly introduced. These functions are utilized in the predicting future plant outputs. The cost functions and the constraints of the controller are constructed based on the Laguerre functions. The optimal solution of the MPC under constrained and unconstrained problems are proposed. The roles of the tuning parameters are investigated. The simulation examples using Matlab programming platform present clearly all properties of the MPC.

Chapter 3 summarizes the Quadratic Programming methods which is used to obtain the optimal solutions of the cost functions. They are Primal-Dual Interior Point method using Mehrotra's predictor-corrector algorithm, Hildreth's Quadratic Programming Procedure, Active Set method, and Shor's r-Algorithm. The simulation studies on the properties of these methods are investigated. The advantages and drawbacks of each method are discussed.

Chapter 4 proposes the new algorithm known as Intermittent Model Predictive Control (IMPC). The new algorithm is accomplished by using an intermittent feedback approach whereby the open loop trajectory is implemented between closing of the loop by taking a state measurement and recomputing the open loop control. The IMPC is designed for predicting on-line.

Chapter 5 proposes the two new approaches of Nonlinear Model Predictive Control (NMPC). The first approach utilizes the properties of Taylor series in rounding of error. The second approach uses the Lie's function for prediction. The cost functions and the nonlinear constraints are constructed.

Chapter 6 revisits the construction of the Autonomous Underwater Vehicle in six degree of freedom. The model is highly nonlinear and comprises all forces and moments of the ocean, which affect on the vehicle. The nonlinear model is applied to the nonlinear model predictive control. The model is improved in order to apply to the Intermittent Model Predictive Control. All of the properties of the IMPC such as constrained control, tuning parameters, and disturbance rejection are investigated.

Finally, **Chapter 7** provides a summary of this work and discusses possible further research in this area.

Chapter 2

Model Predictive Control

2.1 Introduction

The design principle of the advanced control strategy known as Model Predictive Control is introduced. The controller utilizes the properties of the Laguerre functions to reduce the cost of computation. The merits of the MPC algorithm include its ability to obtain prediction, to make use of the well known state space model, and to handle imposed hard constraints in the system. This section also derives the solutions of the cost function under constrained and unconstrained problems. The optimization algorithm is described to obtain the optimal solution.

This thesis is also to contribute the improvement of the prediction of the MPC under constrained control problem. The future instant time is added to the constrained set. The constrained set is much larger and the computation is more expensive.

Two industrial applications, which are controlled by the MPC methods, are introduced. In the first application in section 2.3.2, the MPC is applied to control the variables of the food cooking extruder which produced by the food cooking extruder. The system has two inputs and two outputs. The Copolymerization process is a second application and is more complex. It is described in section 2.4.2. The system model has five inputs and four outputs. The MPC properties, such as tuning parameters, disturbance rejection, and constrained control, are investigated. Section 2.5 states the conclusion of the MPC design.

Chapter 2. Model Predictive Control

2.2 Design Principle

2.2.1 Orthonormal Expansion and Laguerre Functions

An arbitrary function $f(t)$ is expressed in terms of a series expansion as:

$$f(t) = \sum_{i=0}^n \xi_i l_i(t) \quad (2.1)$$

where ξ_i , $i = 1, 2, \dots, n$, and $l_i(t)$, $i = 1, 2, \dots, n$, are orthogonal functions if satisfying:

$$\int_0^n l_i(t)^2 dt = 1 \quad \int_0^n l_i(t) l_j(t) dt = 0, \quad i \neq j \quad (2.2)$$

Assuming that:

$$\int_0^n f(t)^2 dt < \infty \quad (2.3)$$

and for any $\epsilon > 0$, there exists an integer N such that for all $k \geq N$:

$$\int_0^n (f(t) - \sum_{i=1}^k \xi_i l_i(t))^2 dt < \epsilon \quad (2.4)$$

The truncated expansion $\sum_{i=1}^n \xi_i l_i(t)$ is used to closely approximate $f(t)$. The system that satisfying (2.3) can be approximated by using orthonormal functions. The orthonormal function frequently used is the set of Laguerre functions.

$$\int_0^n l_i(t) e^{-st} dt = \sqrt{2p} \frac{(s-p)^{i-1}}{(s+p)^i} \quad (2.5)$$

where p is a positive number. Let $L(t) = [l_1(t) \ l_2(t) \ \dots \ l_N(t)]^T$ and $L(0) = \sqrt{2p}[1 \ 1 \ \dots \ 1]^T$. The Laguerre functions satisfy the differential equation:

$$\dot{L}(t) = A_p L(t) \quad (2.6)$$

where:

$$A_p = \begin{bmatrix} -p & 0 & \dots & 0 \\ -2p & -p & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -2p & -2p & \dots & -p \end{bmatrix} \quad (2.7)$$

The solution of (2.6) is:

$$L(t) = e^{A_p t} L(0) \quad (2.8)$$

2.2 Design Principle

2.2.2 Prediction

One of the advantages of the MPC is to utilize the state space model which is understood by many practitioners. The linear dynamic model is usually described as follows [43]:

$$\begin{aligned}\dot{X}_m(t) &= A_m X_m(t) + B_m u(t) \\ y(t) &= C_m X_m(t)\end{aligned}\quad (2.9)$$

where $X_m(t)$ is the state vector of dimension n . We are more interested in the rate of the control input signal $\dot{u}(t)$ which is represented by Laguerre functions to obtain

$$\dot{u}(t) = L^T \eta \quad (2.10)$$

By taking derivative both sides of (2.9), the state space model can be rewritten in the augmented model and still keeps the same properties.

$$\begin{aligned}\dot{X}(t) &= AX(t) + B\dot{u}(t) \\ y(t) &= CX(t)\end{aligned}\quad (2.11)$$

where:

$$X(t) = [Z(t)^T \ y(t)^T]^T, \ B = [B_m^T \ 0]^T, \ C = [0 \ I], \ A = \begin{bmatrix} A_m & 0 \\ C_m & 0 \end{bmatrix}, \ Z(t) = \dot{X}_m(t).$$

Assume that at the current time, say t_i , the state variable $X(t_i)$ is available. Then at the future time $t_i + \tau$, ($\tau > 0$), the predicted state variable $X(t_i + \tau)$ is described as the following equations:

$$\begin{aligned}X(t_i + \tau) &= e^{A\tau} X(t_i) + \int_0^\tau e^{A(\tau-\gamma)} B \dot{u}(t_i + \gamma) d\gamma \\ y(t_i + \tau) &= CX(t_i + \tau)\end{aligned}\quad (2.12)$$

The control signal is written as:

$$\dot{u}(t) = [\dot{u}_1(t) \ \dot{u}_2(t) \ \dots \ \dot{u}_r(t)]$$

and the input matrix is partitioned as:

$$B = [B_1 \ B_2 \ \dots \ B_r]$$

where B_i is the i^{th} column of B matrix. We express the i^{th} control signal $\dot{u}_i(t)$ ($i = 1, 2, \dots, r$).

$$\dot{u}_i(t) = L_i(t)^T \eta_i$$

where $L_i(t)^T = [l_1^i(t) \ l_2^i(t) \ \dots \ l_{N_i}^i(t)]$ and $\eta_i(t)^T = [\eta_1^i(t) \ \eta_2^i(t) \ \dots \ \eta_{N_i}^i(t)]$. The parameters p_i and N_i are pre-chosen. The predicted future state at

Chapter 2. Model Predictive Control

the time $t_i + \tau$ is:

$$X(t_i + \tau) = e^{A\tau}X(t_i) + \int_0^\tau e^{A(\tau-\gamma)}[B_1L_1(\gamma)^T \ B_2L_2(\gamma)^T \dots \ B_rL_r(\gamma)^T]d\gamma\eta$$

$$y(t_i + \tau) = CX(t_i + \tau)$$
(2.13)

Let us define:

$$I_{int}(\tau)^i = \int_0^\tau e^{A(\tau-\gamma)}[B_1L_1(\gamma)^T \ B_2L_2(\gamma)^T \dots \ B_rL_r(\gamma)^T]d\gamma$$
(2.14)

where $I_{int}(\tau)^i$ is $(n + q) \times N_i$ matrix and is calculated as follows:

$$AI_{int}(\tau) - I_{int}(\tau)A_p^T = -BL(\tau)^T + e^{A\tau}BL(0)^T$$
(2.15)

By letting I_i as the i^{th} column of I_{int} and J_i as the i^{th} column on the left hand side of equation (2.15), the elements in the matrix I_{int} can be determined by taking advantage of the lower triangular structure of the matrix A_p :

$$(A + pI)I_1 = J_1$$

For $i = 2, \dots, N$

$$(A + pI)I_i = J_i - 2p \sum_{k=1}^{i-1} I_k$$
(2.16)

2.2.3 Optimal Control Strategy

We say that the set of the future setpoint $r(t_i + \tau) = [r_1(t_i + \tau) \ r_1(t_i + \tau) \dots \ r_q(t_i + \tau)]$, $0 \leq \tau \leq T_p$ are available where T_p is the prediction horizon. The purpose of the MPC is to drive the predicted plant output as close as possible to the future trajectory of the setpoint. Therefore, the problem becomes to minimize the cost function.

$$J = \int_0^{T_p} [r(t_i + \tau) - y(t_i + \tau)]^T Q [r(t_i + \tau) - y(t_i + \tau)] d\tau + \int_0^{T_p} \dot{u}^T(\tau) R \dot{u}(\tau) d\tau$$
(2.17)

Q and R are symmetric matrices with $Q > 0$ and $R > 0$. For simplicity, Q can be set to the unit matrix I and R is a diagonal matrix with

$$R = \text{diag}(\lambda_k), \quad 0 \leq k \leq r$$

From the property of orthonormal equation (2.3), the second term of (2.17) can be represented in term of η

$$\int_0^{T_p} \dot{u}_k(\tau)^T \dot{u}_k(\tau) d\tau = \eta_k^T \eta_k$$

2.2 Design Principle

thus the cost function J can be rewritten as:

$$J = \int_0^{T_p} [r(t_i + \tau) - y(t_i + \tau)]^T Q [r(t_i + \tau) - y(t_i + \tau)] d\tau + \eta^T \bar{R} \eta \quad (2.18)$$

where $\bar{R} = \text{diag}(R_i)$, and $R_i = \lambda_i I_{N_i \times N_i}$, $I_{N_i \times N_i}$ is a unit matrix with dimension N_i .

2.2.4 Unconstrained Solution

To find the optimal solution, the cost function (2.17) can be formulated as a function of the parameter η by substituting $y(t_i + \tau)$ in (2.12) with its predictions:

$$J = \eta^T \Pi \eta - 2\eta^T [\Psi_1 r(t_i) - \Psi_2 X(t_i)] + \int_0^{T_p} \dot{u}^T(\tau) R \dot{u}(\tau) d\tau \quad (2.19)$$

Π , Ψ_1 , and Ψ_2 are functions that depend on τ and independent on η :

$$\begin{aligned} \Pi &= \int_0^{T_p} \Phi(\tau) Q \Phi(\tau)^T d\tau + \bar{R} \\ \Psi_1 &= \int_0^{T_p} \Phi(\tau) Q d\tau, & \Psi_2 &= \int_0^{T_p} \Phi(\tau) Q C e^{A\tau} d\tau \\ \Phi(\tau) &= (C [I_{int}^1 \ I_{int}^2 \ \dots \ I_{int}^r])^T \\ w(t_i + \tau) &= r(t_i + \tau) - C e^{A\tau} X(t_i) \end{aligned}$$

The analytical solution to minimize (2.19) is:

$$\eta = \Pi^{-1} (\Psi_1 r(t_i) - \Psi_2 X(t_i)) \quad (2.20)$$

2.2.5 Constrained Solution

Handling the constraints in the MPC design is an important issue. It is generally true that higher levels of performance are associated with operating on constrained boundary. Therefore, ignoring the constraints will incur the performance penalty. The problem now is how to formulate the various constraints into inequality linear equations. The constraint problems are considered at the inputs, outputs and states. The input and output constraints are frequently used. In this thesis, we introduce hard constraints on the control input variable and its first derivative. It is because the input constraints are most frequently encountered and can be imposed naturally

Chapter 2. Model Predictive Control

through the physical system. Whereas, the output constraints can result in infeasibility problems and the constraints can be violated because the model plants to mismatch. The manipulated variable constraints and rate constraints are specified as:

$$\dot{u}_{low}(t_i + \tau) \leq \dot{u}(t_i + \tau) = L(\gamma)^T \eta \leq \dot{u}_{high}(t_i + \tau) \quad (2.21)$$

$$u_{low}(t_i + \tau) \leq u(t_i + \tau) = \int_0^{\tau_i} L(\gamma)^T d\gamma \eta + u(t_i - \Delta t) \leq u_{high}(t_i + \tau) \quad (2.22)$$

After re-arranging the constraints, the model predictive control problem with respect to hard constraints is finally proposed as:

Minimize:

$$J = \eta^T \Pi \eta - 2\eta^T [\Psi_1 r(t_i) - \Psi_2 X(t_i)] + \int_0^{T_p} \dot{u}^T(\tau) R \dot{u}(\tau) d\tau \quad (2.23)$$

Subject to:

$$\begin{bmatrix} L(\tau_i) \\ -L(\tau_i) \\ \int_0^{\tau_i} L_k(\gamma) d\gamma \\ -\int_0^{\tau_i} L_k(\gamma) d\gamma \end{bmatrix} \eta \leq \begin{bmatrix} \dot{u}_{high}(t_i + \tau_i) \\ -\dot{u}_{low}(t_i + \tau_i) \\ u_{high}(t_i + \tau_i) - u(t_i - \Delta t) \\ -u_{low}(t_i + \tau_i) - u(t_i - \Delta t) \end{bmatrix} \quad (2.24)$$

In the next section, two industrial applications are introduced to investigate the properties of the Model Predictive Control approaches. They are food cooking extruder process and copolymerization process.

2.3 Case Study 1: Food Cooking Extruder Process

In this section, the twin screw food cooking extruder process is used to receive further understanding of the MPC approach and its properties. The process has two inputs and two outputs.

2.3.1 Investigation Procedure

The twin screw food cooking extruder system was used for the case study. The relevant papers, which discussed and implemented on this extruder, were performed by Wang and her colleagues [41, 46, 89]. The extruder machine and its schematic diagram are shown in figures 2.1, and 2.3.

2.3 Case Study 1: Food Cooking Extruder Process

The food cooking extrusion has been developed into one of the most efficient process in the food industry. The extruder has been used to make variety of products (see figure 2.2), such as cereals, biscuits, pasta, and pet foods. The ingredients are forced through a barrel by a rotating screw. They are transported, mixed, shaped, stretched, and sheared under elevated temperature and pressure before getting out through the narrow die opening. The extrusion process is needed to be controlled to satisfy a certain product quality. The process can be adjusted on-line by the screw speed u_1 , and liquid pump speed u_2 . The measured outputs of the process are the specific mechanical energy y_1 (SME is the ratio of the motor power and the mass of the output) and motor torque y_2 [6, 45]. The multi-frequency relay feedback control system [42] was implemented on the extruder to ensure a safe operation of the process. The continuous-time transfer function model, which is estimated using the state variable filter approach [44], is presented as follows:

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{0.193278}{s^2+0.44745s+0.273987} & \frac{-0.005441}{s^2+0.622671s+0.195854} \\ \frac{0.000976s-0.000226}{s^3+0.422036s^2+0.091833s+0.003434} & \frac{-0.000017}{s^2+0.060324s+0.006836} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} \quad (2.25)$$

This state space model is used to demonstrate the controlling purpose in the model predictive control approach. The extrusion system has two inputs and two outputs. The unit setpoints are set to both outputs. The behavior of the inputs is to lead the outputs to the setpoints are shown in figure 2.4 . For further investigation, there are three tuning factors needed to be considered under unconstrained control problem. They are a pole placement parameter p , a prediction horizon T_p , and parameter N , which used in Laguerre model to capture the control signal.

In the first case, the parameters T_p and N are kept constants at ($T_{p1} = T_{p2} = 150$) and ($N_1 = N_2 = 3$). The pole placement parameters $p_1 = p_2$ vary from $1/20$, $1/15$, to $1/10$. The corresponding observer poles are respectively at:

- Case 1: [-0.1 -0.6 -1.1 -1.6 -2.1 -2.6 -3.1 -3.6 -4.1 -4.6 -5.1 -5.6 -6.1 -6.6].
- Case 2: [-0.1333 -0.6333 -1.1333 -1.6333 -2.1333 -2.6333 -3.1333 -3.6333 -4.1333 -4.6333 -5.1333 -5.6333 -6.1333 -6.6333].
- Case 3: [-0.2 -0.7 -1.2 -1.7 -2.2 -2.7 -3.2 -3.7 -4.2 -4.7 -5.2 -5.7 -6.2 -6.7].

Chapter 2. Model Predictive Control



Figure 2.1: The food cooking extruder machine



Figure 2.2: Variety foods made by the extruder

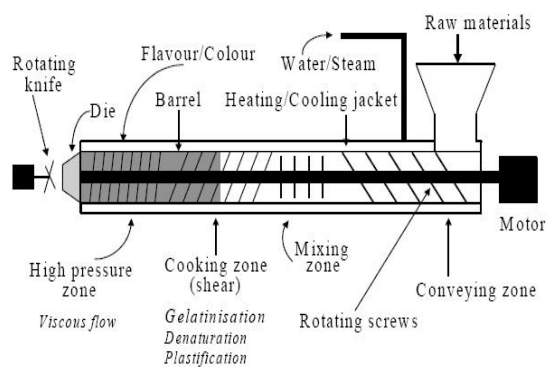


Figure 2.3: A schematic diagram of the food cooking extruder

2.3 Case Study 1: Food Cooking Extruder Process

The comparison of the control signal responses is shown in figure 2.4.

Similarly, the pole placement parameters and the prediction horizon are chosen as $(p_1 = p_2 = 1/15)$ and $(T_{p1} = T_{p2} = 150)$, respectively. The parameters $(N_1 = N_2)$ are increased from 2, 3 to 5. The figure 2.5 depicts the input and output responses versus the varying of N. These figures show the evident to prove that the closed-loop responses are faster while increasing pole placement parameters p or parameters N. While keeping p and N as constants which are 1/15 and 3, respectively, the predictive horizon $T_{p1} = T_{p2}$ are changing from 50, 100, and 150. It can be seen in the figure 2.6 that the increasing of the parameters T_p slow down the speed of the response.

The disturbance, which affected on the system performance, is also investigated. The pulse disturbance with 0.01 in magnitudes are added to the system at the time $t = 100$ to 150 sec. The figure 2.8 shows a comparison of the results in the case that the system has and does not have disturbance. The pole placement parameters are chosen as $p = 1/50$. The disturbance affects to both input and output responses. However, the MPC method still leads the outputs to the setpoint. To reduce the effect of the disturbance, the tuning parameter p can be increased to reduce the observer gain. The figure 2.7 shows the reduce significantly of the disturbance comparing to the figure 2.8 while increasing p to 1/15.

The constrained continuous-time model predictive control is applied to the multivariable system. The constrained equations are based on (2.24) with $\tau_i = 0$ and are chosen as in (2.26). This system has eight linear inequality constraint equations. The tuning parameters are: $p_1 = p_2 = 0.1$, $T_{p1} = T_{p2} = 150$, and $N_1 = N_2 = 3$.

$$\begin{aligned}
 -0.001 &\leq \dot{u}_1 \leq 0.01 \\
 -0.005 &\leq \dot{u}_2 \leq 0.002 \\
 -0.003 &\leq u_1 \leq 0.079 \\
 -0.02 &\leq u_2 \leq 0.002
 \end{aligned} \tag{2.26}$$

The figure 2.9 compares the response of the system with and without constrained control problem. In this case, the constraints imposed on the amplitudes of the control input and its derivative have a little impact on the system outputs.

Chapter 2. Model Predictive Control

2.3.2 Simulation Results

Case 1: Effects of the pole location tuning parameters \mathbf{p}

The purpose of the first simulation is to investigate the effect of the pole location parameters \mathbf{p} on the system responses. The prediction horizon parameters T_p and tuning parameters N are kept constant.

$$T_{p1} = T_{p2} = 150, \quad N_1 = N_2 = 3$$

The pole placement parameters \mathbf{p} vary from slow response speeds to the fast response speed. The corresponding parameters \mathbf{p} are chosen as:

$$\text{Case1} : p_1 = p_2 = 1/20$$

$$\text{Case2} : p_1 = p_2 = 1/15$$

$$\text{Case3} : p_1 = p_2 = 1/10$$

Observation: The response speed in case 1 is slow. The response speed in case 2 is normal. The response speed in case 3 is fast.

Conclusion: Closed-loop responses are faster while increasing the pole placement parameters \mathbf{p} .

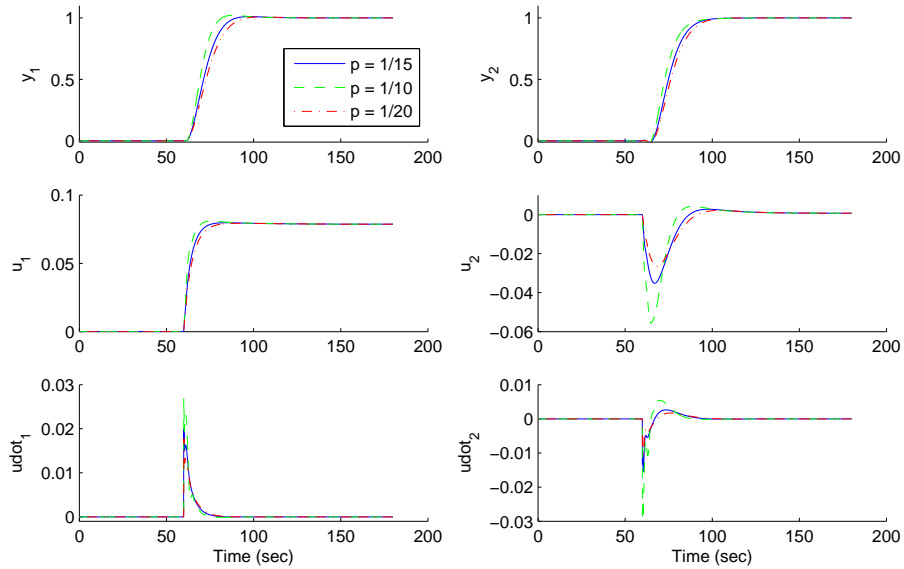


Figure 2.4: Process output (top), control signal (middle) and its derivative (bottom) of the food cooking extruder ($N=3$, $T_p = 100$): solid- $p = 1/15$, dash-dashed- $p = 1/10$, and dash-dotted- $p = 1/20$.

2.3 Case Study 1: Food Cooking Extruder Process

Case 2: Effects of the tuning parameters N

The purpose of this simulation is to investigate the effect of the tuning parameters N on the system responses. The tuning parameters T_p and p are kept constant.

$$T_{p1} = T_{p2} = 150, \quad p_1 = p_2 = 1/15$$

The parameters N are chosen as the following:

$$\text{Case1} : N_1 = N_2 = 2$$

$$\text{Case2} : N_1 = N_2 = 3$$

$$\text{Case3} : N_1 = N_2 = 5$$

Observation: The response speed in case 1 is slow. The response speed in case 2 is normal. The response speed in case 3 is fast.

Conclusion: Closed-loop responses are faster while increasing the tuning parameters N.

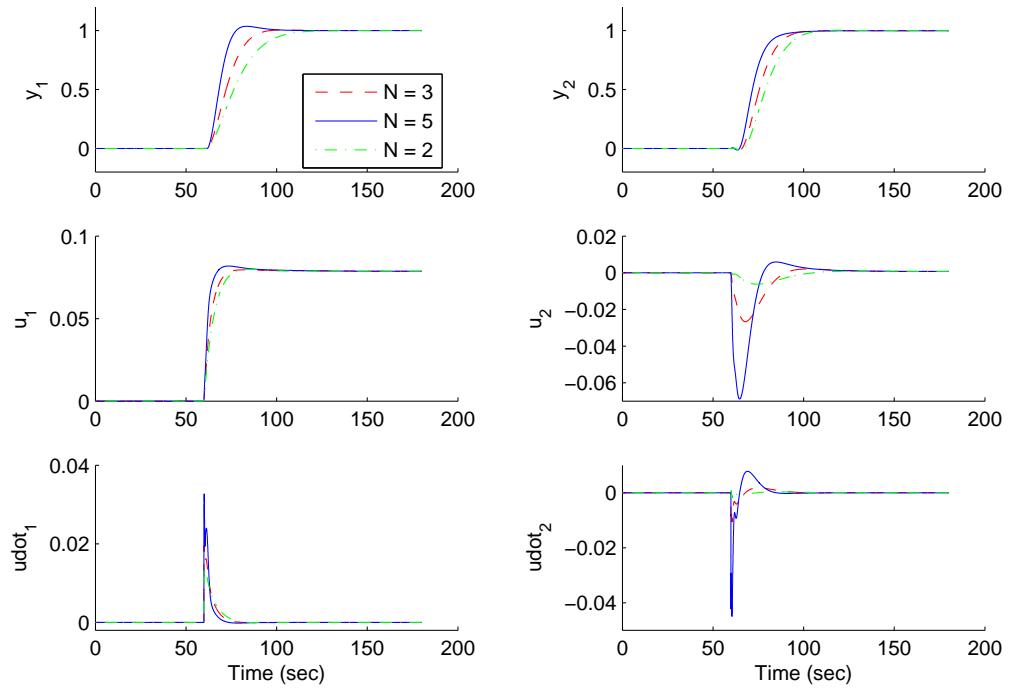


Figure 2.5: Process output (top), control signal (middle) and its derivative (bottom) of the food cooking extruder ($p=1/15$, $T_p = 100$): solid- N = 5, dash-dashed- N = 3, and dash-dotted- N = 2.

Chapter 2. Model Predictive Control

Case 3: Effects of the prediction horizon parameters T_p

The purpose of the simulation is to investigate the effects of the prediction horizon parameters T_p on the system responses. The other parameters N and p are kept constant.

$$N_1 = N_2 = 3, \quad p_1 = p_2 = 1/15$$

The parameters N are chosen as the following:

$$\text{Case1} : T_{p1} = T_{p2} = 50$$

$$\text{Case2} : T_{p1} = T_{p2} = 100$$

$$\text{Case3} : T_{p1} = T_{p2} = 150$$

Observation: The response speed in case 1 is fast. The response speed in case 2 is normal. The response speed in case 3 is slow.

Conclusion: Closed-loop responses are faster while decreasing the tuning parameters N .

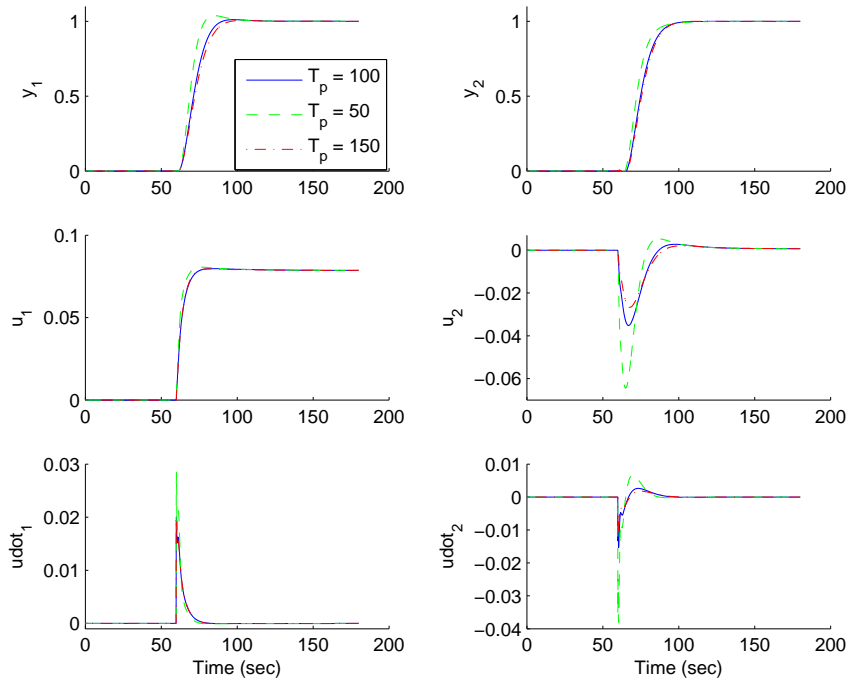


Figure 2.6: Process output (top), control signal (middle) and its derivative (bottom) of the food cooking extruder ($N=3$, $p = 1/15$): solid- $T_p = 100$, dash-dotted- $T_p = 50$, and dashed- $T_p = 150$.

2.3 Case Study 1: Food Cooking Extruder Process

Case 4: Disturbance rejection

The purpose of this simulation is to investigate how the MPC approach handles the disturbance of the system. This process is known as *disturbance rejection*.

The high disturbances with 0.01 in magnitudes are added to the input of the system from the time $t = 100$ to 150 seconds. The tuning parameters $T_{p1} = T_{p2} = 150$, $N_1 = N_2 = 3$, and $p_1 = p_2 = 1/50$.

Observation: The system responses (\dot{u} , u , y) are strongly impacted. The outputs still reach to the setpoints when the disturbances occur ($t = 100$ - 150 seconds) and when the disturbances are no more influence ($t \geq 150$ second).

Conclusion: The MPC approach can handle well the process when the disturbances are in the system.

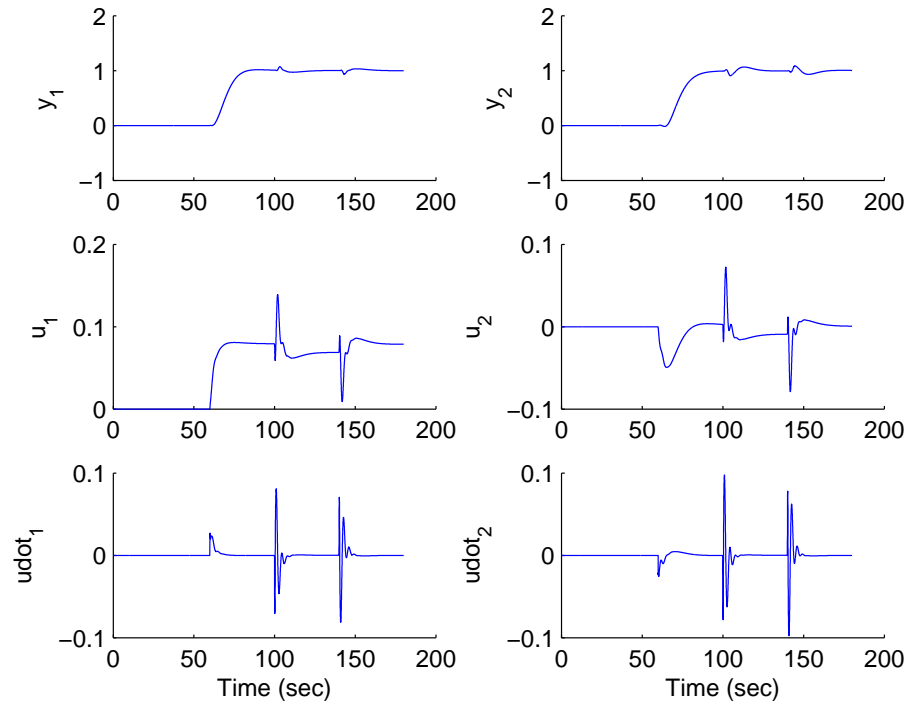


Figure 2.7: Food cooking extruder control with disturbances: solid- with disturbance, dash-dashed- without disturbance.

Chapter 2. Model Predictive Control

Case 5: Reducing the effect of the disturbance

The purpose of this simulation is to show how to reduce the effect of the disturbance on the system response.

The parameters in this case are the same as in case 4, except the pole location parameters are reduced to $p_1 = p_2 = 1/15$. The simulation shows that the disturbance is reduced when changing the pole location parameters.

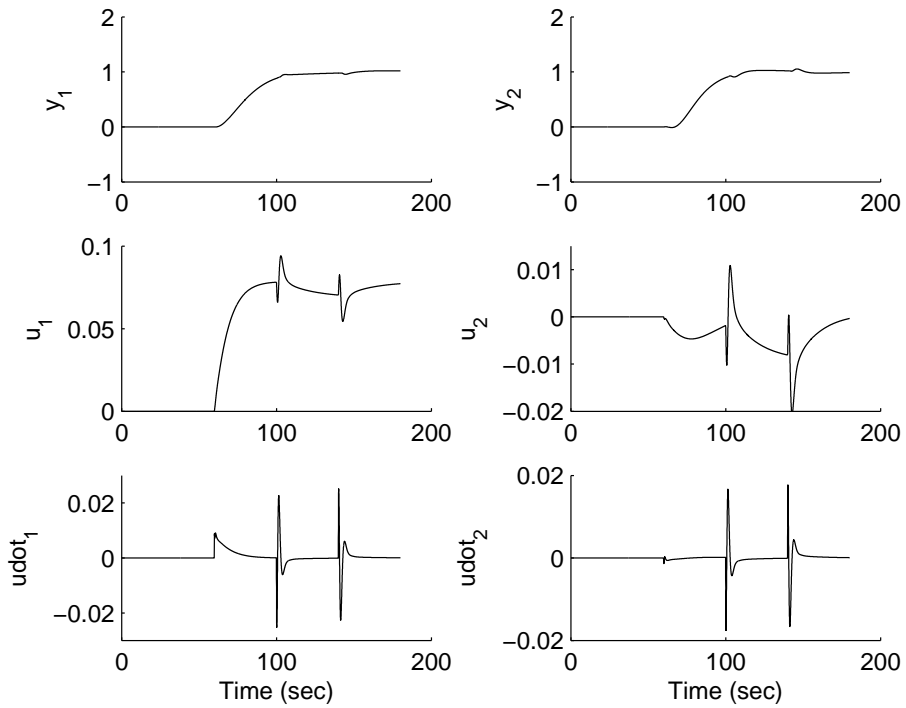


Figure 2.8: Increasing p to $1/15$ to reduce the effect of disturbance

2.3 Case Study 1: Food Cooking Extruder Process

Case 6: Constrained Control

The simulation shows how the MPC handles the constrained problems. The tuning parameters are kept the same as in case 4. The imposed constraints are:

$$\begin{aligned}
 -0.001 &\leq \dot{u}_1 \leq 0.01 \\
 -0.005 &\leq \dot{u}_2 \leq 0.002 \\
 -0.003 &\leq u_1 \leq 0.079 \\
 -0.02 &\leq u_2 \leq 0.002
 \end{aligned} \tag{2.27}$$

Conclusion: The operation constraints are satisfied by the control system.

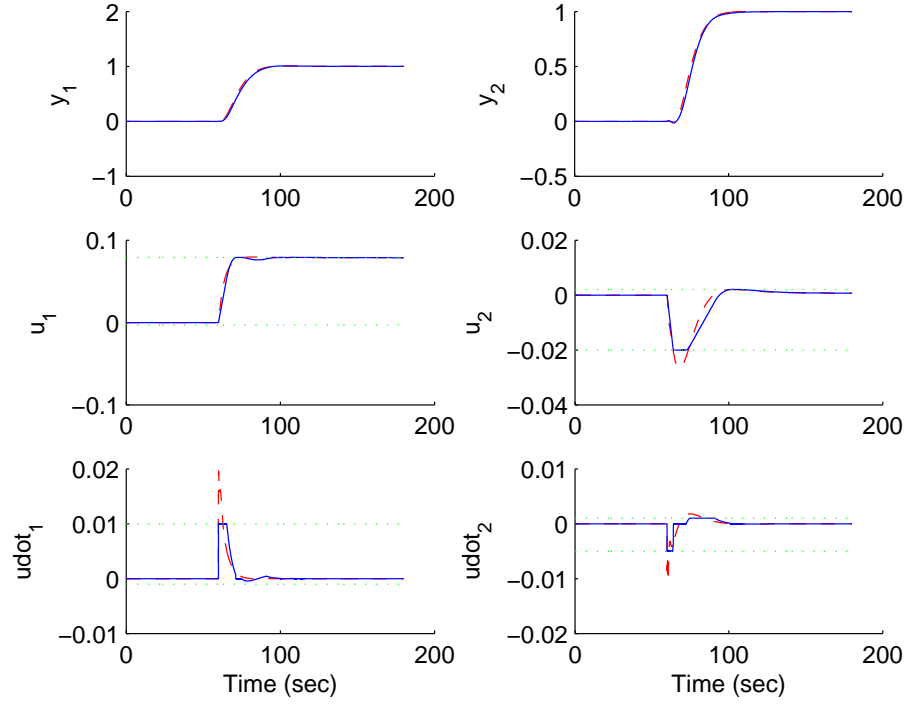


Figure 2.9: Continuous-time MPC results with constraints of the food cooking extruder (solid line: constrained control; dash-dashed line: unconstrained control. Process output: top-left: y_1 top-right: y_2 . Control signal: middle-left: u_1 middle-right: u_2 . Derivative of control signal: bottom-left: \dot{u}_1 bottom-right: \dot{u}_2).

Chapter 2. Model Predictive Control

2.4 Case Study 2: Copolymerization Process

This section introduces the case study of Copolymerization process. The process is more complicated with five inputs and two four outputs. The MPC method is used to lead the five outputs to setpoints.

2.4.1 Copolymerization Process

The second study was based the model for the methyl methacrylate (MMA) and vinyl acetate (VA) in a continuous stirred tank reactor, which appeared in the paper by Rao et al [10]. The model has five inputs, four outputs, and eighteen states. The purpose of this model was to testify the ability of the model predictive control approach while handling complicated system. The parameters are chosen as $p_1 = p_2 = 0.1$, $N_1 = N_2 = 3$, and $T_{p1} = T_{p2} = 100$. The transfer function is presented as the following:

$$G(s) = \begin{bmatrix} \frac{0.34}{0.85s+1} & \frac{0.21}{0.42s+1} & \frac{0.5(0.5s+1)}{12s^2+0.4s+1} & 0 & \frac{6.46(0.9s+1)}{0.07s^2+0.03s+1} \\ -0.41 & \frac{0.66}{1.51s+1} & \frac{-0.3}{1.45s+1} & 0 & \frac{-3.72}{0.8s+1} \\ \frac{2.41s+1}{0.3} & \frac{0.49}{1.54s+1} & \frac{-0.71}{1.35s+1} & \frac{-0.2}{2.7s+1} & \frac{-4.71}{0.008s^2+0.41s+1} \\ 0 & 0 & 0 & 0 & \frac{1.02}{0.07s^2+0.41s+1} \end{bmatrix} \quad (2.28)$$

where:

- u_1 : flows of monomer MMA.
- u_2 monomer VA.
- u_3 : initiator.
- u_4 : transfer agent.
- u_5 : temperature of the reactor jacket.
- y_1 : polymer production rate.
- y_2 : mole fraction of MMA in the polymer.
- y_3 : average molecular weight of the polymer.
- y_4 : reactor temperature.

The figures from 2.10 to 2.15 show the effects of tuning parameters to the response speed of the system. In this case study, the tuning parameters p

2.4 Case Study 2: Copolymerization Process

are strongly impacted to the control inputs.

The unit step disturbances (see figure 2.16) are introduced to the input responses in order to investigate the MPC performance. The comparison between the responses with and without disturbance are shown from figures 2.17 to 2.19. It is seen that the simulation results in these figures follow the same pattern as the noise free case. This is known as the "disturbance rejection" property of the MPC. Under the disturbance, the performance of the system can be improved by changing the pole positions.

The constraints are set to the inputs and its derivative. There are 20 constraints which are imposed to the system, where the future instant time τ_i in the cost function 2.17 is 0.01 and $i = 0$.

$$\begin{aligned}
 -3 &\leq \dot{u}_1 \leq 8 \\
 0 &\leq \dot{u}_2 \leq 2 \\
 -6 &\leq \dot{u}_3 \leq 0 \\
 0 &\leq \dot{u}_4 \leq 40 \\
 0 &\leq \dot{u}_5 \leq 0.1 \\
 0 &\leq u_1 \leq 362 \\
 0 &\leq u_2 \leq 100 \\
 -300 &\leq u_3 \leq 0 \\
 0 &\leq u_4 \leq 1800 \\
 0 &\leq u_5 \leq 1
 \end{aligned} \tag{2.29}$$

Figures 2.20, 2.21, and 2.22 shows derivative of inputs, inputs, and outputs under the constrained control. The constrains impact to the response of the system because the control inputs interact with each other. The output responses show a delay.

The constrained problems can be improved in order to meet the requirement in the practical. At each $\tau_i = 0.01$, the high and low limits of control inputs and derivative of control inputs $u_{high}(\tau_i)$, $u_{low}(\tau_i)$, $\dot{u}_{high}(\tau_i)$, and $\dot{u}_{low}(\tau_i)$ are reduced 0.01 in amplitude. With $i = 0, 1$, there are 40 constraints in the system. The result of the simulation is depicted in figures 2.23, 2.24, and 2.25 .

Chapter 2. Model Predictive Control

2.4.2 Simulation Results

Case 1a: Effects of the pole location tuning parameters p on the derivative of the inputs

The purpose of the first simulation is to investigate the effect of the pole location parameters p on the system responses. The prediction horizon parameters T_p and tuning parameters N are kept constant.

$$T_{p1} = T_{p2} = 100, \quad N_1 = N_2 = 3$$

The pole placement parameters p vary from slow response speed to the fast response speed. The corresponding parameters p are chosen as:

$$\text{Case1} : p_1 = p_2 = 0.075$$

$$\text{Case2} : p_1 = p_2 = 0.1$$

$$\text{Case3} : p_1 = p_2 = 0.15$$

The figure shows the simulation results of the derivative of the control inputs.

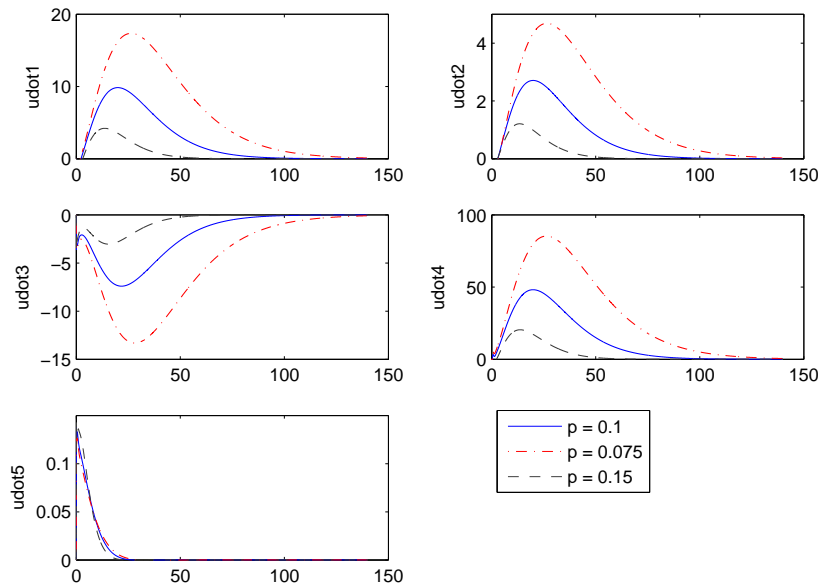


Figure 2.10: Closed-loop response for the derivative of the control inputs with different pole location tuning parameters p (solid- $p = 0.1$, dash-dotted- $p = 0.15$, and dash-dotted- $p = 0.075$).

2.4 Case Study 2: Copolymerization Process

Case 1b: Effects of the pole location tuning parameters p on the inputs

$$T_{p1} = T_{p2} = 100, \quad N_1 = N_2 = 3$$

The pole placement parameters p vary from slow response speeds to the fast response speed. The corresponding parameters p are chosen as:

$$\text{Case1} : p_1 = p_2 = 0.075$$

$$\text{Case2} : p_1 = p_2 = 0.1$$

$$\text{Case3} : p_1 = p_2 = 0.15$$

The figure shows the simulation results of the control inputs.

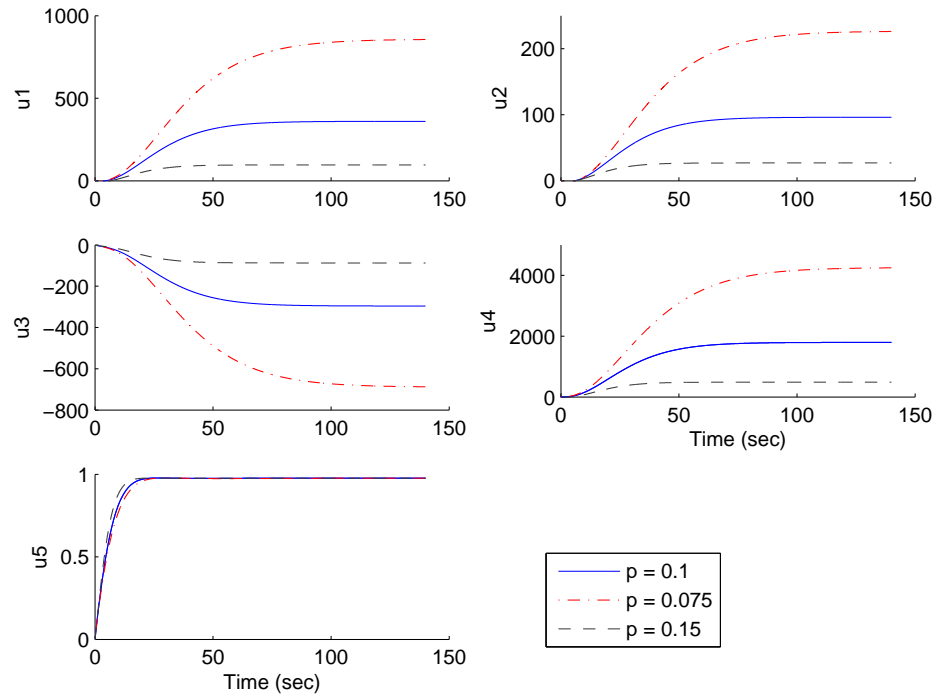


Figure 2.11: Closed-loop response for control inputs with different pole location tuning parameters p (solid- $p = 0.1$, dash-dashed- $p = 0.15$, and dash-dotted- $p = 0.075$).

Chapter 2. Model Predictive Control

Case 1c: Effects of the pole location tuning parameters p on the outputs

$$T_{p1} = T_{p2} = 100, \quad N_1 = N_2 = 3$$

The pole placement parameters p vary from slow response speeds to the fast response speed. The corresponding parameters p are chosen as:

$$\text{Case1} : p_1 = p_2 = 0.075$$

$$\text{Case2} : p_1 = p_2 = 0.1$$

$$\text{Case3} : p_1 = p_2 = 0.15$$

The figure shows the simulation results of the outputs.

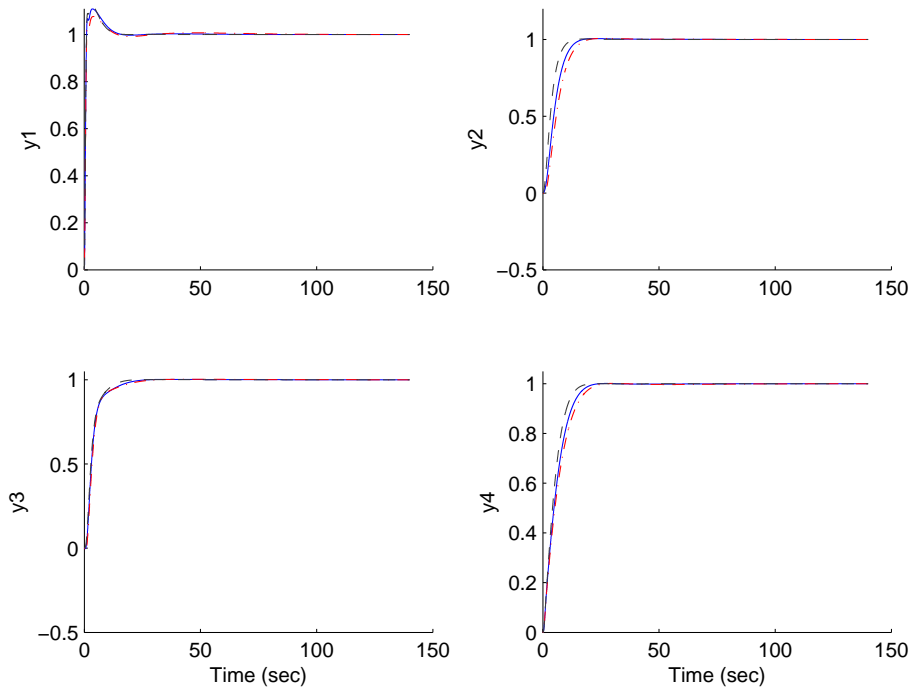


Figure 2.12: Closed-loop response for outputs with different pole location tuning parameters p (solid- $p = 0.1$, dash-dotted- $p = 0.075$, and dashed- $p = 0.15$).

2.4 Case Study 2: Copolymerization Process

Case 2a: Effects of the prediction horizon parameters T_p on the derivative of the inputs

The purpose of the first simulation is to investigate the effect of the prediction horizon parameters T_p on the system responses. The prediction horizon parameters p and tuning parameters N are kept constant.

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3$$

The pole placement parameters p vary from slow response speeds to the fast response speed. The corresponding parameters p are chosen as:

$$\text{Case1} : T_{p1} = T_{p2} = 50$$

$$\text{Case2} : T_{p1} = T_{p2} = 100$$

$$\text{Case3} : T_{p1} = T_{p2} = 350$$

The figure shows the simulation results of the derivative of the control inputs.

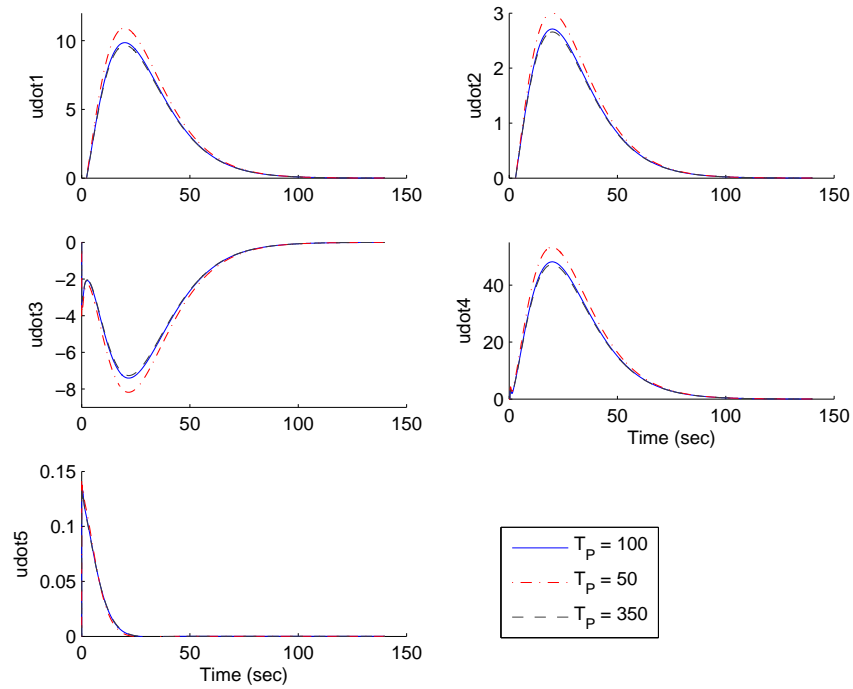


Figure 2.13: Closed-loop response for the derivative of the inputs with different prediction horizon parameters T_p (dash-dotted- $T_p = 50$, and dashed- $T_p = 350$).

Chapter 2. Model Predictive Control

Case 2b: Effects of the prediction horizon parameters T_p on the inputs

The prediction horizon parameters p and tuning parameters N are kept constant.

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3$$

The pole placement parameters p vary from slow response speeds to the fast response speed. The corresponding parameters p are chosen as:

$$\text{Case1} : T_{p1} = T_{p2} = 50$$

$$\text{Case2} : T_{p1} = T_{p2} = 100$$

$$\text{Case3} : T_{p1} = T_{p2} = 350$$

The figure shows the simulation results of the control inputs.

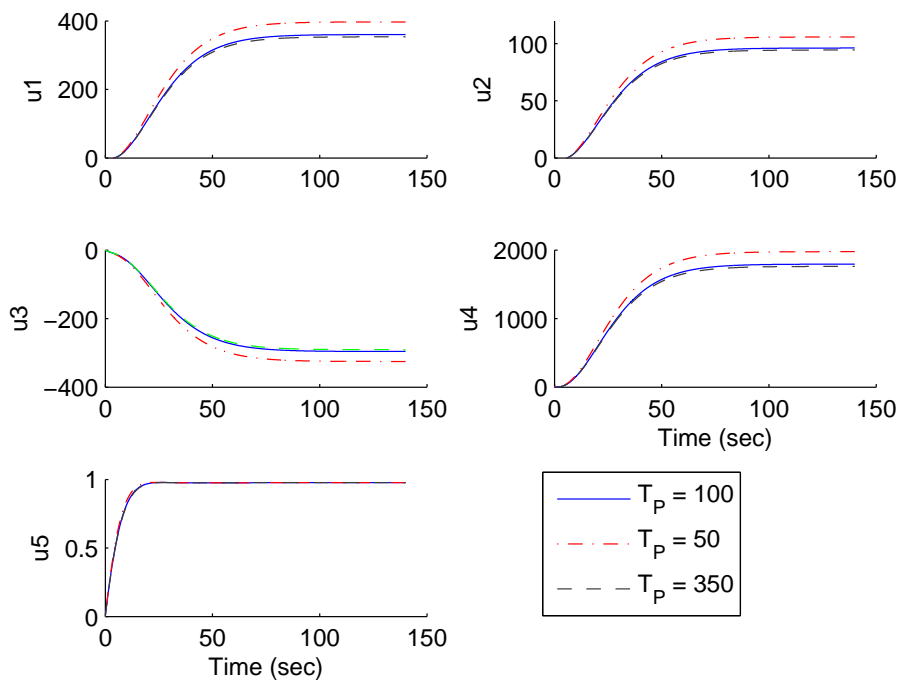


Figure 2.14: Closed-loop response for the inputs with different prediction horizon parameters T_p (dash-dashed- $T_p = 350$, and dash-dotted- $T_p = 50$).

2.4 Case Study 2: Copolymerization Process

Case 2c: Effects of the prediction horizon parameters T_p on the outputs

The prediction horizon parameters p and tuning parameters N are kept constant.

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3$$

The pole placement parameters p vary from slow response speeds to the fast response speed. The corresponding parameters p are chosen as:

$$\text{Case1} : T_{p1} = T_{p2} = 50$$

$$\text{Case2} : T_{p1} = T_{p2} = 100$$

$$\text{Case3} : T_{p1} = T_{p2} = 350$$

The figure shows the simulation results of the outputs.

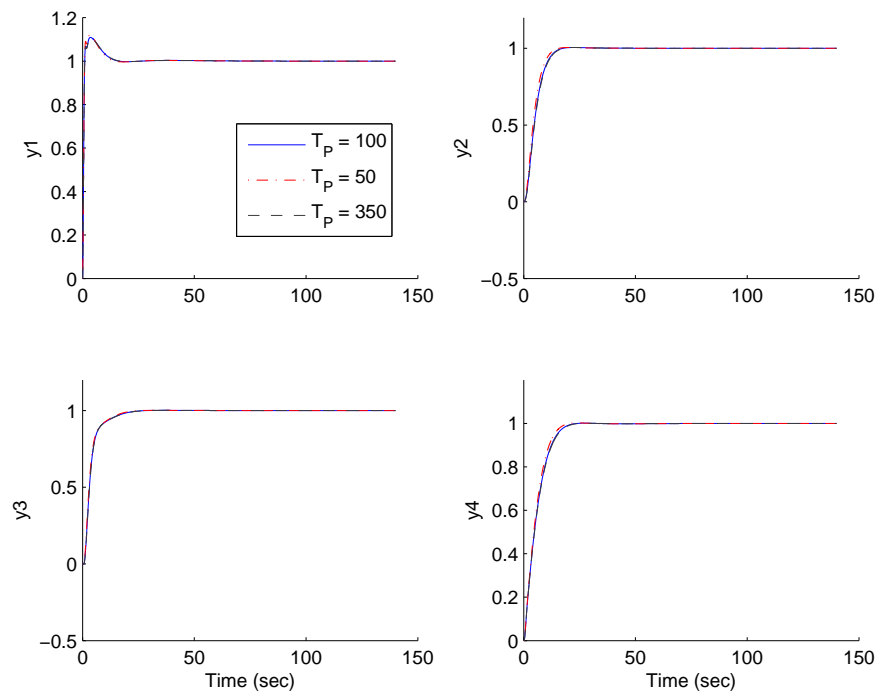


Figure 2.15: Closed-loop response for the outputs with different prediction horizon parameters T_p (dash-dashed- $T_p = 350$, and dash-dotted- $T_p = 50$).

Chapter 2. Model Predictive Control

Case 3a: Input Step Disturbance Rejection

The purpose of the simulation is to investigate how the MPC approach handles the disturbance of the system. The tuning parameters are:

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3, \quad T_{p1} = T_{p2} = 100$$

The figure shows step input disturbances added into the inputs.

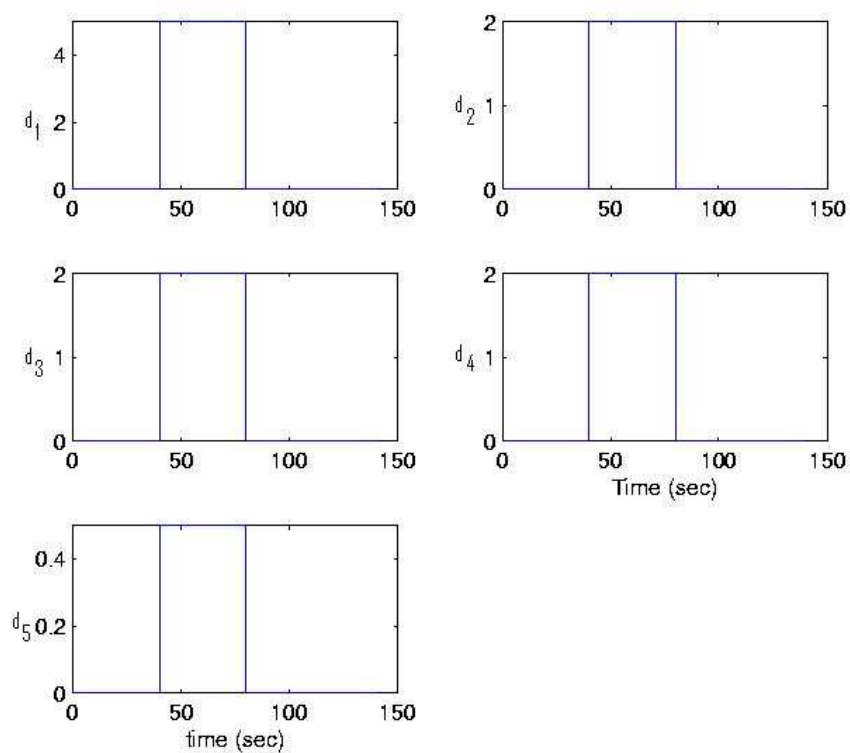


Figure 2.16: Disturbances are added into the inputs.

2.4 Case Study 2: Copolymerization Process

Case 3b: Disturbance Rejection: Derivative of the input \dot{u}

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3, \quad T_{p1} = T_{p2} = 100$$

The figure shows the simulation results of the derivative of the inputs.

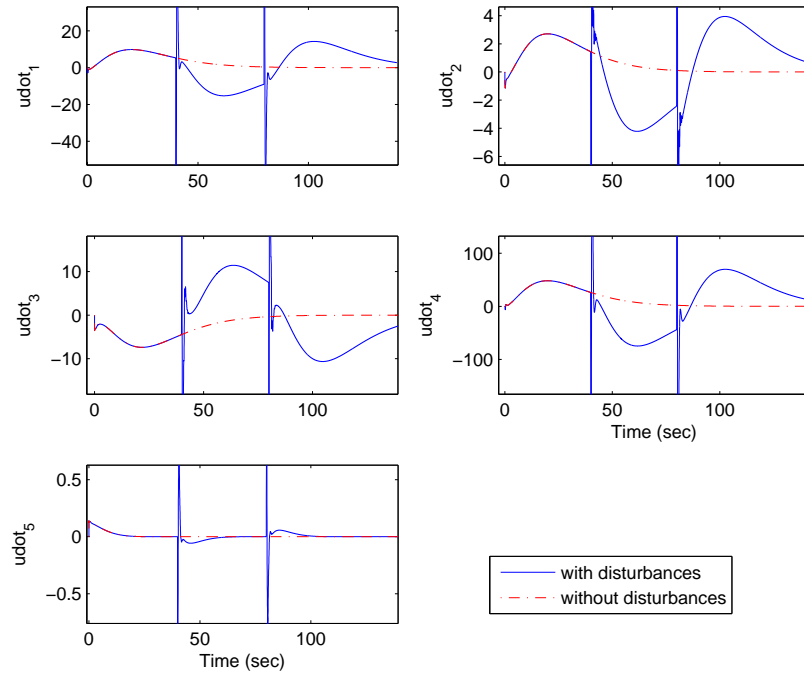


Figure 2.17: Copolymerization process control with disturbances: solid- with disturbance, dash-dashed- without disturbance).

Chapter 2. Model Predictive Control

Case 3c: Disturbance Rejection: the input u

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3, \quad T_{p1} = T_{p2} = 100$$

The figure shows the simulation results of the inputs.

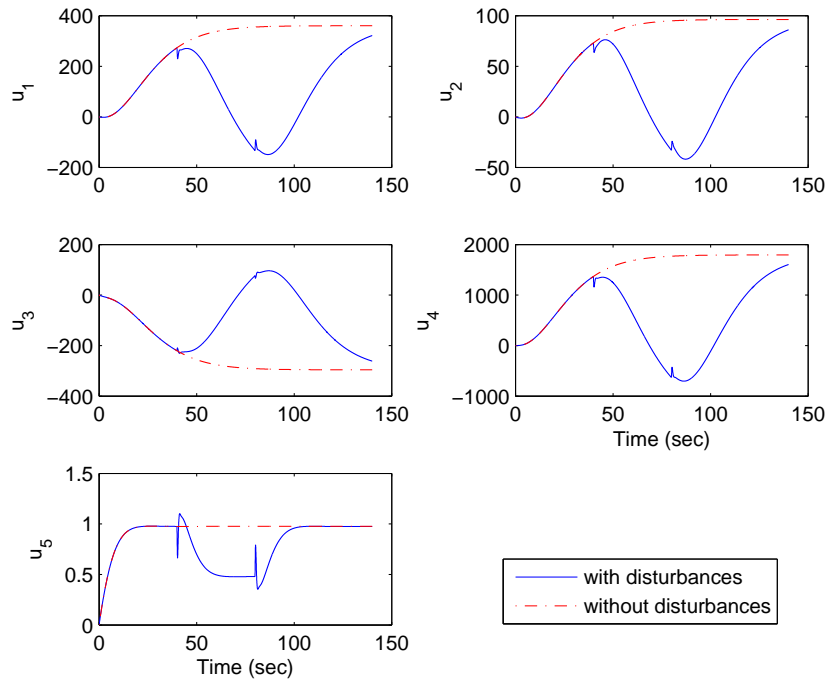


Figure 2.18: Copolymerization process control with disturbances: solid- with disturbance, dash-dashed- without disturbance).

2.4 Case Study 2: Copolymerization Process

Case 3d: Disturbance Rejection: the outputs y

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3, \quad T_{p1} = T_{p2} = 100$$

The figure shows the simulation results of the outputs.

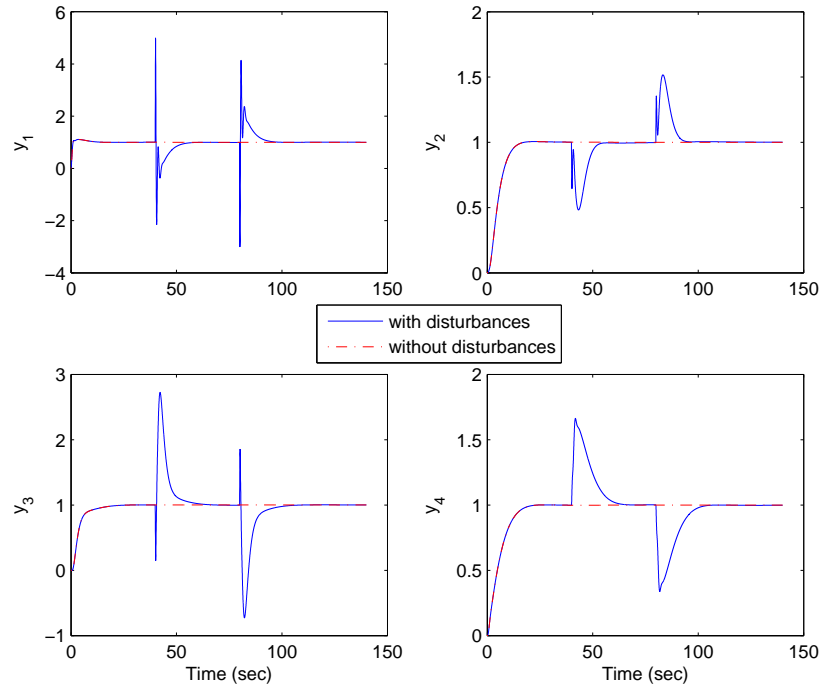


Figure 2.19: Copolymerization process control with disturbances: solid- with disturbance, dash-dashed- without disturbance).

Chapter 2. Model Predictive Control

Case 4a: Constraints on derivative of inputs, $\tau_i = 0.01$, $\mathbf{i} = \mathbf{0}$

The figure shows the simulation results of the derivative of inputs under constrained problems. The tuning parameters are;

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3, \quad T_{p1} = T_{p2} = 100 \quad \tau_0 = 0.01$$

There are 20 constraints in the system.

$$-3 \leq \dot{u}_1 \leq 8, \quad 0 \leq \dot{u}_2 \leq 2, \quad -6 \leq \dot{u}_3 \leq 0, \quad 0 \leq \dot{u}_4 \leq 40, \quad 0 \leq \dot{u}_5 \leq 0.1,$$
$$0 \leq u_1 \leq 362, \quad 0 \leq u_2 \leq 100, \quad -300 \leq u_3 \leq 0, \quad 0 \leq u_4 \leq 1800, \quad 0 \leq u_5 \leq 1.$$

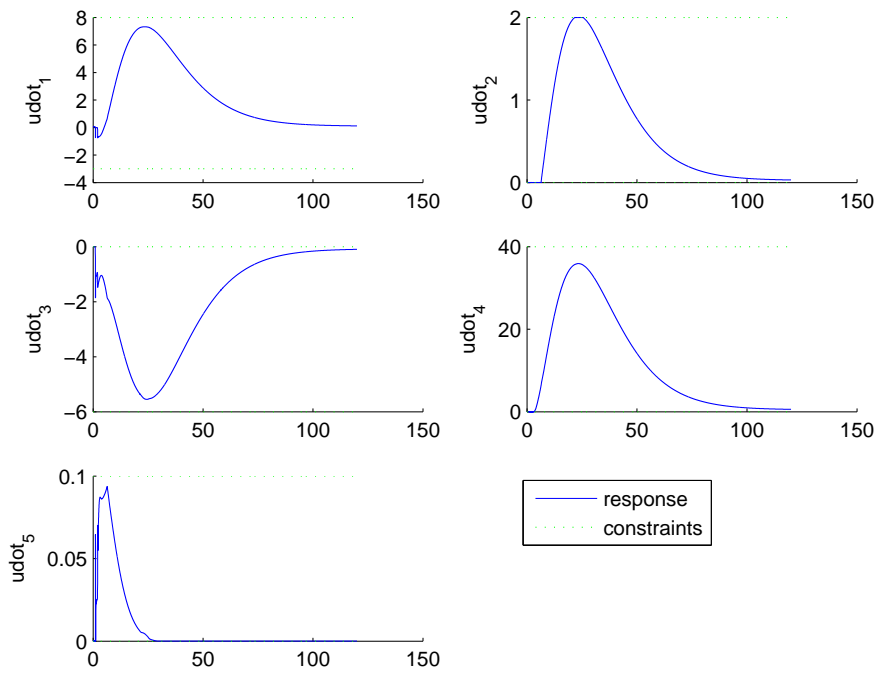


Figure 2.20: Continuous-time MPC results with constraints on the derivative of inputs, where $\tau_i = 0.01$ and $\mathbf{i} = \mathbf{0}$.

2.4 Case Study 2: Copolymerization Process

Case 4b: Constraints on inputs, $\tau_i = 0.01$, $i = 0$

The parameters and constraints are the same as in the Case 4a. This figure shows the simulation results of the inputs under the constrained problems.

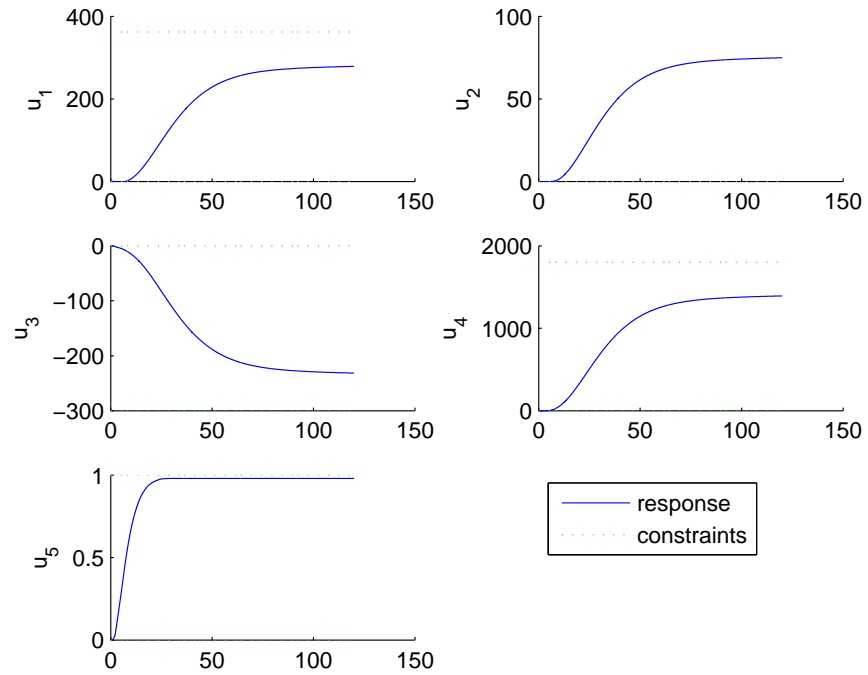


Figure 2.21: Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0$.

Chapter 2. Model Predictive Control

Case 4c: Constraints on outputs, $\tau_i = 0.01$, $\mathbf{i} = \mathbf{0}$

The parameters and constraints are the same as in the Case 4a. This figure shows the simulation results of the outputs under the constrained problems.

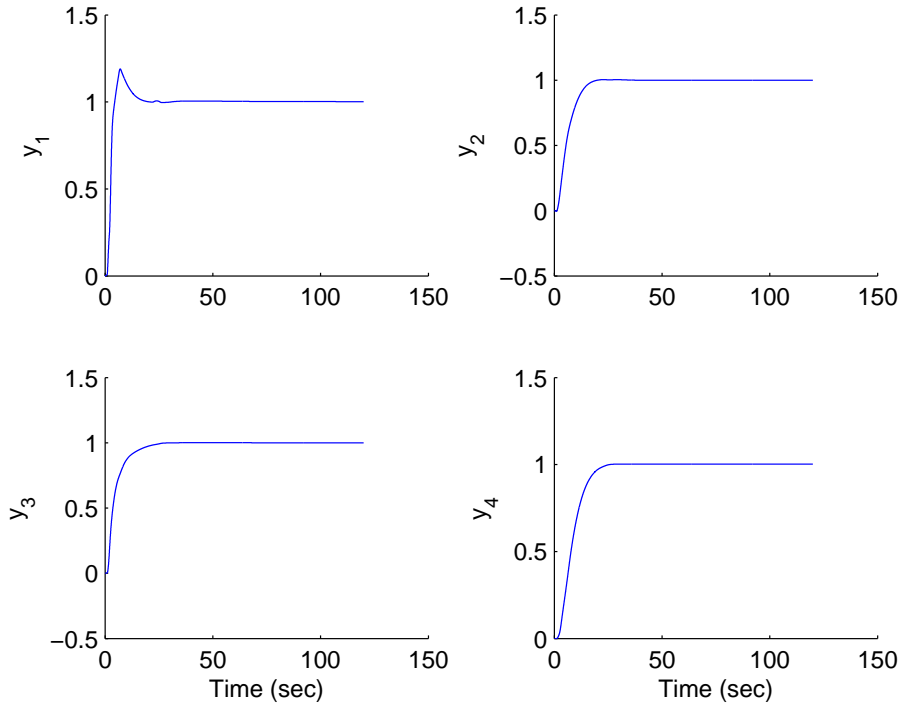


Figure 2.22: Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $\mathbf{i} = \mathbf{0}$.

2.4 Case Study 2: Copolymerization Process

Case 5a: Constraints on derivative of inputs, $\tau_i = 0.01$, $i = 0, 1$

The figure shows the simulation results of the derivative of inputs under constrained problems. The tuning parameters are:

$$p_1 = p_2 = 0.1, \quad N_1 = N_2 = 3, \quad T_{p1} = T_{p2} = 100 \quad \tau_i = 0.01$$

There are 40 constraints in the system.

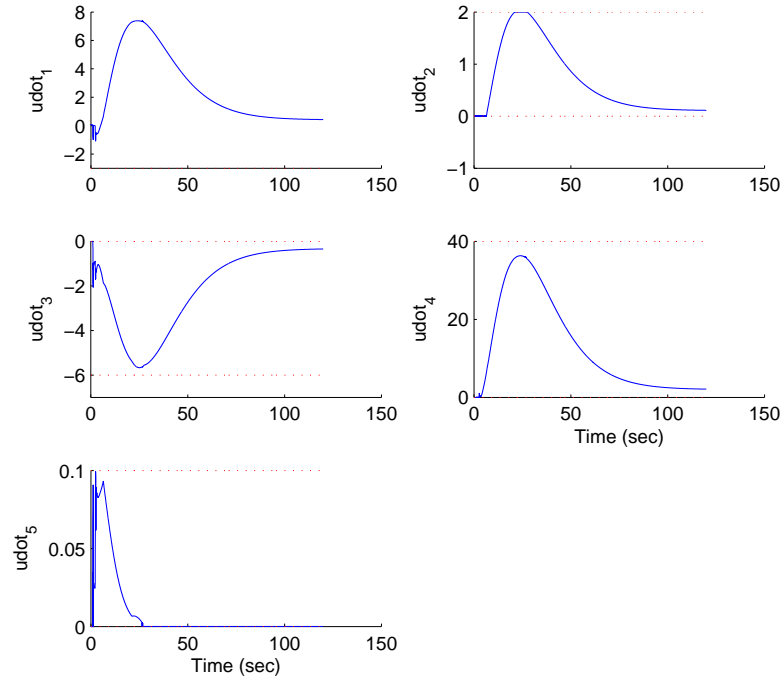


Figure 2.23: Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0, 1$

Chapter 2. Model Predictive Control

Case 5b: Constraints on inputs, $\tau_i = 0.01$, $\mathbf{i} = \mathbf{0}, \mathbf{1}$

The parameters and constraints are the same as in the Case 5a. This figure shows the simulation results of the inputs under the constrained problems.

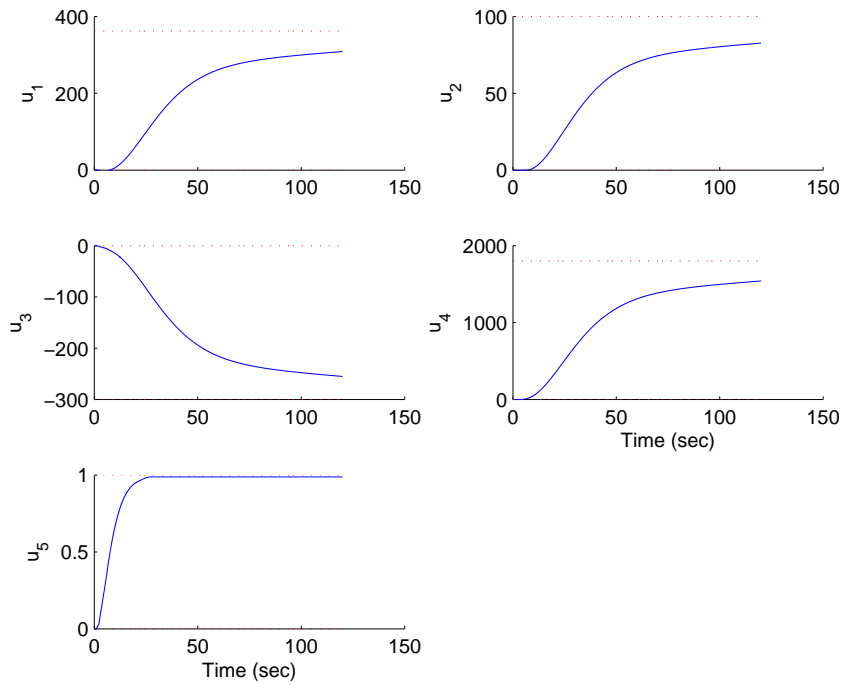


Figure 2.24: Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $\mathbf{i} = \mathbf{0}, \mathbf{1}$.

2.4 Case Study 2: Copolymerization Process

Case 5c: Constraints on outputs, $\tau_i = 0.01$, $i = 0, 1$

The parameters and constraints are the same as in the Case 5a. This figure shows the simulation results of the outputs under the constrained problems.

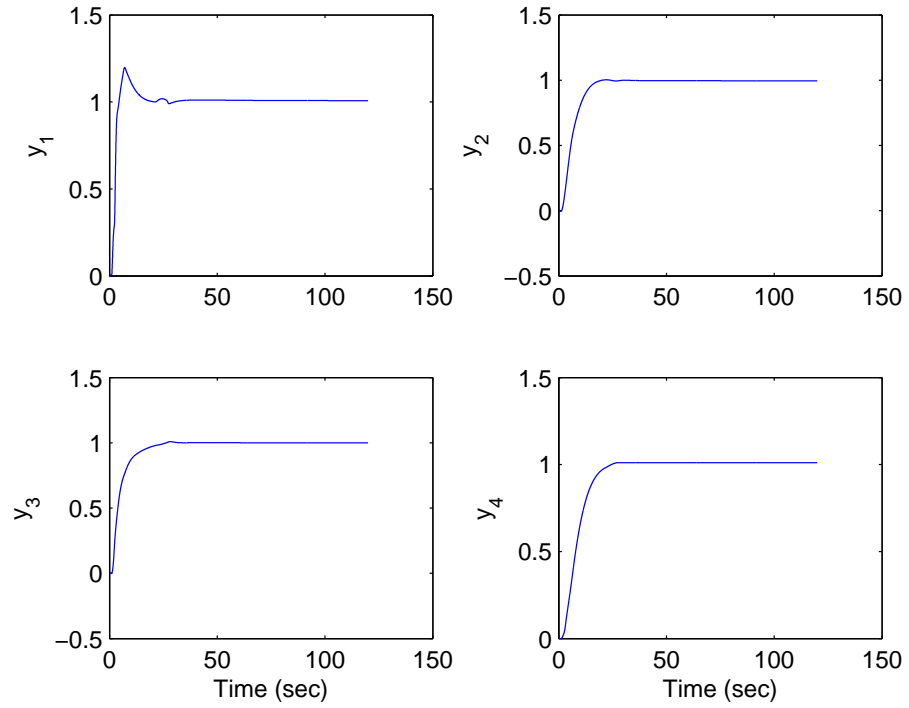


Figure 2.25: Continuous-time MPC results with constraints of the Copolymerization process with $\tau_i = 0.01$ and $i = 0, 1$.

Chapter 2. Model Predictive Control

2.5 Summary

In this chapter, the concepts of continuous time model predictive control algorithm are reviewed. In the unconstrained problem, the optimal solution of the system is derived directly and is also obtained by using the optimization algorithm. The MPC in this case is similar to Linear Quadratic Regulator (LQR). When the constraints are imposed to the system, the Quadratic Programming method is applied to the MPC to handle the constraints. The case studies show evidence of the ability of constrained control of the MPC. The MPC produces stable and reliable responses. The delays may occur due to the restriction on the control inputs. However, they can be improved by changing the constrained sets or adjusting the tuning parameters which are a key success of the MPC design using Laguerre functions. The MPC also guarantees the stability and fast responses once the disturbances happen in the system model. The effects of the disturbances can be reduced by choosing appropriate tuning parameters.

Chapter 3

Review Quadratic Programming Methods

3.1 Introduction

Inequality constraints in the MPC method can be treated by optimization algorithms. Optimization problems begin with a set of independent variables and often include conditions that define acceptable values of variables. The conditions are termed as the constraints of the problems. The algorithms will be applied to the cost equations to obtain the optimal derivative of the control signal and to reduce the cost of computations. There are four types of methods devised for solving constrained problem. They are primal methods, dual methods, penalty and barrier methods, and Lagrange methods.

In the formula of the MPC problem, some state constraints are imposed by physical problems. Other constraints are less important. They may represent desired ranges of operation for the plant. In some situations, no set of inputs and states for the MPC problem may satisfy all these constraints. Rather than having the algorithm declares infeasibility and returns without a result, it is better to obtain the solution that enforces some constraint strictly (hard constraints), while relaxing others. The Active Set method as described in section 3.2 is a typical method which is able to handle such problems.

Prior to solve the problem, we cannot know how large the constraints could be so that the MPC is able to handle. In some process plant such as chem-

Chapter 3. Review Quadratic Programming Methods

ical process, the *Active Set method* can not prove the advantage to deal with large amount of constraints. In section 3.3, the general description of the Primal Dual Interior Point method using Mehrotra's predictor corrector algorithm is introduced to solve the problem. The method solves the solution in dual problem and tests the convergence in the primal problem to guarantee the feasibility of the solutions. Another primal dual method for solving QP problem called Hildreth's Quadratic Programming procedure is described in section 3.4. The advantage of this procedure is the ease for computer programming. Section 3.5 describes the last method which is a Shor's r-Algorithm. The method is able to handle the nonlinear constrained problem which occurs in the physical applications.

Two case studies to investigate the performances of the QP methods are shown in section 3.6. In the first case study, the basic constrained set is constructed. In the second one, the constrained set is developed to support the future prediction purpose of the MPC methods. The constrained set varies with time, is depend on variables, is larger, and is more complex. The food cooking extruder process is used for comparing the Active Set method, Hildreth's QP Procedure, Primal Dual Interior Point method, and Shor's r-Algorithm. The calculation speeds of those QP methods are also recorded. The conclusion and comments of this chapter are stated in section 3.7.

Some successful methods for addressing this class of problems are the active set method [78, 83], the primal-dual interior point method [88], Hildreth's quadratic programming procedure [14], and Shor's r-Algorithm [64, 65].

3.2 Active Set Method

Before introducing the method, the definition is introduced. At the feasible point η , the constraint $m_i^T \eta \leq \gamma$ is said to be active if $m_i^T \eta = \gamma$, and is said to be inactive if $m_i^T \eta < \gamma$.

\hat{M} is a constraint matrix for active constraints and Z is a null space matrix of the matrix \hat{M} . Matrix Z can be found by applying the QR factorization for the active set matrix. At η_0 , the original set of constraints is assumed to be active. The search direction p in the equation (3.1) within the line search method is used and the appropriate step length α if found in the equation (3.2)

$$p = -ZZ^T \nabla f(\eta_k) \quad (3.1)$$

$$\bar{\alpha} = \max\{\alpha : \eta + \alpha p \text{ is feasible}\} \quad (3.2)$$

3.3 Primal Dual Interior Point Algorithm

$$= \min\{(a_i^T \eta - b_i)/(-a_i^T p) : a_i^T p < 0, i \notin W\}$$

The step length in the inequality constraint might lead to the infeasible solution that may violate one or more constraints that are eliminated. As the solution η moves along feasible search direction p , the boundary of constraint for some step length $\bar{\alpha}$ is constructed as in (3.2). The value $\bar{\alpha}$ represents the largest step that may be taken without violating feasibility. If the step length is $\bar{\alpha}$, no step is taken and the constraint is added to the working set. Whereas, to determine with constraint is dropped from the working set, the Lagrange multipliers λ are computed as follows:

$$\lambda = \hat{M}^{-T} \nabla f(\eta_k) \quad (3.3)$$

If the Lagrange multipliers are not all nonnegative, then η is also a solution to the original inequality constrained problem. The problem is solved. However, if some Lagrange multipliers are negative, the η is not an optimal solution. The negative multiplier indicates that the function can be decreased if we move away from the corresponding constraint into the interior of the feasible region. Hence we can drop this constraint from the working set. We now have a new working set and the process is repeated.

$$f(\eta_k + \alpha p) < f(\eta_k), \alpha \leq \bar{\alpha} \quad (3.4)$$

$$\eta_{k+1} = \eta_k + \alpha p \quad (3.5)$$

3.3 Primal Dual Interior Point Algorithm

In this section, an interior point based approach, which is used to solve the MPC problem, is described. The approach is efficient when dealing with large constrained problems. It maintains feasibility to both primal problem and dual problem. The primal problem has the standard form:

$$\begin{aligned} \text{Minimize : } & J = \frac{1}{2} \eta^T E \eta + \eta^T F \\ \text{Subject to : } & M \eta \leq \gamma \end{aligned} \quad (3.6)$$

The appropriate dual problem is also established by introducing the Lagrange multiplier λ . The dual interior point methods operate on dual problem.

$$\begin{aligned} \text{Minimize : } & J = \frac{1}{2} \eta^T E \eta + \eta^T \gamma \\ \text{Subject to : } & E \eta + F + M^T \lambda = 0, \lambda \geq 0 \end{aligned} \quad (3.7)$$

Chapter 3. Review Quadratic Programming Methods

The primal dual methods attempt to solve the primal and dual problems simultaneously. The objective functions (3.6) and (3.7) coincide at the optimal solutions as the solutions λ and η are solved. The dual method is used to compute the estimated of the optimal primal variable; whereas the primal method is used to test for convergence. The primal-dual methods find the solution to the primal-dual linear programming pair by applying the modified Newton's method to the Karush Kuhn Kucker (KKT) conditions. The solutions of the primal dual programming, which satisfied the KKT conditions, is stated as follows:

$$\begin{aligned} E\eta + F + M^T\lambda &= 0 \\ M^T\eta - \gamma &\leq 0 \\ \lambda &\geq 0 \\ \lambda_i(M\eta - \gamma) &= 0 \end{aligned} \tag{3.8}$$

The advantage of the primal dual method is that it does not solve the inequality constraints directly. It starts more compactly by rewriting the inequality constraints $M\eta \leq \gamma$ into the equality constraint (3.9) which a slack variable t is introduced:

$$\begin{bmatrix} E\eta + F + M^T\lambda \\ M^T\eta - \gamma + t \\ \Gamma\Lambda e \end{bmatrix} = 0 \tag{3.9}$$

where Γ and Λ are diagonal matrices defined by:

$$\begin{aligned} \Gamma &= \text{diag}(t_1, t_2, \dots, t_m), & \Lambda &= \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_m) \\ e &= [1, 1, \dots, 1]^T \end{aligned}$$

The method generates iterates $[\eta^k, \lambda^k, t^k]^T$ that satisfy strictly conditions $(\lambda^k, t^k) \geq 0$. The general linear system can be written to solve for the search direction as:

$$\begin{bmatrix} E & M^T & 0 \\ M & 0 & I \\ 0 & \Gamma & \Lambda \end{bmatrix} \begin{bmatrix} \Delta\eta \\ \Delta\lambda \\ \Delta t \end{bmatrix} = - \begin{bmatrix} E\eta + F + M^T\lambda \\ M^T\eta - \gamma + t \\ \Gamma\Lambda e \end{bmatrix} = \begin{bmatrix} r_Q \\ R_C \\ r_T \end{bmatrix} \tag{3.10}$$

The Mehrotra's predictor-corrector algorithm is used to solve the quadratic programming in the equation (3.6) rather than using the Newton's search direction as in the equation (3.5). The reason is that the Newton's search direction does not reach quickly to the feasible solution as the step length is too small ($\alpha \ll 1$). The first part of the Mehrotra's search direction is known as the predictor step or affine scaling step. It is a pure Newton step for the system (3.9) and is obtained from a linear approximation to the equality

3.3 Primal Dual Interior Point Algorithm

conditions in (3.10). It results the set of solution $[\Delta\eta^{aff}, \Delta\lambda^{aff}, \Delta t^{aff}]^T$. The step lengths to the boundary along the direction are taken as follows:

$$\begin{aligned}\alpha_{aff}^{primal} &= \max(\alpha \in [0, 1], t + \alpha\Delta t^{aff} > 0) \\ \alpha_{aff}^{dual} &= \max(\alpha \in [0, 1], \lambda + \alpha\Delta\lambda^{aff} > 0)\end{aligned}\quad (3.11)$$

The duality gap μ_{aff} attained from this full step to the boundary is:

$$\mu_{aff} = (\lambda + \alpha_{aff}^{dual}\Delta\lambda^{aff})^T(t + \alpha_{aff}^{primal}\Delta t^{aff})/m \quad (3.12)$$

Finally, the centering parameter σ is calculated by the following heuristic:

$$\sigma = \left(\frac{\mu_{aff}}{\mu}\right)^3 \quad (3.13)$$

The second part of the search direction is the centering-corrector direction. The solution $[\Delta\eta^{cor}, \Delta\lambda^{cor}, \Delta t^{cor}]^T$ are calculated by solving the following system:

$$\begin{bmatrix} E & M^T & 0 \\ M & 0 & I \\ 0 & \Gamma & \Lambda \end{bmatrix} \begin{bmatrix} \Delta\eta^{cor} \\ \Delta\lambda^{cor} \\ \Delta t^{cor} \end{bmatrix} = - \begin{bmatrix} 0 \\ 0 \\ \Delta\Gamma\Delta\Lambda e + \sigma\mu \end{bmatrix} \quad (3.14)$$

This is effectively using Newton's method to try to solve for a point on the central path given by (3.8). The combined search direction is obtained by adding the predictor and centering-corrector directions:

$$[\Delta\eta, \Delta\lambda, \Delta t]^T = [\Delta\eta^{aff}, \Delta\lambda^{aff}, \Delta t^{aff}]^T + [\Delta\eta^{cor}, \Delta\lambda^{cor}, \Delta t^{cor}]^T \quad (3.15)$$

The actual step lengths $(\alpha^{primal}, \alpha^{dual})$ are chosen according to the heuristic. The next iterate is computed as follows:

$$\begin{aligned}(\eta^{k+1}, t^{k+1}) &= (\eta^k, t^k) + \alpha^{primal}(\Delta\eta, \Delta t) \\ \lambda^{k+1} &= \lambda^k + \alpha^{dual}\Delta\lambda\end{aligned}\quad (3.16)$$

Since the matrix $\Gamma^{-1}\Lambda$ and $(E + M^T\Gamma^{-1}\Lambda)$ are diagonal and nonsingular, the search direction solution is obtained as follows:

$$\begin{aligned}\Delta\eta &= (E + M^T\Gamma^{-1}\Lambda)[r_Q + M^T\Gamma^{-1}(\Lambda r_C - r_T)] \\ \Delta\lambda &= \Gamma^{-1}\Lambda(M\Delta\lambda - r_C) + \Gamma^{-1}r_T \\ \Delta t &= \Lambda^{-1}[r_T - \Gamma\Delta\lambda]\end{aligned}\quad (3.17)$$

The interior point algorithms terminate when first order optimality conditions (3.8) are satisfied with some predetermined tolerance, which is usually chosen as 10^{-8} :

$$(r_Q, r_C, r_T) \leq 10^{-8}$$

Chapter 3. Review Quadratic Programming Methods

3.4 Hildreth's Quadratic Programming Procedure

Similar to the primal-dual interior point algorithm, the Hildreth's method can be presented in the primal problem:

$$\begin{aligned} \text{Minimize :} & \quad \frac{1}{2}\eta^T E\eta + F^T\eta & (3.18) \\ \text{Subject to :} & \quad M\eta \leq \gamma \end{aligned}$$

and obtains the problem that equivalent to the dual problem:

$$\begin{aligned} \text{Maximize :} & \quad -\frac{1}{2}\eta^T P\eta - \eta^T d - \frac{1}{2}F^T Q^{-1} & (3.19) \\ \text{Subject to :} & \quad \lambda \geq 0 \end{aligned}$$

where $P = ME^{-1}M^T$, $d = MQ^{-1}F + \gamma$, or equivalently:

$$\begin{aligned} \text{Minimize :} & \quad \frac{1}{2}\eta^T P\eta - \eta^T d & (3.20) \\ \text{Subject to :} & \quad M\eta \geq \gamma \end{aligned}$$

As aforementioned, the dual method can be solved more efficiently than primal method. The resulted optimal solution η_0 is:

$$\eta_0 = E^{-1}(F - M^T \lambda_0) \quad (3.21)$$

To obtain the strictly condition $\lambda \geq \theta$, vector λ_i is adjusted to minimize the function and is set to zero if λ_i is negative. From one complete cycle through the components to be one iteration taking the vector λ^k to λ^{k+1} , the appropriate vector λ_i^{k+1} is expressed as:

$$\lambda_i^{k+1} = \max\left(0, -\frac{1}{p_{ii}}\left(d_i + \sum_{j=1}^{i-1} p_{ij}\lambda_j^{k+1} + \sum_{j=i+1}^n p_{ij}\lambda_j^k\right)\right) \quad (3.22)$$

If the objective function is decreased, the next component λ_{i+1} is considered.

3.5 Shor's r-Algorithm

$F(\eta)$ is assumed to be an objective function under unconstrained problem. In case of the nonlinear constraints $h(\eta) \leq 0$ available, the objective function can be rewritten as follows:

$$f(\eta) = F(\eta) + \lambda(0, \max_{i=1, \dots, m_e} |h_i(\eta)|, \max_{j=m_e+1, \dots, m} |h_j(\eta)|) \quad (3.23)$$

3.6 Case Studies

The Shor's r-algorithm uses the same procedure to handle either unconstrained problem $F(\eta)$ or constrained problem $f(\eta)$. Therefore, $g(\eta)$ is assumed to be a gradient of the objective function, say $f(\eta)$. The objective function $f(\eta)$, which subjected to constraints, is minimized by using the iterative algorithm.

The key feature of the Shor's algorithm is to make step in the direction opposite to a sub-gradient at the current point. The steps are computed in the transformed space by taking the difference between current and previous sub-gradient.

$$r_{k+1} = \bar{g}_f(\eta_k) - \bar{g}_f(\eta_{k-1}) \quad (3.24)$$

The direction of the next dilatation is performed:

$$\eta_{k+1} = \frac{B_k^T r_k}{\|B_k^T r_k\|} \quad (3.25)$$

in order to obtain the operators of the dilation of the space:

$$R_\alpha(\xi) = I_n + (\alpha - 1)\xi\xi^T \quad (3.26)$$

where α is the coefficient of dilatation ($\alpha > 1$) and $\beta = 1/\alpha$. The matrix of the space transformation is calculated as follows:

$$B_{k+1} = B_k R_{\beta_k}(\xi_k) \quad (3.27)$$

The new approximation of η_{k+1} is:

$$\eta_{k+1} = \eta_k - h_k B_k \frac{B_k^T \bar{g}_f(\eta_k)}{\|B_k^T \bar{g}_f(\eta_k)\|} = \eta_k - h_k B_k \delta_k \quad (3.28)$$

A step factor h_k can be determined from the condition of the approximate minimum toward the direction $B_k \delta_k$ in the initial space. The process is repeated until the following stopping criteria is satisfied:

$$\begin{aligned} |\eta_{k+1}^i - \eta_k^i| &\leq \delta_x |\eta_{k+1}^i| \\ |f(\eta_{k+1}) - f(\eta_k)| &\leq \delta_f |f(\eta_{k+1}^i)| \end{aligned} \quad (3.29)$$

where δ_x and δ_f are very small ($\approx 10^{-8}$).

3.6 Case Studies

This section is to investigate the performance of the Quadratic Programming methods. The food cooking extruder example is still used as a case

Chapter 3. Review Quadratic Programming Methods

study. There are eight (8) constraints as in (3.30) are imposed into the cost functions.

$$\begin{aligned} -0.0015 &\leq \dot{u}_1 \leq 0.015 \\ -0.001 &\leq \dot{u}_2 \leq 0.003 \\ -0.003 &\leq u_1 \leq 0.08 \\ -0.001 &\leq u_2 \leq 0.001 \end{aligned} \tag{3.30}$$

The future time instant $\tau_i = 0.01$, where $i = 0$, is chosen. The Hildreth Quadratic Programming procedure, Active Set method, and Primal Dual Interior Point method give the same results. The simulation results are shown in figure 3.1. The Shor's r-Algorithm is mainly used to handle the nonlinear model. Its simulation result is depicted in figure 3.2.

The computational time of each QP method is also recorded while simulating the single-input single-output (SISO) and multiple-inputs multiple-outputs plants. The time (in seconds) is shown in table below.

	Unconstraint	Hildreth	Active Set	PDIP	Shor's r
SISO	3.52	5.92	5.68	8.95	36.94
MIMO	4,98	7.16	7.05	21.75	113.69

3.6 Case Studies

Case 1: Active Set, DPIP, and Hildreth QP Methods, $\tau_0 = 0.01$

The purpose of the simulation is to investigate the performance of the QP methods. The Active Set method, Primal Dual Interior Point (DPIP) method, and Hildreth Quadratic Programming Procedure are used. The tuning parameters of the MPC algorithm are chosen as $p_1 = p_2 = 1/15$, $T_{p1} = T_{p2} = 150$, and $N_1 = N_2 = 3$. The future time instants is $\tau_i = 0.01$, $i = 0$. The imposed constraints are:

$$\begin{aligned} -0.0015 &\leq \dot{u}_1 \leq 0.015 & -0.001 &\leq \dot{u}_2 \leq 0.003 \\ -0.003 &\leq u_1 \leq 0.08 & -0.001 &\leq u_2 \leq 0.001 \end{aligned}$$

Dashed-dashed line: unconstrained control, solid line: constrained control. The simulation results using Active Set, DPIP, and Hildreth QP methods are identical.

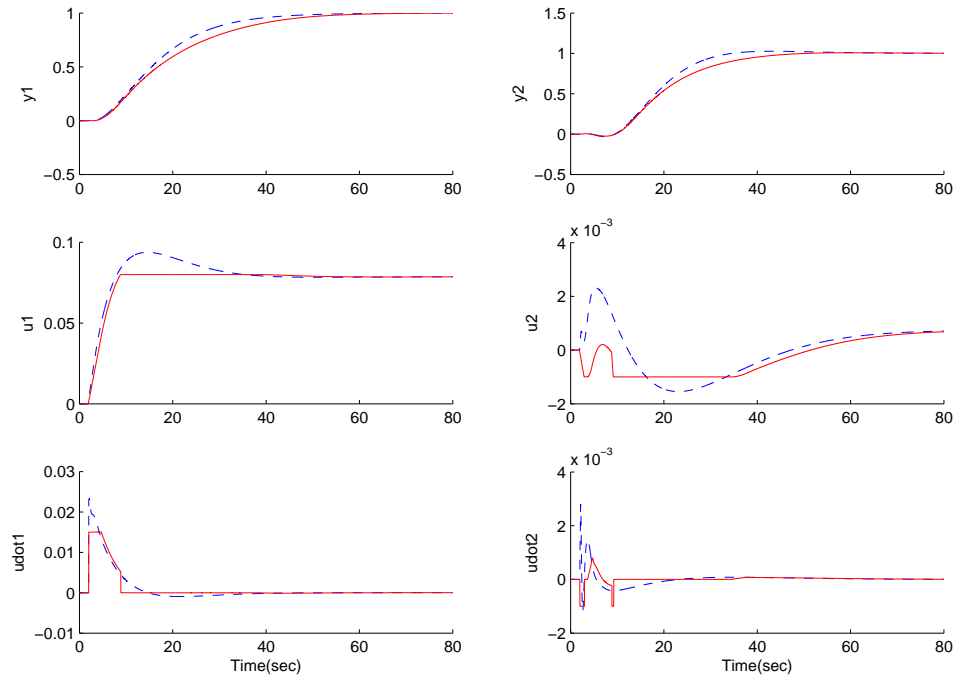


Figure 3.1: Quadratic Programming methods used where $\tau_i = 0$, $i = 0$: Hildreth QP Procedure, Active Set method, and Primal Dual Interior Point method.

Chapter 3. Review Quadratic Programming Methods

Case 2: Shor's r-Algorithm, $\tau_0 = 0.01$

The purpose of the simulation is to investigate the performance of the Shor's r-Algorithm. The tuning parameters of the MPC algorithm are chosen as $p_1 = p_2 = 1/15$, $T_{p1} = T_{p2} = 150$, and $N_1 = N_2 = 3$. The future time instant $\tau_i = 0.01$ and $i = 0$. The imposed constraints are:

$$\begin{aligned} -0.0015 &\leq \dot{u}_1 \leq 0.015 & -0.001 &\leq \dot{u}_2 \leq 0.003 \\ -0.003 &\leq u_1 \leq 0.08 & -0.001 &\leq u_2 \leq 0.001 \end{aligned}$$

Conclusion: The Shor's r-Algorithm handles well the constrained control problem. However, noise happens at the derivative of the inputs.

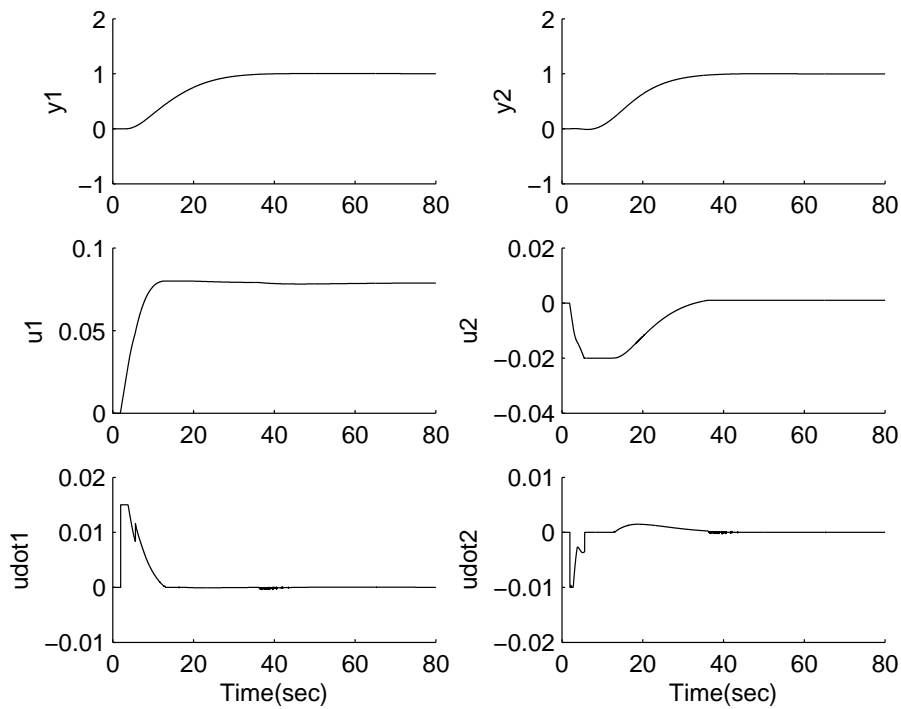


Figure 3.2: Quadratic Programming method used where $\tau_i = 0$, $i = 0$: Shor's r-Algorithm

3.6 Case Studies

In the above simulations, the constraints are imposed into the prediction at current time where $\tau_i \leq T_p$, $i = 0$. In this section, the constraints are imposed into the future prediction of the MPC where $\tau_i \leq T_p$, $i = 0, 1, 2, \dots$. The purpose is to enhance the accuracy of the prediction of the MPC method. However, the constrained set is not only larger, but also change with time. The QP methods are investigated in order to find a suitable method to handle this case.

The future time instant is chosen as $\tau_i = 0.01$, $i = 0, 1, 2$. There are 24 constraints imposed into the model. At each τ_i , the high and low limits of control inputs and its derivative are reduced 0.01 in amplitude. The imposed constraints in both cases are:

$$M\eta \leq \gamma \quad (3.31)$$

where: δ_1 and δ_2 are constant and are chosen as 0.01 in this case. Subscripts 'h' and 'l' of the input u represent for high and low, respectively.

$$M = \begin{bmatrix} L_1^T & 0 \\ -L_1^T & 0 \\ (e^{A_p \tau_1} L_1)^T & 0 \\ -(e^{A_p \tau_1} L_1)^T & 0 \\ (e^{A_p \tau_2} L_1)^T & 0 \\ -(e^{A_p \tau_2} L_1)^T & 0 \\ 0 & L_2^T \\ 0 & -L_2^T \\ 0 & (e^{A_p \tau_1} L_2)^T \\ 0 & -(e^{A_p \tau_1} L_2)^T \\ 0 & (e^{A_p \tau_2} L_2)^T \\ 0 & -(e^{A_p \tau_2} L_2)^T \\ L_1^T \Delta t & 0 \\ -L_1^T \Delta t & 0 \\ (A_p^{-1}[(e^{A_p \tau_1} - I)L_1])^T \Delta t & 0 \\ -(A_p^{-1}[(e^{A_p \tau_1} - I)L_1])^T \Delta t & 0 \\ (A_p^{-1}[(e^{A_p \tau_2} - I)L_1])^T \Delta t & 0 \\ -(A_p^{-1}[(e^{A_p \tau_2} - I)L_1])^T \Delta t & 0 \\ 0 & L_2^T \Delta t \\ 0 & -L_2^T \Delta t \\ 0 & (A_p^{-1}[(e^{A_p \tau_1} - I)L_2])^T \Delta t \\ 0 & -(A_p^{-1}[(e^{A_p \tau_1} - I)L_2])^T \Delta t \\ 0 & (A_p^{-1}[(e^{A_p \tau_2} - I)L_2])^T \Delta t \\ 0 & -(A_p^{-1}[(e^{A_p \tau_2} - I)L_2])^T \Delta t \end{bmatrix}$$

Chapter 3. Review Quadratic Programming Methods

$$\gamma = \begin{bmatrix} \dot{u}_{1h} \\ -\dot{u}_{1l} \\ \dot{u}_{1h} - \delta_1 \\ \dot{u}_{1l} + \delta_1 \\ \dot{u}_{1h} - 2\delta_1 \\ \dot{u}_{1l} + 2\delta_1 \\ \dot{u}_{2h} \\ -\dot{u}_{2l} \\ \dot{u}_{2h} - \delta_1 \\ \dot{u}_{2l} + \delta_1 \\ \dot{u}_{2h} - 2\delta_1 \\ \dot{u}_{2l} + 2\delta_1 \\ u_{1h} - u_1(t) \\ -u_{1l} + u_1(t) \\ u_{1h} - u_1(t + \tau_1) - \delta_2 \\ -u_{1l} + u_1(t + \tau_1) + \delta_2 \\ u_{1h} - u_1(t + \tau_2) - 2\delta_2 \\ -u_{1l} + u_1(t + \tau_2) + 2\delta_2 \\ u_{2h} - u_2(t) \\ -u_{2l} + u_2(t) \\ u_{2h} - u_2(t + \tau_1) - \delta_2 \\ -u_{2l} + u_2(t + \tau_1) + \delta_2 \\ u_{2h} - u_2(t + \tau_2) - 2\delta_2 \\ -u_{2l} + u_2(t + \tau_2) + 2\delta_2 \end{bmatrix} \quad (3.32)$$

From the simulation, the Shor' r-Algorithm can not produce feasible solution. The simulation results of the MPC using Hildreth Quadratic Programming Procedure and Active Set method are shown in the figure 3.3. The simulation results of the Interior Point Primal Dual method are very fluctuated in the derivative of the control input. However, it still produce a good estimation of the outputs. The result of this method is shown in the figure 3.4.

3.6 Case Studies

Case 3: The Active Set method and Hildreth QP Method, $\tau_i = 0.01, i = 0, 1, 2$

The set of constraints are enlarged in order to verify the ability of handling constraints of the QP methods. The Active Set method, Primal Dual Interior Point (DPIP) method, and Hildreth QP Procedure are used. The tuning parameters of the MPC algorithm are chosen as $p_1 = p_2 = 1/15$, $T_{p1} = T_{p2} = 150$, and $N_1 = N_2 = 3$. The future time instant $\tau_i = 0.01, i = 0, 1, 2$, is chosen to apply to the cost function 2.24. The constrained limits are the same as in the equation 3.31.

Observation: The Active Set method and Hildreth's Quadratic Programming method produce the same results. However, the results are different which those in case 1.

Conclusion: The Active Set method and Hildreth's QP method can handle the constraints well.

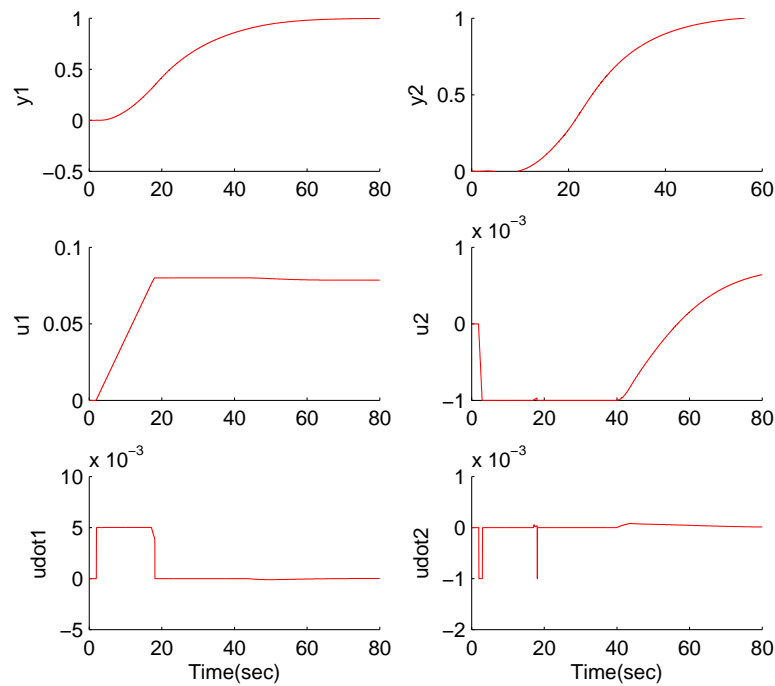


Figure 3.3: Quadratic Programming methods where $\tau_i = 0, i = 0, 1, 2$: Active Set method and Hildreth's QP method are used

Chapter 3. Review Quadratic Programming Methods

Case 4: Primal Dual Interior Point Method, $\tau_i = 0.01$, $i = 0, 1, 2$

The Primal Dual Interior Point Method is used to handle the large scale constraints. The tuning parameters of the MPC algorithm are chosen as $p_1 = p_2 = 1/15$, $T_{p1} = T_{p2} = 150$, and $N_1 = N_2 = 3$. The future time instant $\tau_i = 0.01$, $i = 0, 1, 2$, is chosen to apply to the cost function 2.24. The constrained limits are as in equation 3.31.

Observation: The responses of the inputs and outputs are acceptable, even though there are fluctuations which are appeared at the derivative of the inputs

Conclusion: The PDIP method shows the limit in handling the constraints which vary with time.

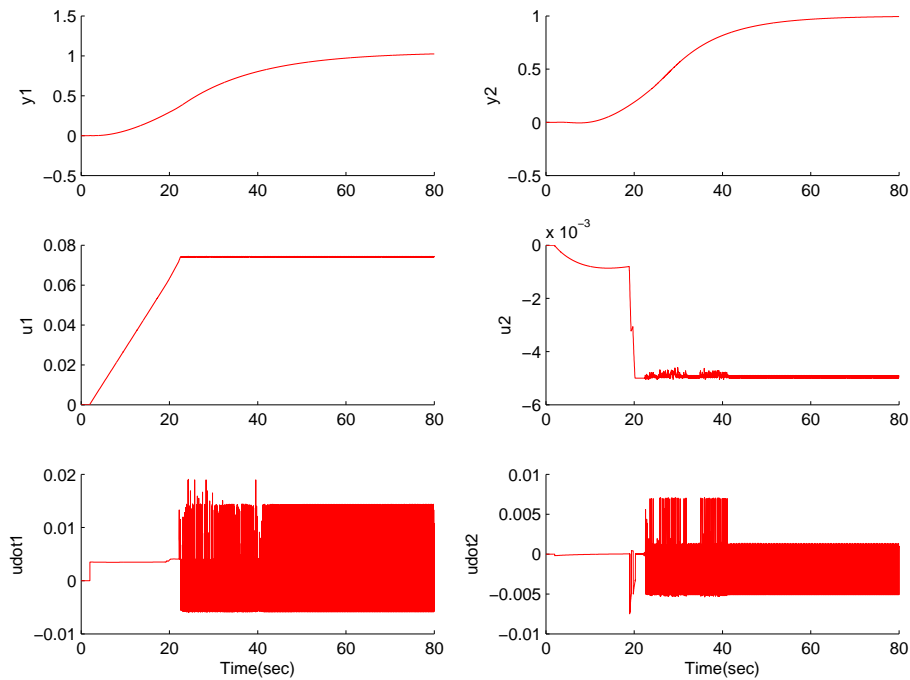


Figure 3.4: Quadratic Programming method where $\tau_i = 0$, $i = 0, 1, 2$: Interior Point Primal Dual method is used

3.7 Summary

The Shor's r-Algorithm can be applied to the MPC to handle either the linear or nonlinear constrained problems where the constrained set is fixed. Although the method uses the linear search procedure as in the PDIP method, the stopping criterion, the step size strategy, and the heuristic procedures cause the serious problem. The process does more iterations and takes more time to obtain the feasible solution. In case of the constrained set varying with time and affected by the inputs, the Shor's r-Algorithm is not able to obtain the feasible solutions for the MPC methods. It is obvious that the algorithm is not a good choice for the optimization process on-line. However, it may result very robust, efficient, and accurate in the predicting solution.

In the small and medium scale constrained set, the primal-dual interior-point (PDIP) method requires more time for calculating than the active set method, even though it still delivers the accurate solutions. The reason is that its procedure does the Mehrotra's predictor-corrector search direction for two times. If considering the solution (3.17), the matrices E and $(E + M^T \lambda M)$ are required to be positive definite. For that reason, the method is used to solve the large scale constraints. Further more, the PDIP method uses the linear search direction and the Cholesky factorizations to maintain the positive defined of some matrices. The PDIP method is not efficient to handle those constraints, which change with time and affected by other elements, such as the inputs. However, the advantage of the PDIP method is to handle few thousands of constraints. The method is applied to the MPC to solve the large scale applications such as chemical industries, data reconciliation, optimal control, multi-period design and process optimization problems.

The active set method is the best suit for the MPC design using Laguerre functions, where the problems have the small to medium scale and the constraints, are fixed or changed during iteration. The method eliminates the conflict constraints so that the solution is converged quickly. However, in the cases such as there are degenerate or ill-conditioned, and especially large scale constraints, the active set method is not a good choice. The reason is while eliminating the inactive constraints, the feasible solutions may be eliminating. It results the incorrect solution.

The Hildreth's Quadratic Programming Procedure is easy for computer programming while calculating the Lagrange multipliers directly rather than estimating. It works well and also suitable for the MPC method. However, its performances may be changed depend on applications because the non-negative Lagrange multipliers will slow down the prediction of the MPC.

Chapter 3. Review Quadratic Programming Methods

The process also produces unreliable solution in the large scale problems.

There are major steps were used to implement the QP algorithm, such as QR and Cholesky factorizations, step size strategy, stopping criterion, and heuristic procedures. Similarly, other excellent properties, which are tuning methods, pole placement methods and disturbance rejection, are used in MPC. These methods are able to guarantee the system stability and improve the performance of the QP solver as well as the MPC method.

Chapter 4

Intermittent Continuous-time MPC

4.1 Introduction

In this section, a new intermittent continuous time model predictive control is introduced. The approach is based on the Model Predictive Control design using Laguerre functions and Intermittent Control (IMPC). The approach is called an Intermittent Model Predictive Control and is shown in section 4.2. A coupled tank model is derived and used as a case study for the IMPC. This work is shown in section 4.4. The properties and performance of the IMPC are investigated based on the simulation results which is presented in section 4.5. The general conclusion of this chapter is in section 4.7.

4.2 Intermittent Continuous-time Model Predictive Control

This section develops a method that allows the model predictive control to work in the conditions where the system has the short time constant and slow sampling interval. The characteristic in this case is the time to complete the i^{th} optimisation Δ_{ol} , which is smaller than the measurement time interval Δ_i from t_i to t_{i+1} . For this reason, as mentioned above, there is a gap of time that no information is available. That means, the system can not be updated and the control inputs can not be developed. To overcome

Chapter 4. Intermittent Continuous-time MPC

the problem, we try to construct an unknown control signal by using the prediction of the optimal results. The prediction is based on the orthonormal expansion and Laguerre functions. The concept is now described as below.

Without hard constraints on the variables, the solution of the cost function (2.19) is given as:

$$\eta = \Pi^{-1}(\Psi_1 r(t_i) - \Psi_2 X(t_i)) \quad (4.1)$$

The advantage of this solution is that the matrices Π , Ψ_1 and Ψ_2 are independent of time t_i . So, η varies together with the changes of the setpoint $r(t_i)$ and state variables $X(t_i)$.

By applying the principle of receding horizon control, the derivative of the optimal control signal is:

$$\dot{u}(t_i) = \begin{bmatrix} L_1(0)^T & 0 & \dots & 0 \\ 0 & L_2(0)^T & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & L_r(0)^T \end{bmatrix} \Pi^{-1}(\Psi_1 r(t_i) - \Psi_2 X(t_i)) \quad (4.2)$$

In the unknown information section of our intermittent control theorem, the first control signal $\dot{u}(0)$ is the optimal results which the optimisation is completed. The next information is predicted based on the optimal control signal by using Laguerre functions.

$$\begin{aligned} \dot{u}_1(0) &= L^T(0)\eta \\ \dot{u}_1(\Delta t) &= L^T(\Delta t)\eta \\ \dot{u}_1(2\Delta t) &= L^T(2\Delta t)\eta \end{aligned} \quad (4.3)$$

where:

$$L(n\Delta t) = e^{nA_p\Delta t} L(0)$$

Δt is the fast sampling interval.

The concept can be described as the figure 4.1.

Now, we only need to integrate the derivative of control signal to reveal the control law:

$$u(t) = \int_0^t \dot{u}(\tau) d\tau$$

The IMPC is also developed to optimise the process on-line. The IMPC is very useful for this purpose because the results can be produced in short period of time. The model of the IMPC is constructed in Matlab Simulink and is shown in figure 4.2.

4.3 Design IMPC for Online Simulation

The IMPC approach is constructed in the Matlab Simulink model in order to control and monitor the process plant on-line. The IMPC is programmed in the Matlab function. The function is imported into the embedded function block in Simulink. The advantages of the function block are:

- build stand-alone simulation applications. It supports the programmed code in Real Time.
- easy to use. It inherits the properties of Simulink and m-files.

The system has three inputs which are the setpoint, open-loop time interval and sampling time interval, and outputs. The function block is shown in figure 4.2.

Chapter 4. Intermittent Continuous-time MPC

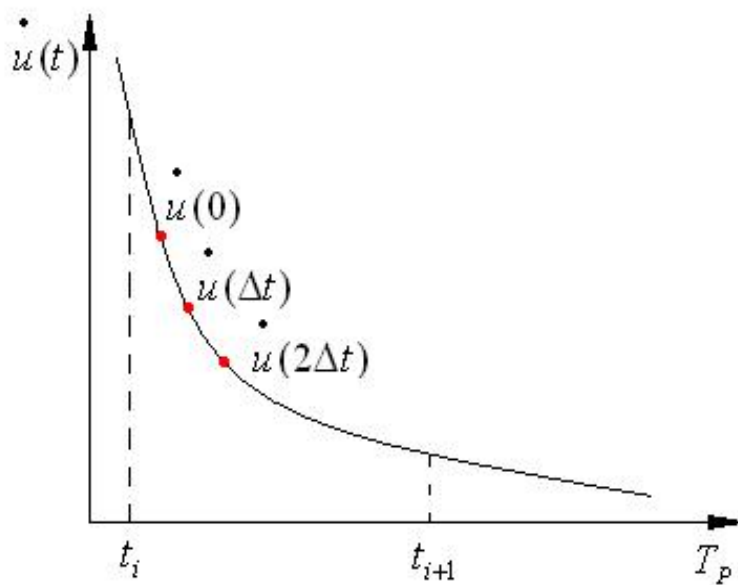


Figure 4.1: Intermittent MPC.

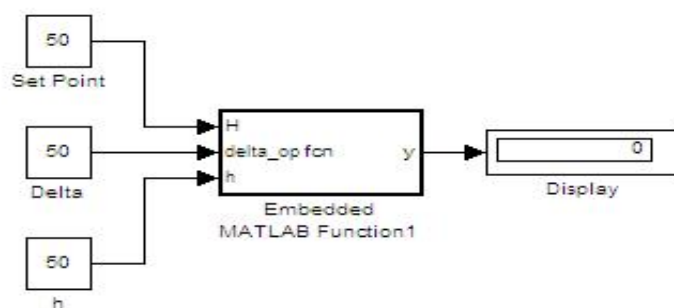


Figure 4.2: Intermittent MPC on-line.

4.4 Case Studies: A Coupled Tanks Model

4.4 Case Studies: A Coupled Tanks Model

The system model is determined by the relation among the input flow-rate q_{in} , output flow-rate q_{out} , which leaving the valves and the water level h . The coupled tank system is formed when the tanks are combined together. The system has only one output which means valve 1 is closed and valve 2 opens. In figure 4.3, the three tanks system is shown. However, one communication valve is closed so that the system can be treated as a doubled tank system. The operation of the tanks is also depicted. For simplicity in calculation but not losing the meaning of modelling, we can assume that the surface areas of two tanks are the same. It is similar for the cross section areas of the valves. We now consider the level in each tank.

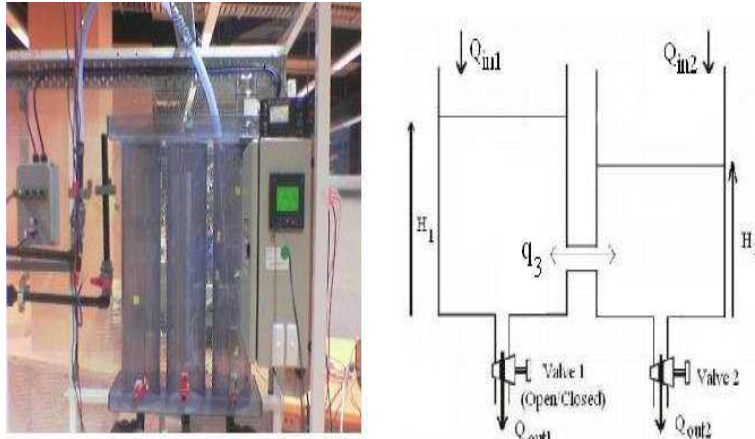


Figure 4.3: Water tank apparatus.

For tank 1:

$$A \frac{dh_1}{dt} = q_{in1} - q_3 \quad (4.4)$$

and tank 2:

$$A \frac{dh_2}{dt} = q_{in2} + q_3 - q_{out2} \quad (4.5)$$

The mathematical equations for flow-rate elements:

$$q_3 = A_v \sqrt{2g|h_1 - h_2|} \quad (4.6)$$

$$q_{out} = A_v \sqrt{2gh_2} \quad (4.7)$$

The overall equations are:

$$A \frac{dh_1}{dt} = q_{in1} - A_v \sqrt{2g|h_1 - h_2|} \quad (4.8)$$

Chapter 4. Intermittent Continuous-time MPC

$$A \frac{dh_2}{dt} = q_{in2} + A_v \sqrt{2g|h_1 - h_2|} - A_v \sqrt{2gh_2} \quad (4.9)$$

where $A_v = 2.45 * 10^{-4} \text{ m}^2$, $A = 0.0176 \text{ m}^2$, $g = 9.81 \text{ m/s}^2$, $h_0 = 0.05 \text{ m}$.

Applying the Taylor series for the square roots terms on the right hand side of the above equations, the state space model of the coupled tanks system is obtained:

$$\begin{bmatrix} \frac{dh_1}{dt} \\ \frac{dh_2}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{A_v}{A} \sqrt{\frac{g}{2h_{12}}} & \frac{A_v}{A} \sqrt{\frac{g}{2h_{12}}} \\ \frac{A_v}{A} \sqrt{\frac{g}{2h_{12}}} & -\frac{A_v}{A} \sqrt{\frac{g}{2}} \left(\frac{1}{\sqrt{h_{12}}} + \frac{1}{\sqrt{h_0}} \right) \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{A} & 0 \\ 0 & \frac{1}{A} \end{bmatrix} \begin{bmatrix} q_{in1} \\ q_{in2} \end{bmatrix} \quad (4.10)$$

where: h_{12} is the different between the desired of levels h_1 and h_2 . h_0 is the initial level. $h_{12}, h_0 \neq 0$. We measure the height of the tank in percentage where 0% is empty and 100% is full.

4.5 Simulation Studies

In the system, the diameters of the cylindrical tanks are 15 cm. The valve which is 2.5 cm in diameter is opened 50%. The water levels are controlled so that the level in tank 1 is 45% and 40% for tank 2. The pole assignment technique is used. In most of the cases in this paper, the poles are placed at [-0.13 -0.63 -1.13 -1.63]. The parameter N which used in Laguerre model to capture the control signal is set 3 and the predicted horizon T_p is 100. The open loop time interval is 4 seconds and the sampling rate is 0.02 seconds.

In the simulation, the number of parameters N are chosen from 1, 3 to 9. From figure 4.5, it can conclude that the observing closed loop is stability for all parameters N. The choices of parameters p, which used to assign the pole locations, are important. To investigate the effects of parameters p, the input and output responses are depicted in figure 4.4 with varying parameters p from $\frac{1}{30}$ (slow response) to $\frac{10}{30}$ (fast response). The specific procedures for choosing the parameters p can be found in [43] and [45]. It can be seen that the water levels and control signals response faster with higher values of either parameters N or p.

Unlike the model predictive control approach, the open-loop time interval and the sampling time interval are introduced to the intermittent model predictive control approach. It is important to know the influences of those parameters. After the simulation, it is concluded that those parameters do not strongly affect to the responses of the system. According to figure 4.7, the open-loop time interval causes the a small changing at the input but

4.6 Simulation Results

not significant at the output responses. The figure 4.8 shows the responses of the system with different values of the sampling rates. The input and output responses are almost the same for all cases. Another new parameter related to the derivative of the control signals that does not affect to the prediction is the number of steps that predicted ahead. The simulation results are plotted in figure 4.6. The simulation results are the evidences to show that IMPC approach inherits most of the properties of the MPC.

The disturbance rejection and the constrained control of the IMPC are investigated. The step input disturbances, which 0.15 liter/sec in magnitude, are added to the system at time $t = 10$ seconds in order to consider how the system detects noises. The figure 4.9 shows the comparison of the results with and without disturbances. It can be seen that the large disturbances affect to the closed-loop response, especially the control signals. By changing the gain of the observer, the effect of the disturbance can be reduced. The parameter p is increased 10 times to assign the poles at $[-1.33 -6.33 -11.33 -16.33]$. The effects of disturbances are significant reduced and the response speed are also faster. The results are shown in figure 4.10 where the sampling rate is 0.02 seconds and the open loop is 4 seconds.

The constrained IMPC to the multivariable system is applied to the equation 4.10. The parameters p and N are $\frac{1}{15}$ and 3, respectively. The open-loop time interval is 2 seconds and sampling rate is 0.02 seconds. The imposed constraints are:

$$-0.009 \leq \dot{u}_1 \leq 0.016$$

$$-0.01 \leq \dot{u}_2 \leq 0.02$$

$$0 \leq u_1 \leq 0.58$$

$$0 \leq u_2 \leq 1.532$$

and the comparison of the constrained and unconstrained problems are shown in the figure 4.11.

4.6 Simulation Results

Chapter 4. Intermittent Continuous-time MPC

Case 1: Effects of the pole location tuning parameters \mathbf{p}

The purpose of the first simulation is to investigate the effects of the pole location parameters \mathbf{p} on the system responses. The prediction horizon parameters T_p and tuning parameters N are kept constant.

$$T_{p1} = T_{p2} = 100, \quad N_1 = N_2 = 3$$

The corresponding parameters \mathbf{p} are chosen as:

$$\text{Case1} : p_1 = p_2 = 1/30$$

$$\text{Case2} : p_1 = p_2 = 2/30$$

$$\text{Case3} : p_1 = p_2 = 10/30$$

Observation: The response speed in case 1 is slow. The response speed in case 2 is normal. The response speed in case 3 is fast.

Conclusion: Closed loop responses are faster while increasing the pole placement parameters \mathbf{p} .

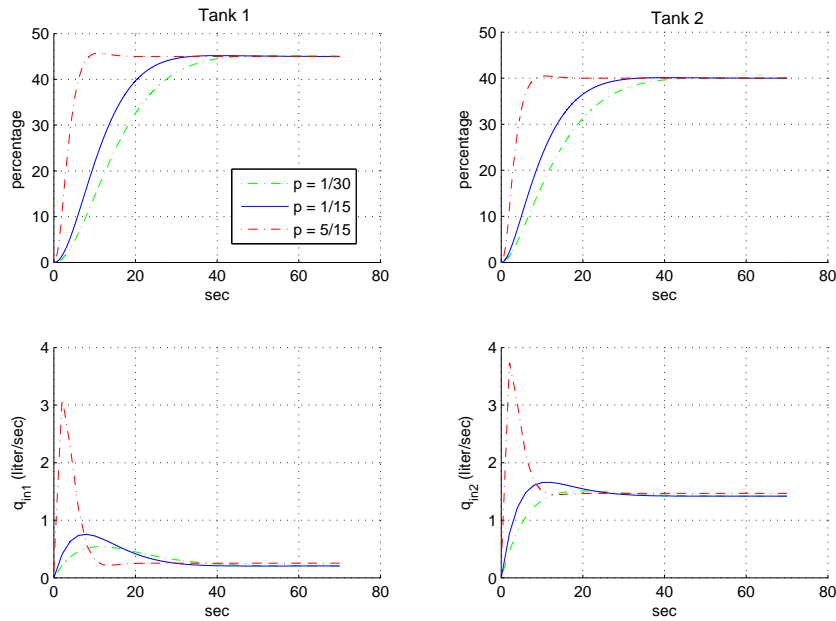


Figure 4.4: The water level control with slow and fast response speed. Parameters \mathbf{p} change from $\frac{1}{30}$, $\frac{2}{30}$ to $\frac{10}{30}$.

4.6 Simulation Results

Case 2: Effects of the tuning parameters N

The purpose of this simulation is to investigate the effects of the tuning parameters N on the system responses. The tuning parameters T_p and p are kept constant.

$$T_{p1} = T_{p2} = 100, \quad p_1 = p_2 = 1/15$$

The parameters N are chosen as the following:

$$\text{Case1} : N_1 = N_2 = 1$$

$$\text{Case2} : N_1 = N_2 = 3$$

$$\text{Case3} : N_1 = N_2 = 9$$

Observation: The response speed in case 1 is slow. The response speed in case 2 is normal. The response speed in case 3 is fast.

Conclusion: Closed loop responses are faster while increasing the tuning parameters N.

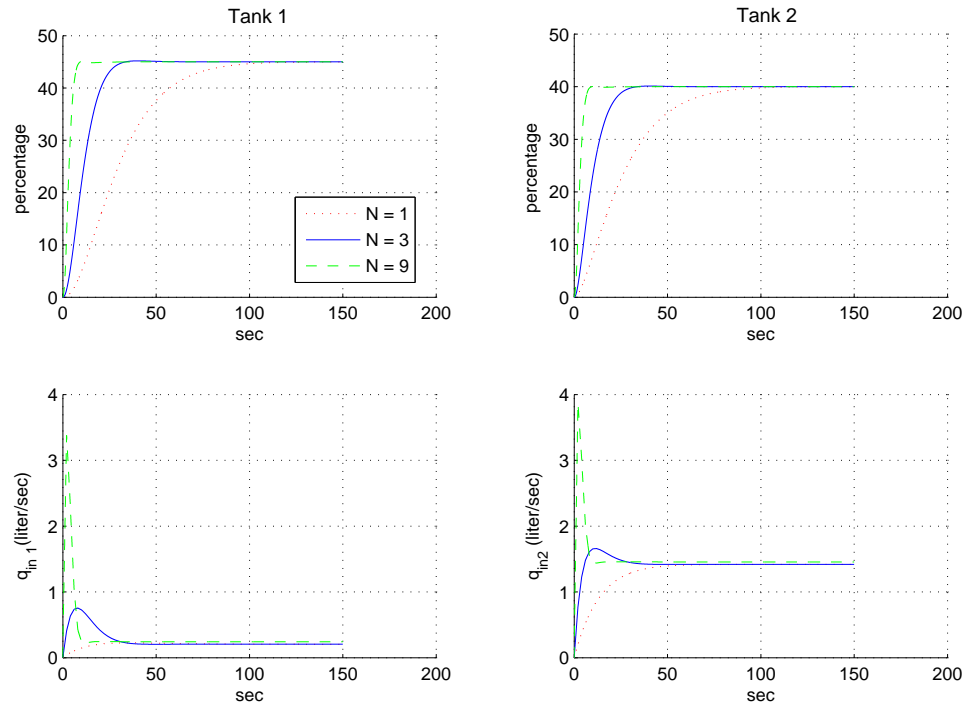


Figure 4.5: The water level control with slow and fast response speed. Parameters N change from 1, 3 to 9.

Chapter 4. Intermittent Continuous-time MPC

Case 3: Effects of the number of ahead prediction

The purpose of this simulation is to investigate the effects of the number of ahead prediction on the system response. The tuning parameters T_p , p , and N are kept constant.

$$T_{p1} = T_{p2} = 100, \quad p_1 = p_2 = 1/15, \quad N_1 = N_2 = 3$$

The system is predicted 1, 3 and 9 steps ahead.

Conclusion: The IMPC produces accurate results and a good prediction.

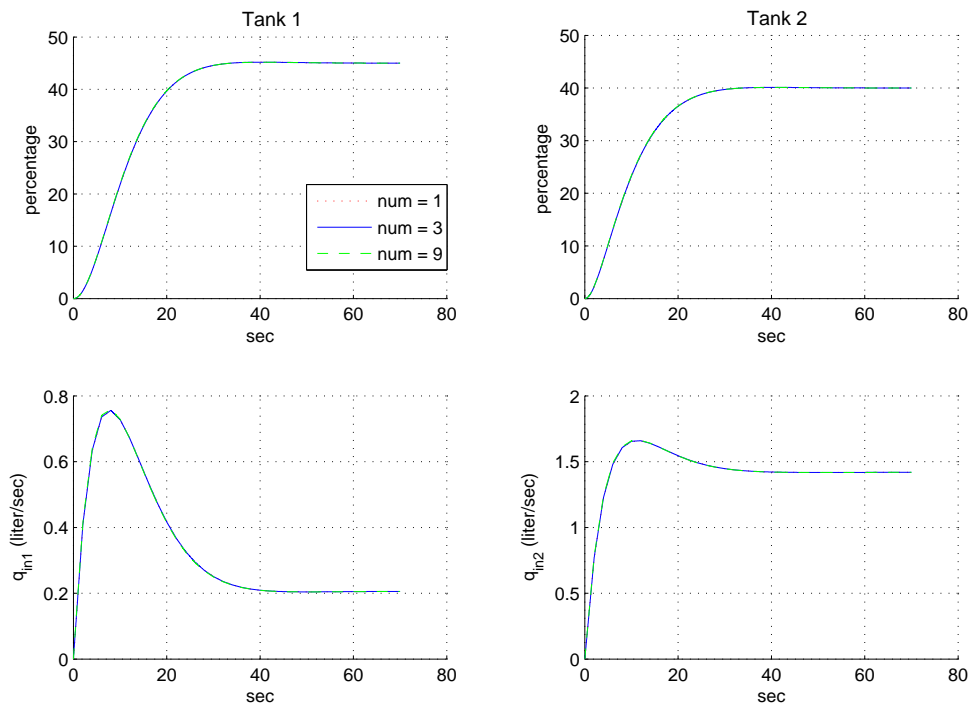


Figure 4.6: Continuous IMPC results with different number of ahead prediction. The derivative of control signal is predicted with 1, 3, and 9 steps ahead.

4.6 Simulation Results

Case 4: Effects of the open-loop time interval

The purpose of this simulation is to investigate the effects of the open-loop time interval on the system response. The tuning parameters T_p , p , and N are kept constant.

$$T_{p1} = T_{p2} = 100, \quad p_1 = p_2 = 1/15, \quad N_1 = N_2 = 3$$

The open loop time intervals (OL) are chosen as follow:

Case1 : $OL = 0.5\text{sec}$

Case2 : $OL = 1\text{sec}$

Case3 : $OL = 2\text{sec}$

Case4 : $OL = 4\text{sec}$

Conclusion: The response of the system is faster and less smoothness when increasing the open-loop time interval.

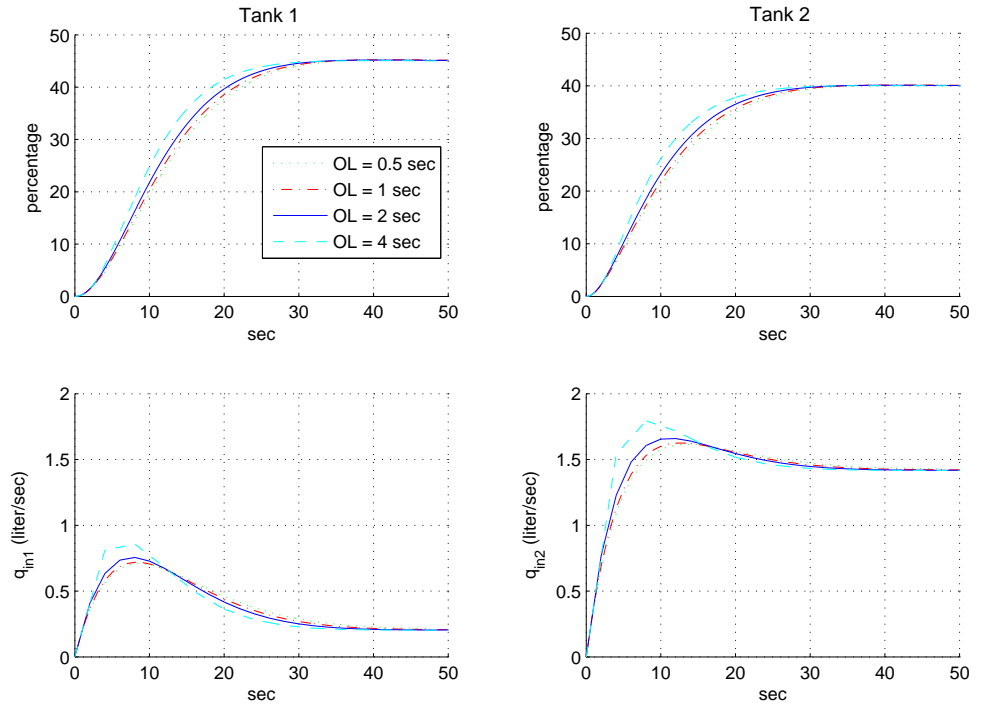


Figure 4.7: Continuous time IMPC results with different open loop time intervals. The open loop time intervals are 0.5, 1, 2, and 4 seconds. The sampling time interval is 0.02 second.

Chapter 4. Intermittent Continuous-time MPC

Case 5: Effects of the sampling time interval

The purpose of this simulation is to investigate the effect of the sampling time interval on the system response. The tuning parameters T_p , p , and N are kept constant.

$$T_{p1} = T_{p2} = 100, \quad p_1 = p_2 = 1/15, \quad N_1 = N_2 = 3$$

The sampling time intervals (SI) are chosen as follow:

Case1 : $SI = 0.005\text{sec}$

Case2 : $SI = 0.01\text{sec}$

Case3 : $SI = 0.02\text{sec}$

Case4 : $SI = 0.1\text{sec}$

Conclusion: The sampling time interval does not show a strong effect on the system response.

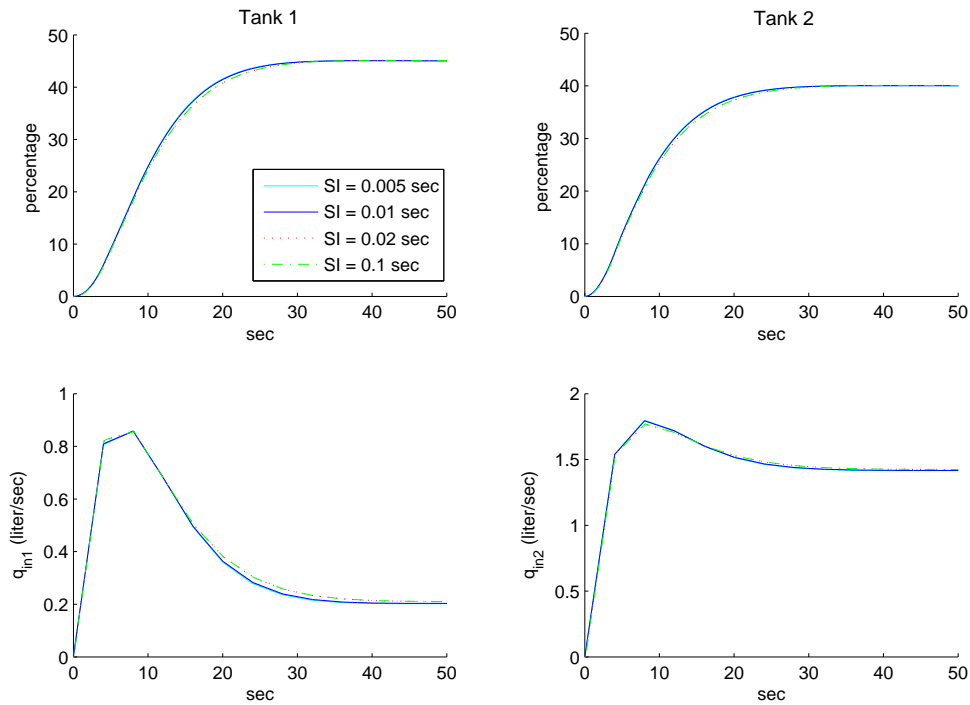


Figure 4.8: Continuous time IMPC results with different open loop time intervals. The open loop time interval is 4 seconds. The sampling time intervals are 0.005, 0.01, 0.02, and 0.1 seconds.

4.6 Simulation Results

Case 6: Design Observer

The purpose of the simulation is to investigate how the IMPC approach handles the disturbance of the system.

The high disturbances with 0.15 liter/sec in magnitudes are added to the input of the system from the time $t = 10$ seconds. The tuning parameters $T_{p1} = T_{p2} = 100$, $N_1 = N_2 = 3$, and $p_1 = p_2 = 1/15$.

Conclusion: The IMPC approach can handle well the process when the disturbances are in the system.

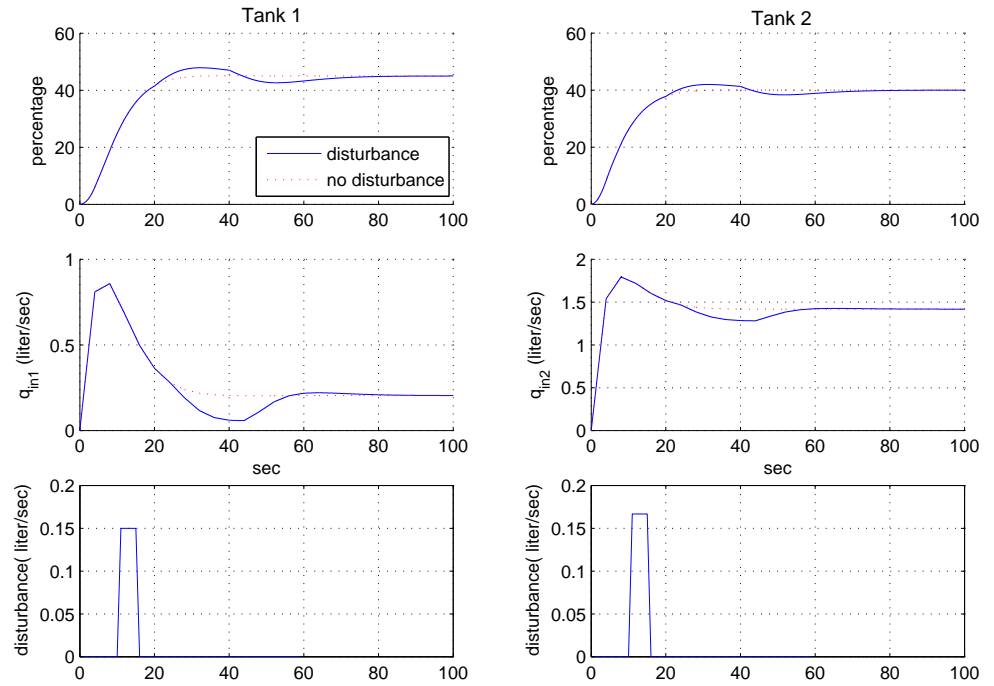


Figure 4.9: The continuous time IMPC with and without step input disturbances.

Chapter 4. Intermittent Continuous-time MPC

Case 7: Design Observer

The purpose of the simulation is to show how to improve the performance of the IMPC under disturbance by using the observer.

The high disturbances with 0.15 liter/sec in magnitudes are added to the input of the system from the time $t = 10$ seconds. The tuning parameters $T_{p1} = T_{p2} = 100$, $N_1 = N_2 = 3$. The pole location parameters is now chosen as $p_1 = p_2 = 10/15$.

Conclusion: The disturbances at the outputs are reduced.

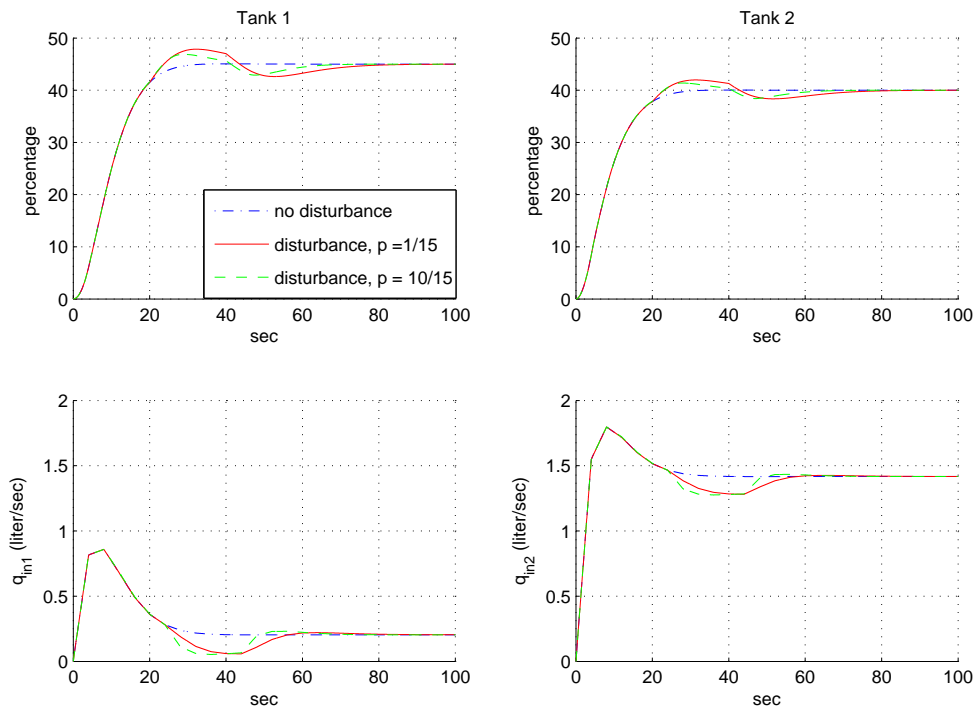


Figure 4.10: The water level control with higher observer gain. The parameters p change from $\frac{1}{15}$ to $\frac{10}{15}$.

4.6 Simulation Results

Case 8: Constrained Control

The simulation shows how the MPC handle the constrained problems. The tuning parameters are kept the same as in case 4. The imposed constraints are:

$$-0.009 \leq \dot{u}_1 \leq 0.016$$

$$-0.01 \leq \dot{u}_2 \leq 0.02$$

$$0 \leq u_1 \leq 0.58$$

$$0 \leq u_2 \leq 1.532$$

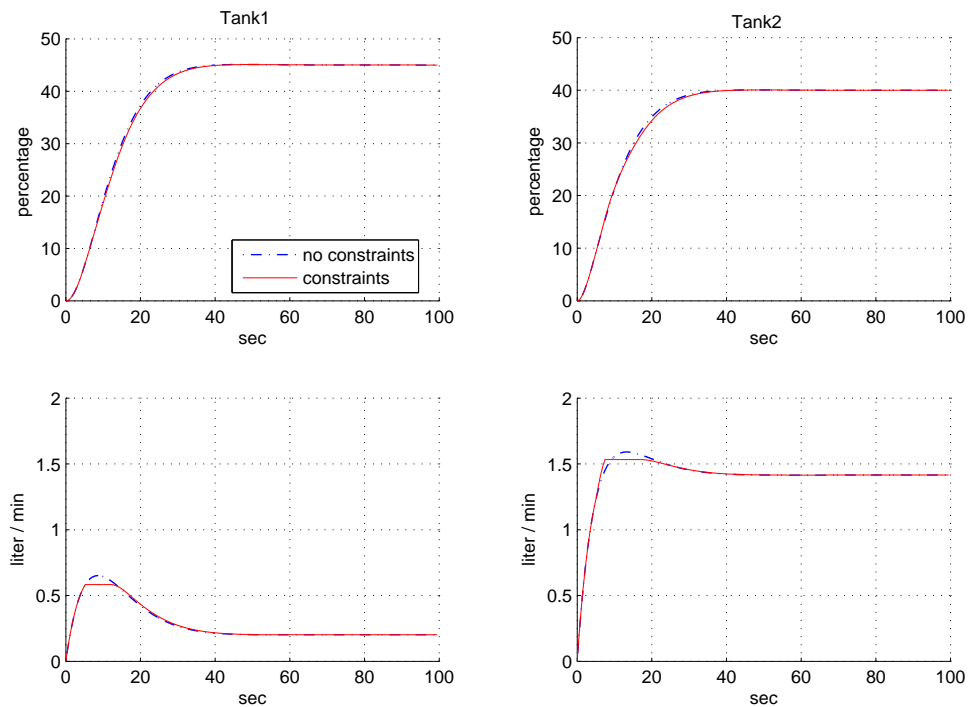


Figure 4.11: Continuous time IMPC with constraints. Sampling time interval = 0.02 seconds and open-loop time interval is 2 seconds.

Chapter 4. Intermittent Continuous-time MPC

4.7 Conclusions

The continuous time intermittent model predictive control approach inherits all the properties of the continuous time model predictive control approach such as easy to tune the poles, number of parameters N , and the predicted horizon. It also can handle well the disturbances and the control of the constraints. Three new parameters that introduced in IMPC are the open loop time interval, the sampling rate, and the steps predicted ahead of the derivative of the control signal. However, those parameters are slightly influent to the predictions.

Chapter 5

Nonlinear Model Predictive Control

5.1 Introduction

This chapter proposed the two new Nonlinear Model Predictive Control (NMPC) designs. The first design uses the Lie's functions to transform the nonlinear model to the Brunowski form. The Laguerre functions is then used for prediction. The nonlinear inequality constrained set is derived. Some other properties of this design, such as unknown parameters and disturbance rejection, are also investigated. The nonlinear MPC design is shown in section 5.2.2.

The second Nonlinear MPC design uses the Taylor's series to predict the state of the nonlinear system. The optimal control strategy and constrained control problem are considered. This design is shown in section 5.4.2.

The simulation results of these two design are presented in sections 5.3 and 5.5, respectively. The conclusion is stated in section 5.6.

Chapter 5. Nonlinear Model Predictive Control

5.2 NMPC Using Laguerre Functions

We consider the MIMO system with no uncertain disturbance where $f(x)$ and $g(x)$ are smooth vector fields defined on R^n with $g(x) \neq 0$.

$$\dot{x} = f(x) + \sum_{i=1}^m g_i(x)u_i \quad (5.1)$$

$$y = [h_1, h_2, \dots, h_m]^T$$

The state variables in (5.1) can be represented as follows:

$$\dot{x} = F(x) + G(x)U$$

The system (5.1) can be transformed into a new coordinate which is linear and controllable

$$z = T(x)$$

such that the system (5.1) in z coordinates is in Brunowski form:

$$\dot{z} = Az + Bv \quad (5.2)$$

At the order k^{th} where $L_g(L_f^{r_k-1}(h_k)) \neq 0$,

$$v = \begin{bmatrix} z_1^{(n_1)} \\ \dots \\ z_m^{(n_m)} \end{bmatrix} \equiv \begin{bmatrix} \frac{d^{n_1}y_1}{dt^{n_1}} \\ \dots \\ \frac{d^{n_m}y_m}{dt^{n_m}} \end{bmatrix} = \beta(x)u + \alpha(x) \quad (5.3)$$

where $\beta(x)$ is an $(m \times m)$ and $\alpha(x)$ is an $(m \times 1)$ matrix such that:

$$\beta(x) = \begin{bmatrix} L_{g_1}(L_f^{r_1-1}(h_1)) & \dots & L_{g_m}(L_f^{r_1-1}(h_1)) \\ \dots & \dots & \dots \\ L_{g_1}(L_f^{r_m-1}(h_m)) & \dots & L_{g_m}(L_f^{r_m-1}(h_m)) \end{bmatrix} \quad (5.4)$$

and

$$\alpha(x) = \begin{bmatrix} L_f^{n_1}(h_1) \\ \dots \\ L_f^{n_m}(h_m) \end{bmatrix}$$

The linearization state feedback is:

$$u = \beta^{-1}(x)(v - \alpha(x)) \quad (5.5)$$

5.2 NMPC Using Laguerre Functions

5.2.1 Predicted Plant Response

At the current time t_i , the state variable $z(t_i)$ is available. At the future time $t_i + \tau$, the predicted state variable $z(t_i + \tau)$ is described by the equation:

$$z(t_i + \tau) = e^{A\tau} z(t_i) + \int_0^\tau e^{A(\tau-\gamma)} B v(t_i + \gamma) d\gamma \quad (5.6)$$

The control input signal is:

$$v(t) = [v_1(t), v_2(t), \dots, v_n(t)]^T$$

where the i^{th} control signal $v_i(t)$, $i = 1, 2, \dots, n$, is:

$$\begin{aligned} v_i(t) &= L_i(t)^T \eta_i \\ L_i(t)^T &= [l_1^i(t), l_n^i(t), \dots, l_n^i(t)] \\ \eta_i &= [\xi_1^i, \xi_2^i, \dots, \xi_n^i]^T \\ B &= [B_1, B_2, \dots, B_n] \end{aligned}$$

The predicted future state at time $(t_i + \tau)$ is:

$$z(t_i + \tau) = e^{A\tau} z(t_i) + \int_0^\tau e^{A\tau} [B_1 L_1(\gamma)^T, B_2 L_2(\gamma)^T, \dots, B_n L_n(\gamma)^T] d\gamma \eta \quad (5.7)$$

and the plant output prediction is:

$$y(t_i + \tau) = h(t_i + \tau)$$

Unconstrained Solution

Let us define:

$$\begin{aligned} I_{int}(\tau)^i &= \int_0^\tau e^{A(\tau-\gamma)} B_i L_i(\gamma)^T d\gamma \\ \Phi(\tau) &= (C [I_{int}(\tau)^1, I_{int}(\tau)^2, \dots, I_{int}(\tau)^n])^T \end{aligned}$$

The cost function is formulated as the function of η :

$$J = \eta^T \Pi \eta - 2\eta^T (\Psi_1 r(t_i) - \Psi_2 z(t_i)) \quad (5.8)$$

where:

$$\begin{aligned} \Pi &= \int_0^{T_p} \Phi(\tau) Q \Phi(\tau)^T d\tau + \bar{R} \\ \Psi_1 &= \int_0^{T_p} \Phi(\tau) Q d\tau & \Psi_2 &= \int_0^{T_p} \Phi(\tau) Q C e^{A\tau} d\tau \end{aligned}$$

The solution is:

$$\eta = \Pi^{-1} (\Psi_1 r(t_i) - \Psi_2 z(t_i)) \quad (5.9)$$

Chapter 5. Nonlinear Model Predictive Control

Constrained Solution

The hard limits on the control signals $u(t_i)$ are specified as:

$$u_{min} \leq u(t_i) \leq u_{max}$$

From (5.5), the constraints on $u(t_i)$ can be represented as in linear form:

$$\begin{bmatrix} \beta(x)^{-1}L^T\eta \\ -\beta(x)^{-1}L^T\eta \end{bmatrix} \leq \begin{bmatrix} u_{max} + \beta^{-1}(x)\alpha(x) \\ -u_{min} - \beta^{-1}(x)\alpha(x) \end{bmatrix} \quad (5.10)$$

The linear Quadratic Programming can solve the cost function (5.8) subject to the inequality nonlinear constraints in equation (5.10). However, the solutions may not be reliable as the system is sensitive. To overcome the problem, the functions $\beta(x)$ and $\alpha(x)$ are rewritten as the functions of η so that the constraints in equation (5.10) become nonlinear. Considering the state vector $x(t_i)$ at the time t_i :

$$\begin{aligned} x(t_i) &= x(t_i - 1) + \dot{x}(t_i - 1)\Delta(t) \\ x(t_i) &= x(t_i - 1) + \end{aligned} \quad (5.11)$$

$$[F(x(t_i - 1)) + G(x(t_i - 1))\beta(x(t_i - 1))^{-1}(x(t_i - 1))(L^T\eta - \alpha(x(t_i - 1)))]\Delta(t)$$

The functions $\beta(x(t_i))$ and $\alpha(x(t_i))$ are nonlinear as functions of η and $x(t_i - 1)$. The system is expressed as follows:

$$J = \eta^T \Pi \eta - 2\eta^T (\Psi_1 r(t_i) - \Psi_2 z(t_i))$$

subject to:

$$\begin{bmatrix} \beta(x(t_i - 1), \eta)^{-1}L^T \\ -\beta(x(t_i - 1), \eta)^{-1}L^T \end{bmatrix} \eta + \begin{bmatrix} -u_{max} - \beta(x(t_i - 1), \eta)^{-1}\alpha(x(t_i - 1), \eta) \\ u_{min} + \beta(x(t_i - 1), \eta)^{-1}\alpha(x(t_i - 1), \eta) \end{bmatrix} \leq 0 \quad (5.12)$$

5.2.2 Disturbance Rejection

This section considers the nonlinear system with uncertainties. While formulating nonlinear mathematical models from physical systems or measurements, the unknown, or not precisely known, parameters are sometimes obtained. The uncertain system which is affected by the unknown constant disturbances is investigated.

$$\dot{x} = f(x) + g(x)u + q(x, \theta) \quad (5.13)$$

$$y = h(x)$$

5.3 Case Studies for NMPC Using Laguerre Functions

the equation (5.13) can be rewritten:

$$\begin{aligned}\dot{x} &= f(x, \theta) + g(x)u \\ y &= h(x)\end{aligned}\tag{5.14}$$

where disturbance $\theta = [\theta_1(t), \theta_2(t) \dots \theta_p(t)]^T$.

- Note:

the integer control characteristic index ρ of the equation (5.14) is defined as:

$$\begin{aligned}L_g L_f^i h(x) &= 0, \quad 0 \leq i \leq \rho - 2 \\ L_g L_f^{\rho-1} h(x) &\neq 0\end{aligned}$$

and the integer disturbance characteristic index μ , ($\mu > \rho$),

is defined as:

$$\begin{aligned}L_q L_f^i h(x) &= 0, \quad 0 \leq i \leq \rho - 2 \\ L_q L_f^{\mu-1} h(x) &\neq 0\end{aligned}$$

The tracking problem is still solvable since the disturbance θ does not affect the output of closed loop system. The state feedback control is still described as:

$$u = \frac{1}{L_g L_f^{\rho-1} h(x)} (-L_f^\rho h(x) + v)$$

The tracking control input signal v in z -coordinates can be predicted by NMPC or:

$$v = k_1 y_r + \dots + k_\rho y_r^{(\rho-1)} + y_r^{(\rho)}$$

where parameters k_i , ($i = 1, \dots, \rho$), are defined by a Hurwitz polynomial:

$$s^\rho + k_\rho s^{\rho-1} + \dots + k_1$$

5.3 Case Studies for NMPC Using Laguerre Functions

Example 1: Nonlinear System and Linear Outputs

This nonlinear system of this example is taken from [81] in order to illustrate

Chapter 5. Nonlinear Model Predictive Control

how the NMPC handles nonlinear system. The system has three inputs and two linear outputs. The equation is shown as in the equation (5.15).

$$\begin{aligned}
 \dot{x}_1 &= x_2 + x_3^2 & (5.15) \\
 \dot{x}_2 &= x_1^2 + (1 + x_1^2)u_1 \\
 \dot{x}_3 &= x_2^2 + u_2 \\
 y_1 &= x_1 \\
 y_2 &= x_3
 \end{aligned}$$

By using the Lie's function, the new state space coordinates are globally defined:

$$\begin{aligned}
 z_1 &= x_1 \\
 z_2 &= x_2 + x_3^2 \\
 z_3 &= x_3
 \end{aligned}$$

Where:

$$\begin{aligned}
 \dot{z} &= Az + Bv \\
 y &= Cz
 \end{aligned}$$

and:

$$\begin{aligned}
 z &= [z_1, z_2, z_3]^T & y &= [y_1, y_2]^T & v &= [v_1, v_2]^T \\
 A &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} & B &= \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} & C &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} & (5.16)
 \end{aligned}$$

The tuning parameters p , N , and T_p are set 5, 3, and 150, respectively. The figure (5.1) shows the process outputs and control signals. The hard constraints are also included into the nonlinear system:

$$\begin{aligned}
 -7 &\leq u_1 \leq 50 \\
 -8 &\leq u_2 \leq 10
 \end{aligned}$$

The linear Quadratic Programming (QP) and nonlinear one are used to handle the constraints. Figure (5.1) shows the results for constrained control. The linear QP is also considered because it requires less computational time. The set of constraints is based on (5.10) with $\tau_i = 0$. We only impose constraints at the current time. The simulation shows that the linear QP can not handle constraints sufficiently. Even though the control signals reduce incredible, their peaks still exceed the limits. The linear QP can be used in the case that the constraints are not strict. Nonlinear QP performs well under constraints. However, the simulation incurs the long computational time.

5.3 Case Studies for NMPC Using Laguerre Functions

The effects of disturbances on closed loop responses are investigated. The unit step input and output disturbances are added to the responses at time $t = 0.2$ seconds. The amplitudes of input and output disturbances are 12 and 0.15 units, respectively and pole location parameters are $p = 6$. The figures (5.2) and (5.4) compare the results with and without the disturbances. The closed responses are influenced. Fortunately, the effects of the disturbances can be reduced by selecting the new pole location parameters p , e.g $p = 15$ as in this example, in order to decrease the gain of the controller. It can be seen that the effects of the disturbances on the closed loop responses are significantly reduced. However, it is required the increasing of control signals. The results are shown in figures (5.3) and (5.5).

Chapter 5. Nonlinear Model Predictive Control

Case 1: Constrained Control Problems

The purpose of this simulation is to investigate the ability of handling nonlinear constraints of the NMPC using Laguerre function. The linear QP and nonlinear QP methods are used. The nonlinear system as in the equation 5.15 has two inputs and two outputs. The tuning parameters are chosen as

$$T_{p1} = T_{p2} = 150 \quad N_1 = N_2 = 3, \quad p_1 = p_2 = 5$$

The imposed constraints are:

$$-7 \leq u_1 \leq 50$$

$$-8 \leq u_2 \leq 10$$

Conclusion: The NMPC method using Laguerre functions handles well the constrained problems. The nonlinear QP method performs better than linear QP in handling the nonlinear constraints. However, it requires more time for computations.

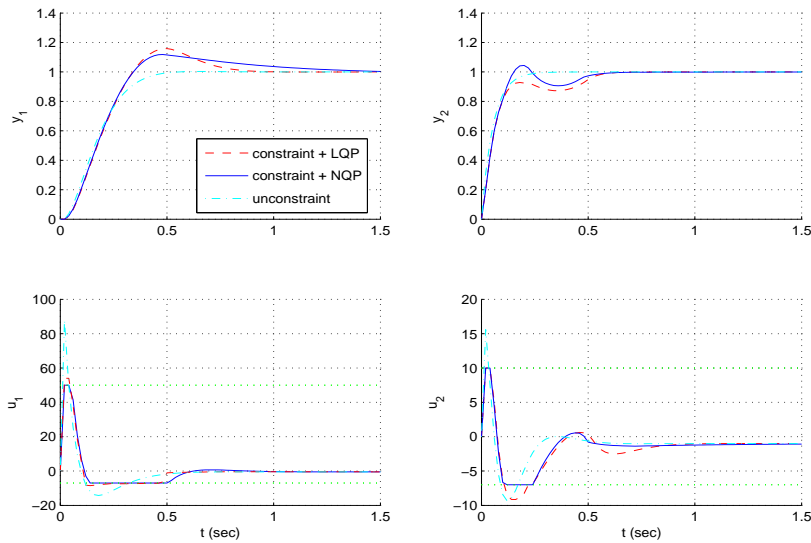


Figure 5.1: Constrained control using linear QP and nonlinear QP for linear output system ($p = 5$)

5.3 Case Studies for NMPC Using Laguerre Functions

Case 2: Disturbance Rejection: Disturbance at The Inputs

The purpose of the simulation is to investigate how the NMPC approach handles the disturbance added to the inputs of the nonlinear system. This process is known as *disturbance rejection*.

The high disturbances with 15 in magnitudes are added to the inputs of the system from the time $t = 0.1$ to 0.6 seconds. The tuning parameters $T_{p1} = T_{p2} = 150$, $N_1 = N_2 = 3$, and $p_1 = p_2 = 6$.

Conclusion: The NMPC approach can handle well the process when the disturbances are in the system.

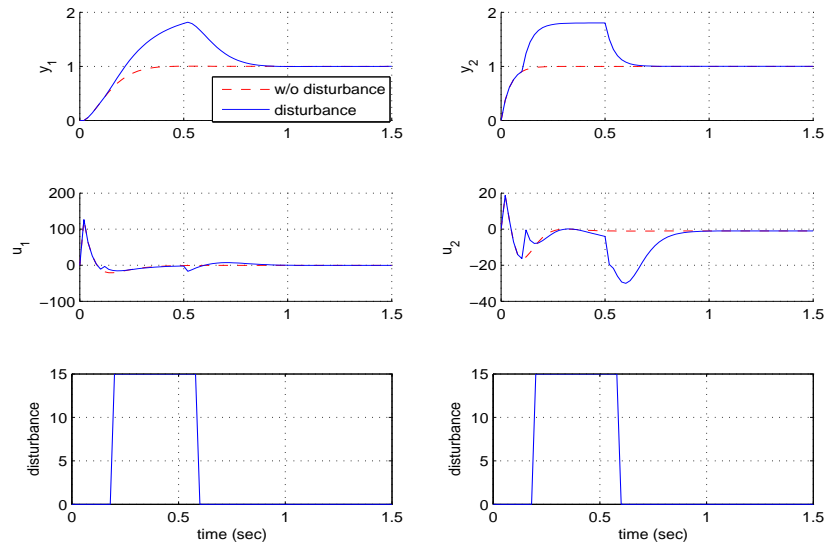


Figure 5.2: Constant disturbance at the inputs at $p = 6$

Chapter 5. Nonlinear Model Predictive Control

Case 3: Disturbance Rejection: Improve the System Response

The purpose of this simulation is to show how to reduce the effect of the disturbance on the system response.

The parameters in this case are the same as in case 2, except the pole location parameters are reduced to $p_1 = p_2 = 15$. The simulation shows that the disturbances at the outputs are reduced significantly when changing the pole location parameters.

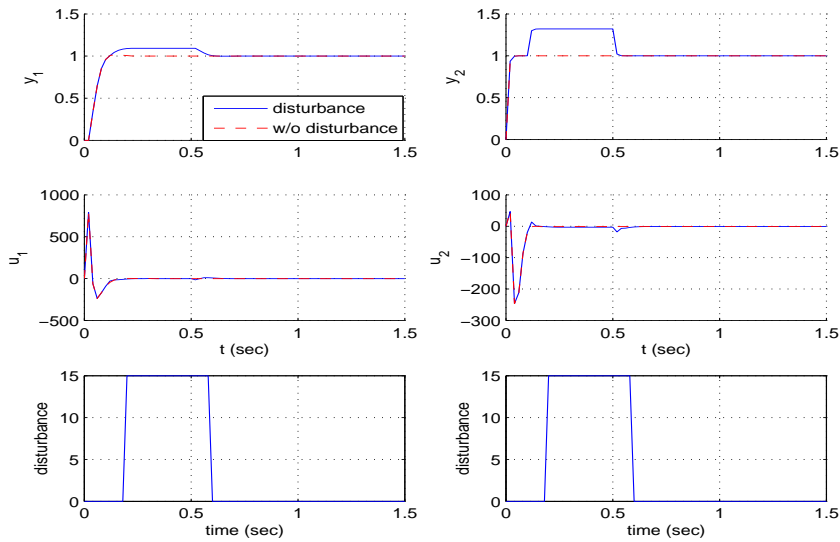


Figure 5.3: Constant disturbance at the inputs at $p = 15$

5.3 Case Studies for NMPC Using Laguerre Functions

Case 4: Disturbance Rejection: Disturbance at the Outputs

The purpose of the simulation is to investigate how the NMPC approach handles the disturbance added to the outputs of the nonlinear system.

The high disturbances with 0.15 in magnitudes are added to the outputs of the system from the time $t = 0.1$ to 0.6 seconds. The tuning parameters $T_{p1} = T_{p2} = 150$, $N_1 = N_2 = 3$, and $p_1 = p_2 = 6$.

Conclusion: The NMPC approach can handle well the process when the disturbances are in the system.

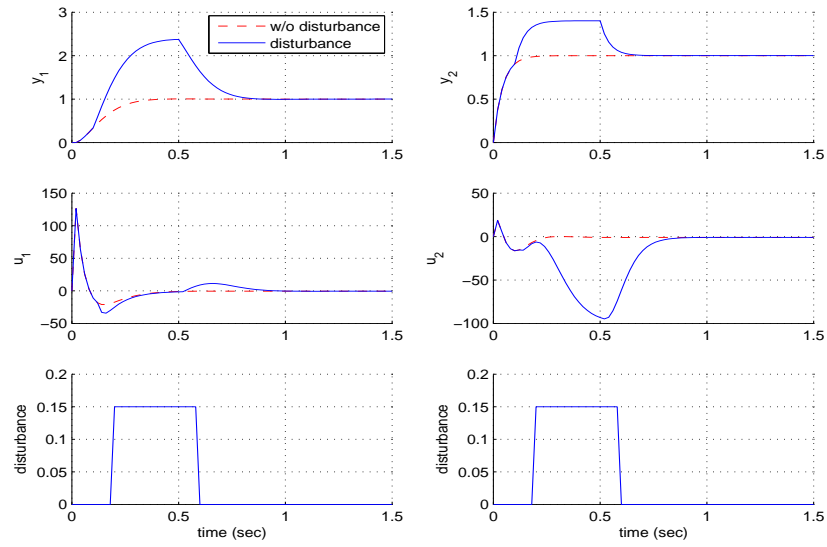


Figure 5.4: Constant disturbance at outputs at $p = 6$

Chapter 5. Nonlinear Model Predictive Control

Case 5: Disturbance Rejection: Improve the System Response

The purpose of this simulation is to show how to reduce the effect of the disturbance on the system response.

The parameters in this case are the same as in case 4, except the pole location parameters are reduced to $p_1 = p_2 = 15$. The simulation shows that the disturbances at the outputs are reduced significantly when changing the pole location parameters.

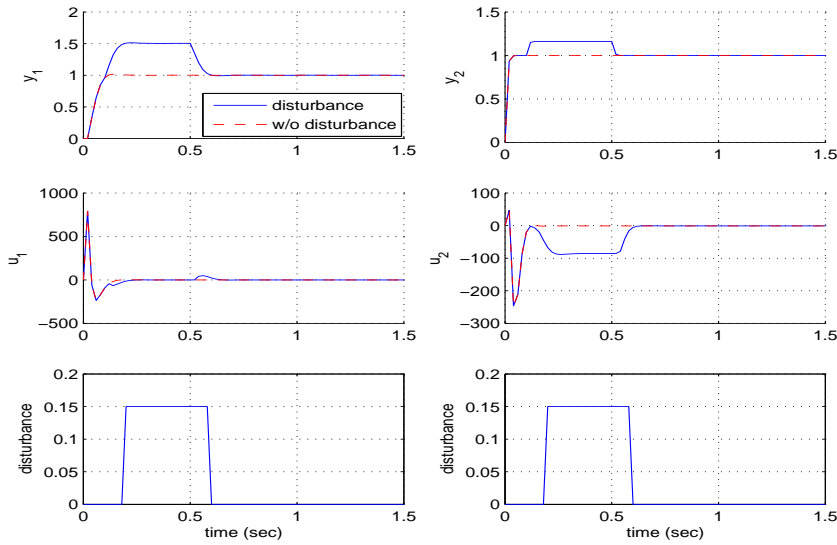


Figure 5.5: Constant disturbance at outputs at $p = 15$

5.3 Case Studies for NMPC Using Laguerre Functions

Example 2: Nonlinear Systems and Nonlinear Outputs

The nonlinear MPC with nonlinear outputs is developed in order to widen its applications. The nonlinear system is considered in the polar system which uses the radius and the angle as coordinate. The simple example is shown in (5.17).

$$\dot{x}_1 = x_1^2 + (1 + x_1^2)u_1 \quad (5.17)$$

$$\dot{x}_2 = x_2^2 + u_2$$

$$y_1 = \sqrt{x_1^2 + x_2^2}$$

$$y_2 = \tan^{-1}\left(\frac{x_1}{x_2}\right)$$

The new state space coordinates are:

$$z_1 = \sqrt{x_1^2 + x_2^2}$$

$$z_2 = \tan^{-1}\left(\frac{x_1}{x_2}\right)$$

The nonlinear MPC has the parameters p at 6 and N at 3. The imposed constraints are:

$$u_1 \leq 10$$

$$u_2 \leq 10$$

The simulation results of the example 2 are shown in figure 5.6.

Chapter 5. Nonlinear Model Predictive Control

Nonlinear System and Nonlinear Outputs

The purpose of this simulation is to investigate the ability of handling nonlinear constraints of the NMPC using Laguerre function. The outputs of this nonlinear system are also nonlinear. The tuning parameters are chosen as

$$T_{p1} = T_{p2} = 150 \quad N_1 = N_2 = 3, \quad p_1 = p_2 = 23$$

The imposed constraints are:

$$u_1 \leq 10$$

$$u_2 \leq 25$$

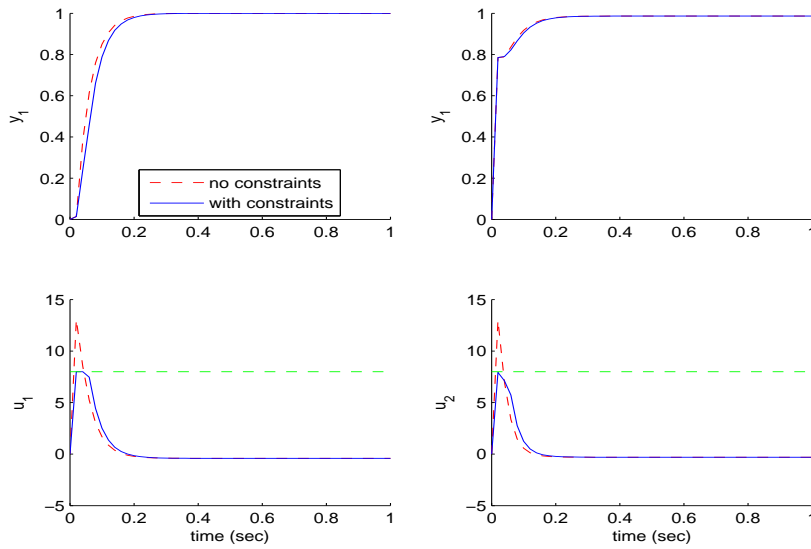


Figure 5.6: NMPC using Laguerre functions with constraints

5.3 Case Studies for NMPC Using Laguerre Functions

Example 3: Nonlinear System and Unknown Disturbance

This section gives the example of the disturbance rejection algorithm which mentioned in section 5.2.2. The Ball and Beam Systems [81] are considered. The nonlinear SISO system has an input, which is a derivative of the state x_4 , and an output, which is a state x_1 . The unknown disturbance $\theta(t)$ is depend on time and represented as a sinusoidal signal. The mathematical equations of the systems are rewritten as in the equation (5.18).

$$\dot{x}_1 = x_2 + \theta x_3^2 \quad (5.18)$$

$$\dot{x}_2 = x_2 - 2\theta x_3 x_4$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = u$$

$$y_1 = x_1$$

$$\theta = \sin(t)$$

The transformed states of the equation 5.18 is:

$$z_1 = x_1$$

$$z_2 = x_2 + \theta x_3^2$$

$$z_3 = x_3$$

$$z_4 = x_4$$

The NMPC is used to control the nonlinear system with the unknown disturbance. The constraint is imposed into the system.

$$-3 \leq u \leq 5$$

The simulation results are shown in figure 5.7.

Chapter 5. Nonlinear Model Predictive Control

Unknown Disturbance Rejection

The purpose of this simulation is to show the properties of the NMPC in handling the unknown disturbance. The unknown disturbance $\theta(t) = \sin(t)$ is added into the nonlinear system (as in equation 5.18). The system has an input $u = \dot{x}_4$ and an output $y = x_1$. The imposed constraints is:

$$-3 \leq u \leq 5$$

Conclusion: The figure shows the property of the MPC in disturbance rejection. The NMPC still produces feasible solutions under unknown disturbance.

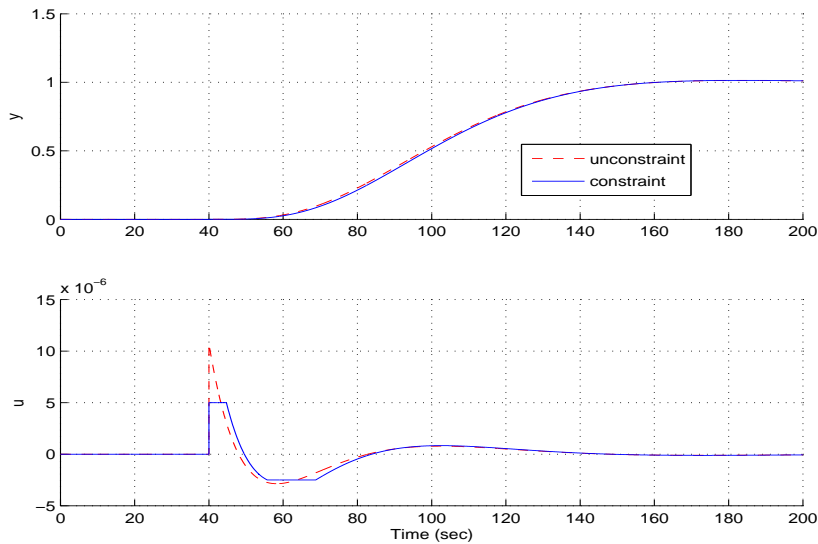


Figure 5.7: Constrained control for SISO under unknown constant disturbance θ

5.4 NMPC Control Using Taylor's Series

5.4 NMPC Control Using Taylor's Series

This section proposed the Nonlinear MPC using Taylor's series. Nonlinear system can be represented as follow:

$$\dot{x} = f(x) + g(x)u \quad (5.19)$$

$$y = h(x)$$

Taking derivative of the above equation for n^{th} times, we obtain:

$$x^{[2]} = \dot{f} + \dot{g}u + g\dot{u} = \dot{f} + C_0^1\dot{g} + C_1^1g\dot{u} \quad (5.20)$$

Deriving in the similar ways:

$$x^{[3]} = f^{[2]} + C_0^2g^{[2]}u + C_1^2\dot{g}u + C_2^2g\dot{u}^{[2]} \quad (5.21)$$

...

$$x^{[\rho]} = f^{[\rho-1]} + C_0^{\rho-1}g^{[\rho-1]}u + \dots + C_{\rho-1}^{\rho-1} \quad (5.22)$$

$$= f^{[\rho-1]} + \sum_{i=0}^{\rho-1} C_i^{\rho-1}g^{\rho-1-i}u^{[i]}$$

where: $C_n^m = \frac{m!}{n!(m-n)!}$

5.4.1 Prediction

At the current time say t_i , the state variable $x(t_i)$ is available. At the future time $t_i + \tau$, the predicted state variable $x(t_i + \tau)$ is derived as the following equation:

$$x(t_i + \tau) = x(t_i) + \tau\dot{x} + \frac{\tau^2}{2!}x^{[2]} + \dots + \frac{\tau^\rho}{\rho!}x^{[\rho]} \quad (5.23)$$

Substituting the equations (5.19) to (5.22) to equation 5.23, the following equation is obtained:

$$\begin{aligned} x(t_i + \tau) = & x(t_i) + [\tau f + \frac{\tau^2}{2!}\dot{f} + \dots + \frac{\tau^\rho}{\rho!}f^{[\rho-1]}] + \\ & + u[C_0^0g\tau + C_0^1\dot{g}\frac{\tau^2}{2!} + C_0^2g^{[2]}\frac{\tau^3}{3!} + \dots + C_0^{\rho-1}g^{[\rho-1]}\frac{\tau^\rho}{\rho!}] + \\ & + \dot{u}[C_1^1g\frac{\tau^2}{2!} + C_1^2\dot{g}\frac{\tau^3}{3!} + \dots + C_1^{\rho-1}g^{[\rho-2]}\frac{\tau^\rho}{\rho!}] + \end{aligned}$$

5.4 NMPC Control Using Taylor's Series

Unconstrained problem

$$u = - \sum_{i=1}^{N_p} ((CG(i\tau))^T Q(CG(i\tau)))^{-1} (CF(i\tau))^T (-CX(t) - CF(t, i\tau) - w) + \frac{1}{2} \eta^T R \eta \quad (5.28)$$

Constrained problem

The cost function can be rewritten as the following:

$$J = u^T \left(\sum_{i=1}^{N_p} (C\Gamma(i\tau)G(t))^T Q(C\Gamma(i\tau)G(t)) \right) u + \quad (5.29)$$

$$2u^T \left(\sum_{i=1}^{N_p} (C\Gamma(i\tau)G(t))^T Q(Cx(t) + C\Gamma(i\tau)F(t) - w(t)) \right)$$

subject to:

- inequality output constraints:

$$\begin{bmatrix} \sum_{j=1}^{N_p} \Gamma(j\tau)G(t_i) \\ -\sum_{j=1}^{N_p} \Gamma(j\tau)G(t_i) \end{bmatrix} U(t_i) \leq \begin{bmatrix} y_{high} - x(t_i) - \sum_{j=1}^{N_p} \Gamma(j\tau)F(t_i) \\ -y_{low} + x(t_i) + \sum_{j=1}^{N_p} \Gamma(j\tau)F(t_i) \end{bmatrix} \quad (5.30)$$

- inequality input constraints:

$$U_{min} \leq U(t_i) \leq U_{max} \quad (5.31)$$

Chapter 5. Nonlinear Model Predictive Control

5.5 Simulation Results for NMPC Using Taylor's Series

The stirred tank reactor is used as an example for the NMPC using Taylor's series. The modelling equation for this CSTR is given as the following:

$$\begin{aligned}\dot{x}_1 &= u_1 + u_2 - 0.2x_1^{0.5} \\ \dot{x}_2 &= \frac{24.9 - x_2}{x_1}u_1 + \frac{0.1 - x_2}{x_1}u_2 - \frac{x_2^2}{x_2 + 1} \\ y_1 &= x_1 \\ y_2 &= x_2\end{aligned}\tag{5.32}$$

The process is simulated for 20 seconds and the sampling time τ is 0.02 seconds. The initial conditions for the state variables is $[x_1, x_2]^T = [0.4, 0.4]^T$. The Taylor's series parameter is chosen as $N_p = 6$.

The constraints are imposed to the system at the inputs. They are presented as the following:

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \leq \begin{bmatrix} 0.2 \\ 5 \end{bmatrix}\tag{5.33}$$

The simulation result is shown in figure 5.8. The constraints are also imposed into the outputs of the system. The imposed constraints are:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix}\tag{5.34}$$

The simulation result is shown in figure 5.9.

5.5 Simulation Results for NMPC Using Taylor's Series

This simulation shows the NMPC using Taylor's series to control the stirred tank reactor. The constraints are imposed into the inputs of the system. The constraints are:

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \leq \begin{bmatrix} 0.2 \\ 5 \end{bmatrix} \quad (5.35)$$

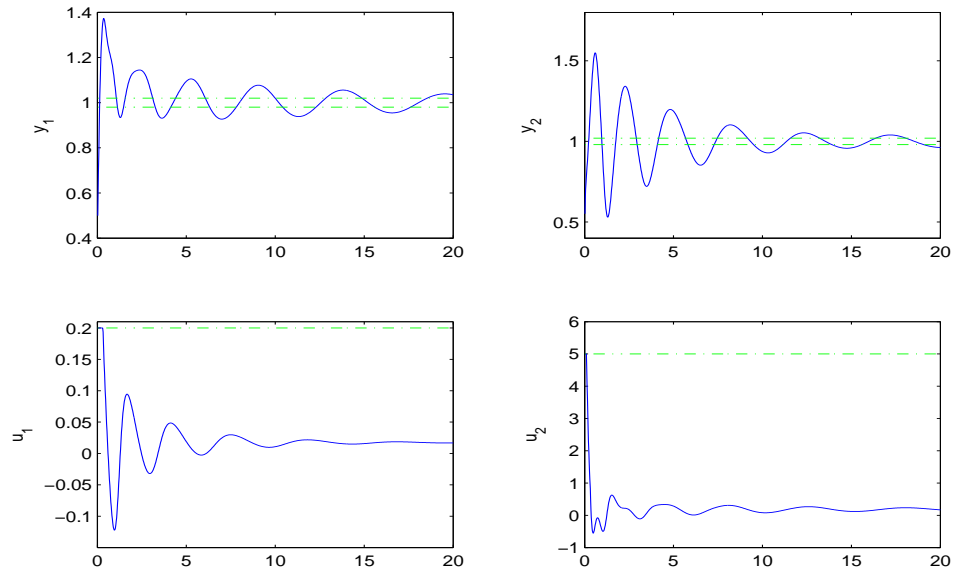


Figure 5.8: NMPC using Taylor expansion: Input constrained problem.

Chapter 5. Nonlinear Model Predictive Control

This simulation shows the NMPC using Taylor's series to control the stirred tank reactor. The constraints are imposed into the outputs of the system. The constraints are:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (5.36)$$

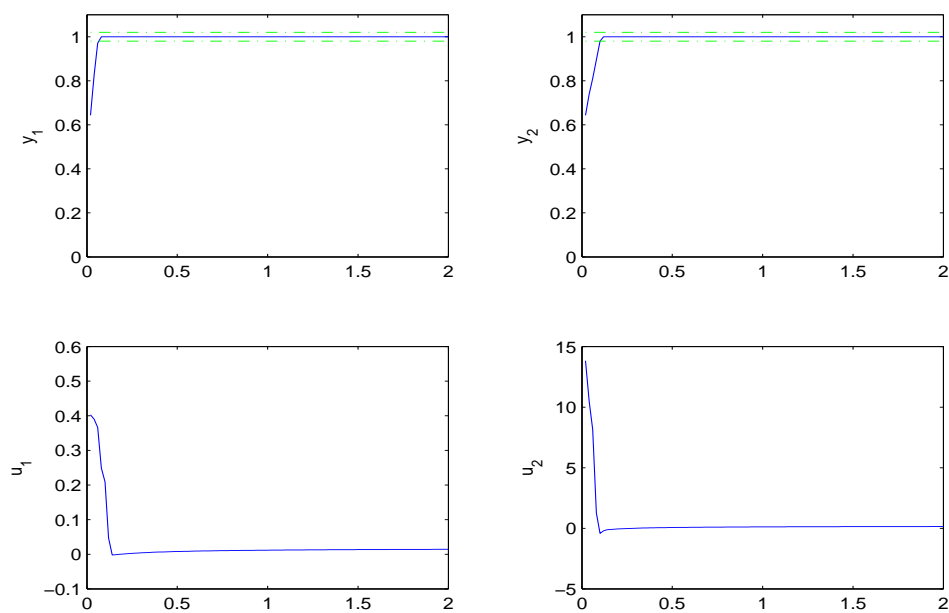


Figure 5.9: NMPC using Taylor expansion: Output constrained problem.

5.6 Conclusion

The NMPC designs using Laguerre and Lie's functions inherit the advantages of the MPC design. They are very robust and produce the reliable outputs. It is simple to obtain the velocity form for the NMPC by using the Lie's function. From that, the design is able to optimise the process plant by using the Laguerre function as in the MPC design. The constraints in this design are nonlinear and inequality. The constraints are imposed in the control inputs. One merit of the design is able to handle the nonlinear system with unknown parameters or disturbances.

The NMPC design using Taylor's series function is easy to understand due to the well known of the Taylor's series. The output response is unstable and slow reach to the setpoint. However, it is very efficient under the constrained control problem. It can handle the constraints which imposed at the outputs.

Chapter 5. Nonlinear Model Predictive Control

Chapter 6

Autonomous Underwater Vehicles

6.1 Introduction

The purpose of this chapter is to improve the AUV motion so that it can be applied to the IMPC and the NMPC algorithms. Sections 6.1 to 6.5 revisit the six degree of freedom formula which proposed by the MIT University in 1969. The ocean effects, such as buoyancy force, gravity force, hydrostatic force, lift force, drag force and also their moments, cause the AUV model to be highly nonlinear. The general nonlinear models of the AUV are shown in sections 6.2 to 6.5 and state space model is shown in section 6.5.3.

There are several ways to control the AUV. Those methods are described in [1, 3, 54]. In section 6.6.1, the IMPC approach is used to control the tuning direction and diving direction of the AUV based on the rotational angle of the rudder and stern. The performance of the IMPC is once again confirmed. The contribution of this work is proposed in [75].

The NMPC algorithm is also applied to the control the nonlinear model of the AUV. The simulation results of the IMPC and NMPC approaches are presented in section 6.6.2. The conclusions are stated in section 6.7.

Chapter 6. Autonomous Underwater Vehicles

6.2 Mathematical Modelling of AUVs

In this section, the equations of 6 degree of freedom of AUVs are presented. Those equations are based on the assumption:

- The AUVs are considered like a rigid body.
- The earth rotation is neglected.
- The forces acting on vehicles are hydrostatic, lift, drag, control forces.

6.2.1 Vehicle Dynamics

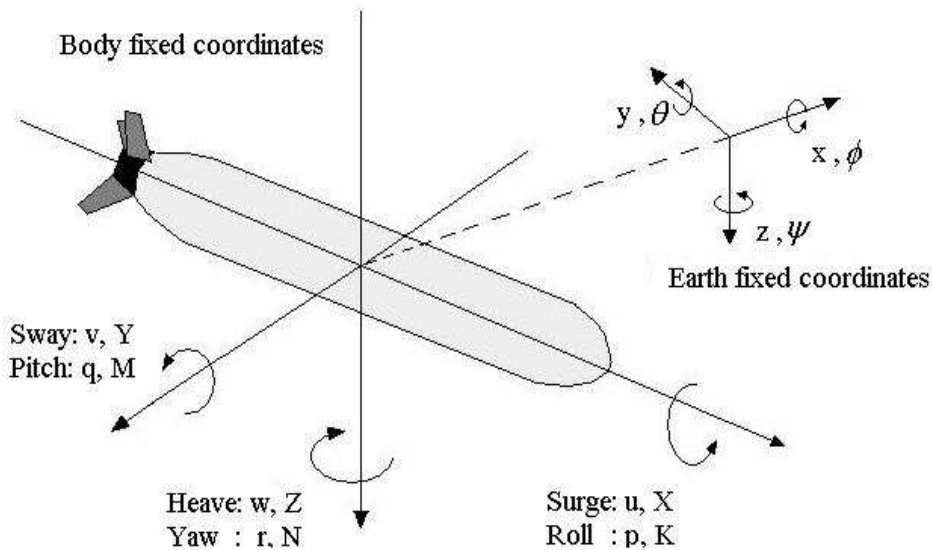


Figure 6.1: Body fixed coordinates and earth fixed coordinates

6 degree of freedom vectors:

- $[u, v, w]^T$: the translations of the vehicle with respect to the body fixed coordinates. u is a surge speed. v is sway speed, and w is heave speed.
- $[x, y, z]^T$: the positions of the vehicle with respect to the earth fixed coordinates. x is positive forward. y is perpendicular to starboard. z is perpendicular to the water line.

6.2 Mathematical Modelling of AUVs

- $[\phi, \theta, \psi]$: the orientations of the vehicle with respect to the earth fixed coordinates. The roll angle ϕ , the pitch angle θ , and the yaw angle ψ are positive rotation about x, y, and z axis, respectively.
- $[p, q, r]$: the rotation velocities of the vehicle with respect to the body fixed coordinates.
- $[X, Y, Z]$: the total forces acting on the vehicle with respect to the body fixed coordinates.
- $[K, M, N]$: the moments acting on vehicles with respect to the body fixed coordinates.

$[x_g, y_g, z_g]^T$ is a center of gravity and is defined by vector \vec{r}_G . $[x_b, y_b, z_b]^T$ is a center of buoyancy of the vehicle and is defined by vector \vec{r}_B . The vehicle is now treated as a rigid body. Let \vec{a}_G be an acceleration at center of gravity. \vec{a}_G is defined as:

$$\vec{a}_G = \frac{\delta \vec{V}}{\delta t} + \vec{\omega} \vec{V} + \dot{\vec{\omega}} \vec{r}_G + \vec{\omega} \vec{\omega} \vec{r}_G$$

From the Newton's equation of motion, the sum of forces is defined as: $\sum F = m\vec{a}_G$. Substituting \vec{a}_G into equation of motion, the force equations in the x, y, z direction will be determined.

With known moment of inertia in I_x , I_y , and I_z , the matrix of moment of inertia $[I] = \text{diagonal}(I_x, I_y, I_z)$. The rate of change of angular momentum about center of gravity is: $\dot{H}_G = [I]\dot{\vec{\omega}} + \vec{\omega} I \vec{\omega}$. From the Euler's equation of motion, the total moments of the rigid body with 6 degree of freedom is defined as: $\sum M = \dot{H}_G + \vec{r}_G m \vec{a}_G$. Substituting \vec{a}_G again and drop the higher term (e.g. x_g^2, y_g^2, z_g^2), the moment equations in x, y, and z directions are determined.

The equations of the motion for the vehicle in 6 degree of freedom are defined as the following [58]

$$\begin{aligned} m[\dot{u} - vr - wq - x_g(q^2 + r^2) + y_g(pq - \dot{r}) + z_g(pr + \dot{q})] &= \sum X_{ext} \\ m[\dot{v} - wp + ur - y_g(r^2 + p^2) + z_g(qr - \dot{p}) + x_g(qp + \dot{r})] &= \sum Y_{ext} \\ m[\dot{w} - uq + vp - z_g(p^2 + q^2) + x_g(rp - \dot{q}) + y_g(rq + \dot{p})] &= \sum Z_{ext} \\ I_x \dot{p} + (I_z - I_y)qr + m[y_g(\dot{w} - uq + vp) - z_g(\dot{v} - wp + ur)] &= \sum K_{ext} \quad (6.1) \\ I_y \dot{q} + (I_x - I_z)rp + m[z_g(\dot{u} - vr + wq) - x_g(\dot{w} - uq + vp)] &= \sum M_{ext} \\ I_z \dot{r} + (I_y - I_x)pq + m[x_g(\dot{v} - wp + ur) - y_g(\dot{u} - vr + wq)] &= \sum N_{ext} \end{aligned}$$

Chapter 6. Autonomous Underwater Vehicles

The external forces acting on the vehicle are:

$$\sum F_{ext} = F_{hydrostatic} + F_{drag} + F_{lift} + F_{control} \quad (6.2)$$

6.3 Vehicle Kinematic

The relationship between the earth fixed coordinates and body fixed coordinates can be described from the translation velocities as well as the rotational velocities [94]

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} c(\psi)c(\theta) & -s(\psi)c(\phi) + c(\psi)s(\theta)s(\phi) & s(\psi)s(\phi) + c(\psi)s(\theta)c(\phi) \\ s(\psi)c(\theta) & c(\psi)c(\phi) + s(\psi)s(\theta)s(\phi) & -c(\psi)s(\phi) + s(\phi)s(\theta)c(\phi) \\ -s(\theta) & c(\theta)s(\phi) & c(\theta)c(\phi) \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (6.3)$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s(\phi)t(\theta) & c(\phi)t(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)/c(\theta) & c(\phi)/c(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (6.4)$$

where: $c = \cos$, $s = \sin$, and $t = \tan$ in the above equations.

6.4 Forces and Moments Acting on AUVs

In this chapter, the forces and moments acting on the AUVs (6.2) are analysed. The coefficients related are also carried out.

6.4.1 Hydrostatic Forces

The rigid body moving in the fluid is affected by:

- W: vehicle weight force that acting at the center of gravity: $W = mg$.
- B: buoyancy force that acting at the center of buoyancy: $B = \rho Vg$.

Both weight and buoyancy force vectors do not change with vehicle attitude. They are expressed as:

$$f_w = 0\hat{i} + 0\hat{j} + W\hat{k}$$

6.4 Forces and Moments Acting on AUVs

$$f_b = 0\hat{i} + 0\hat{j} + B\hat{k}$$

The moment vectors are described as:

$$M = \vec{r}_G f_w - \vec{r}_B f_b$$

The forces are acting on the vertical axis. Back to equation (6.4), the 3rd column affects to those forces. So, the hydrostatic forces and moments are:

$$\begin{aligned} X_{HS} &= -(W - B)\sin(\theta) \\ Y_{HS} &= (W - B)\cos(\theta)\sin(\phi) \\ Z_{HS} &= (W - B)\cos(\theta)\cos(\phi) \\ K_{HS} &= -(y_g W - y_b B)\cos(\theta)\cos(\phi) - (z_g W - z_b B)\cos(\theta)\sin(\phi) \\ M_{HS} &= -(z_g W - z_b B)\sin(\theta) - (x_g W - x_b B)\cos(\theta)\cos(\phi) \\ N_{HS} &= -(x_g W - x_b B)\cos(\theta)\sin(\phi) - (y_g W - y_b B)\sin(\theta) \end{aligned} \quad (6.5)$$

6.4.2 Lift Forces

Body Lift Forces

When the AUV swims with an angle of attack, the body lift forces occurs. The pressure at the aft of the AUV is dropped. That pressure can be described as the force located at the center of pressure.

$$F_{body} = -\frac{1}{2}\rho d^2 C_{yd\beta} u w \quad (6.6)$$

where: ρ : fluid density, d : maximum hull diameter, $C_{yd\beta}$: lift slope coefficient.

Fin Lift Forces

The rudder and stern of the AUV are acted on by the lift force when held at an angle of attack. Equations of lift forces and moments:

$$\begin{aligned} Y_r &= \frac{1}{2}\rho C_{L\alpha} S_{fin} [u^2 \delta r - uv - x_{fin}(ur)] \\ Z_s &= -\frac{1}{2}\rho C_{L\alpha} S_{fin} [u^2 \delta s + uw - x_{fin}(uq)] \\ M_s &= \frac{1}{2}\rho C_{L\alpha} S_{fin} x_{fin} [u^2 \delta s + uw - x_{fin}(uq)] \\ N_r &= \frac{1}{2}\rho C_{L\alpha} S_{fin} x_{fin} [u^2 \delta r - uv - x_{fin}(ur)] \end{aligned} \quad (6.7)$$

where: S_{fin} : fin platform area, x_{fin} : position of fin, $C_{L\alpha}$: fin lift slope.

Chapter 6. Autonomous Underwater Vehicles

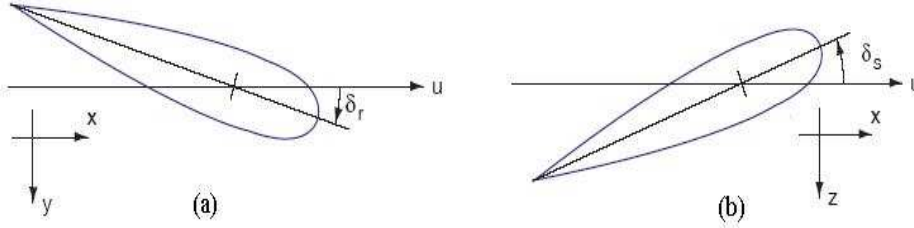


Figure 6.2: (a) Rudder angle attack and (b) Stern angle attack

6.4.3 Drag Forces

It is the skin friction due to the boundary layers. Like other friction force, it opposes the movement of the vehicle. The drag force is expressed as:

$$X = -\left(\frac{1}{2}\rho C_d A_f\right)u|u| \quad (6.8)$$

where: C_d : drag coefficient.

6.4.4 Added Mass

It is a measure of additional inertia created by water when the vehicle accelerates [68]. The forces and moments are:

$$\begin{aligned} X_A &= X_{\dot{u}}\dot{u} + Z_{\dot{w}}w\dot{q} + Z_{\dot{q}}\dot{q}^2 - Y_{\dot{v}}v\dot{r} - Y_{\dot{r}}\dot{r}^2 \\ Y_A &= Y_{\dot{v}}\dot{v} + Y_{\dot{r}}\dot{r} + X_{\dot{u}}u\dot{r} - Z_{\dot{w}}w\dot{p} - Z_{\dot{q}}\dot{p}\dot{q} \\ Z_A &= Z_{\dot{w}}\dot{w} + Z_{\dot{q}}\dot{q} - X_{\dot{u}}u\dot{q} + Y_{\dot{v}}v\dot{p} + Y_{\dot{r}}\dot{r}\dot{p} \\ K_A &= K_{\dot{p}}\dot{p} \\ M_A &= M_{\dot{w}}\dot{w} + M_{\dot{q}}\dot{q} - (Z_{\dot{w}} - X_{\dot{u}})uw - Y_{\dot{r}}v\dot{p} + (K_{\dot{p}} - N_{\dot{r}})r\dot{p} \\ N_A &= N_{\dot{v}}\dot{v} + N_{\dot{r}}\dot{r} - (X_{\dot{u}} - Y_{\dot{v}})uv + Z_{\dot{q}}w\dot{p} - (K_{\dot{p}} - M_{\dot{q}})\dot{p}\dot{q} + Y_{\dot{r}}u\dot{r} \end{aligned} \quad (6.9)$$

6.4.5 Euler's Method

If Δt is the sample of time, the Euler' method is described as the following:

$$x(t + 1) = x(t) + \dot{x}\Delta t \quad (6.10)$$

6.5 Modelling Methodology

In this thesis, the diving and turning of the AUV are considered. Diving is manoeuver in the x-z plane of the vehicle. The motion involved are surge, heave and pitch. At the end of the dive, the depth of the vehicle should get to the constant. In our simulation, the constant depths of the AUV are 10m and 20m. In turning, it will be accomplished in x-y direction.

The trajectory of the AUV is controlled by the rudder and stern. When the rudder is rotated, the AUV will turn right or left. The rotation of rudder influents to the movement in the y direction of AUV. Similarly, the rotation of the stern influent to the movement in the z direction (See figure 6.2).

The question is how much can we steer the rudder or the stern so that the AUV will get to the position y or the depth z in which we want.

The y and z directions are interested. So, we concentrate on the position y, depth z, pitch rate q, yaw rate r, pitch angle θ , and yaw angle ψ . Terms (x, p) are set zero. The surge velocity u, sway velocity v and heave velocity w are set constant. The surge velocity u is very important. The roll angle is also constant.

We now define the equations for the diving and turning model.

6.5.1 Equations of motion

From equations 6.1, the equations of motion for pitch and yaw are:

Pitch:

$$I_y \dot{q} - m z_g V r + m z_g w q + m x_g U q = \sum M_{ext}$$

Yaw:

$$I_z \dot{r} + m x_g U r + m y_g V r + m y_g w q = \sum N_{ext} \quad (6.11)$$

From equation (6.4) and equation (6.3), we have

$$\begin{aligned} \dot{\theta} &= q - \phi r \\ \dot{\psi} &= \phi q + r \\ \dot{y} &= U \psi \\ \dot{z} &= -U \theta \end{aligned} \quad (6.12)$$

Chapter 6. Autonomous Underwater Vehicles

6.5.2 External forces and moments acting on AUV

1. Hydrostatic moments:

From equations of hydrostatic forces and moments (equation 6.5), the moments acting on pitch and yaw are:

$$\sum M_{HS} = -(z_g W - z_b B)\theta$$

$$\sum N_{HS} = -(y_g W - y_b B)\theta$$

2. Lift moments:

From equation (6.7), the fin lift moments can be defined. The coefficients and values of fin lift moments are shown in Appendix (A.2.1 and A.2.2)

$$M_s = M_{uu\delta s} U^2 \delta s + M_{uqf} u q$$

$$N_r = N_{uu\delta r} U^2 \delta r + N_{urf} u r$$

3. Drag moments:

The drag moment acting on y direction is (Appendix A.2.1):

$$M_{qc} = -\frac{1}{2} \rho C_{dc} m_q \int_{x_t}^{x_{b2}} 2x^3 R(x) dx - 2x_{fin}^3 \left(\frac{1}{2} \rho S_{fin} C_{dc} m_q \right)$$

4. Added mass:

$$M_A = M_{\dot{q}} \dot{q} - Z_{\dot{q}} U q$$

$$N_A = N_{\dot{r}} \dot{r} - Y_{\dot{r}} U r$$

5. Total moments acting on AUV:

From the equations of hydrostatic, lift, drag, added mass moments, and equation (6.2), the sum of moments acting on pitch and yaw are defined as:

$$\sum M = M_{\dot{q}} \dot{q} + M_q q + M_{uu\delta s} U^2 \delta s + M_{\theta} \theta$$

$$\sum N_A = N_{\dot{r}} \dot{r} + N_r r + N_{uu\delta r} U^2 \delta r + N_{\theta} \theta \quad (6.13)$$

The equations above are reduced. The coefficients such as $M_{\dot{q}}$, M_q , $N_{\dot{r}}$, N_r , etc. are sum of other coefficients. Their origin equations and values are shown in Appendix (A.2.1 and A.2.2).

6.5 Modelling Methodology

6.5.3 State space equations

Combined all equations together (equations 6.11, 6.12 and 6.13), the equations of motion are:

$$\begin{aligned}
 (I_y - M_{\dot{q}})\dot{q} &= mz_g V r + (M_q - mz_g w - mx_g U)q + M_\theta \theta + M_{uu\delta s} U^2 \delta s \\
 (I_z - N_{\dot{r}})\dot{r} &= (N_r - mx_g U - my_g V)r - my_g w q + N_\theta \theta + N_{uu\delta r} U^2 \delta r \\
 \dot{\theta} &= q - \phi r \\
 \dot{\psi} &= \phi q + r \\
 \dot{y} &= U \psi \\
 \dot{z} &= -U \theta
 \end{aligned}$$

Now, we let the state vector $x = [q, r, y, z, \psi, \theta]^T$, and the input vector $u = [\delta r, \delta s]^T$. We can form the state space model:

$$\begin{bmatrix} (I_y - M_{\dot{q}}) & 0 & 0 & 0 & 0 & 0 \\ 0 & (I_z - N_{\dot{r}}) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{q} \\ \dot{r} \\ \dot{y} \\ \dot{z} \\ \dot{\psi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & M_{uu\delta s} U^2 \\ N_{uu\delta r} U^2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta r \\ \delta s \end{bmatrix} + \quad (6.14)$$

$$\begin{bmatrix} (M_q + mx_q U - mz_g w) & mz_g V & 0 & 0 & 0 & M_\theta \\ -mx_g w & (N_r - mx_g U - my_g V) & 0 & 0 & 0 & N_\theta \\ 0 & 0 & 0 & 0 & U & 0 \\ 0 & 0 & 0 & 0 & 0 & -U \\ \phi & 1 & 0 & 0 & 0 & 0 \\ 1 & \phi & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q \\ r \\ y \\ z \\ \psi \\ \theta \end{bmatrix} \quad (6.15)$$

We rewrite the equation 6.15 as a form:

$$M\dot{x} = Nx + Ku$$

then:

$$\dot{x} = Ax + Bu$$

where: $A = M^{-1}N$ and $B = M^{-1}K$.

With the state space model and all values of coefficients, we are able to model the dive and the turn of the AUV.

6.6 Case Studies

6.6.1 Intermittent Continuous-time Model Predictive Control

The influences of the rudder and stern angles on the AUV's trajectory are considered and depicted in figure (6.3). Under the unconstrained and constrained problems, the tuning parameters and the disturbances are also investigated. The reference trajectories for turn y and dive z directions are chosen as 1. The tuning parameters p , N , and T_p are set $2/15$, 3 and 100 , respectively. The performances of the model under unconstrained problems are depicted in figure (6.8). In this case, the tuning parameters are studied. While increasing parameters p from $1/15$, $2/15$ to $3/15$ or parameters N from 2 , 3 to 4 , the speed of the system responses faster. The speed is inverse proportion with increasing T_p . The influences of the tuning parameters p , N , and T_p are shown in figure (6.4).

After an optimization process, the information of the control signal, which generated by the IMPC, can be predicted at each sampling interval h (or Δt). The vary of h does not affect the output responses. However, the increase of h slowly reduces the amplitude of the control input signals. The figure (6.5) shows the simulation results while values of h are chosen as 0.01 , 0.02 and 0.04 seconds.

To investigate how disturbances affect the performance of the system, unit step inputs were added to the model at time $t = 5$ seconds. With the tuning parameters $p = 1/30$, it can be seen that the disturbances affect both input and output responses. These effects are shown in figure (6.6). To reduce these effects, the gain of observers is also reduced. The new poles p now are $5/30$. The performances of the system are improved and are shown in figure (6.7).

While keeping the same tuning parameters as aforementioned, the constrained controls are applied to the model. The sampling time τ is 0.002 seconds and $\gamma = 0.001$. The set of maximum and minimum limits of $u(\tau_i)$ and $\dot{u}(\tau_i)$ are $[\dot{u}_{1min}, \dot{u}_{2min}, u_{1min}, u_{2min}, \dot{u}_{1max}, \dot{u}_{2max}, u_{1max}, u_{2max}] = [-3 * 10^{-4}, -6 * 10^{-4}, -10^{-3}, -5 * 10^{-5}, 10^{-4}, 3 * 10^{-3}, 4 * 10^{-4}, 3 * 10^{-4}]$. The imposed constraints, where $i = 0, 1, 2$, are:

$$u_{1min} + i\gamma u_{1min} \leq u_1(i\tau) \leq u_{1max} - i\gamma u_{1max}$$

$$u_{2min} + i\gamma u_{2min} \leq u_2(i\tau) \leq u_{2max} - i\gamma u_{2max}$$

$$\dot{u}_{1min} + i\gamma \dot{u}_{1min} \leq \dot{u}_1(i\tau) \leq \dot{u}_{1max} - i\gamma \dot{u}_{1min}$$

$$\dot{u}_{2min} + i\gamma \dot{u}_{2min} \leq \dot{u}_2(i\tau) \leq \dot{u}_{2max} - i\gamma \dot{u}_{2min}$$

6.6 Case Studies

The cost function of the model under constrained problems is solved by active set quadratic programming procedure. There are 24 constraints in the model. The simulation results are shown in figure (6.8).

Chapter 6. Autonomous Underwater Vehicles

Case 1: Effect of the inputs on the system

This simulation is to investigate how the stern and rudder angles affect the outputs. The unit step inputs are used to control the AUV. The figure shows the simulation outputs.

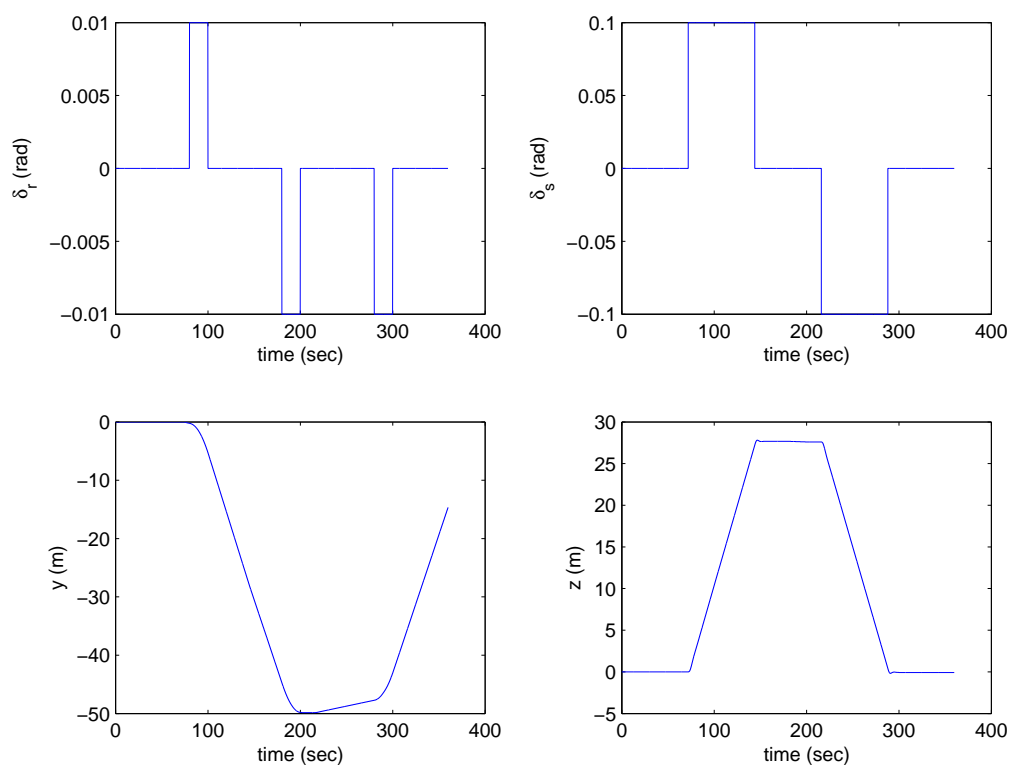


Figure 6.3: Influences of rudder and stern angles on the AUV's trajectory

6.6 Case Studies

Case 2: Effects of T_p , N , and p parameters on the system

The purpose of this simulation is to investigate the effects of the tuning parameters on to the system.

Case 1: Parameters T_p and N are constant. $T_{p1} = T_{p2} = 100$ and $N_1 = N_2 = 3$. The parameters $p_1 = p_2$ vary from $1/15$, $2/15$ to $3/15$.

Case 2: Parameters T_p and p are constant. $T_{p1} = T_{p2} = 100$ and $p_1 = p_2 = 2/15$. The parameters $N_1 = N_2$ vary from 2, 3 to 4.

Case 3: Parameters N and p are constant. $N_1 = N_2 = 3$ and $p_1 = p_2 = 2/15$. The parameters $T_{p1} = T_{p2}$ vary from 50, 100 to 150.

In the figure, the first row shows the outputs of the case 1. The second row shows the outputs of the case 2. The third row shows the outputs of the case 3.

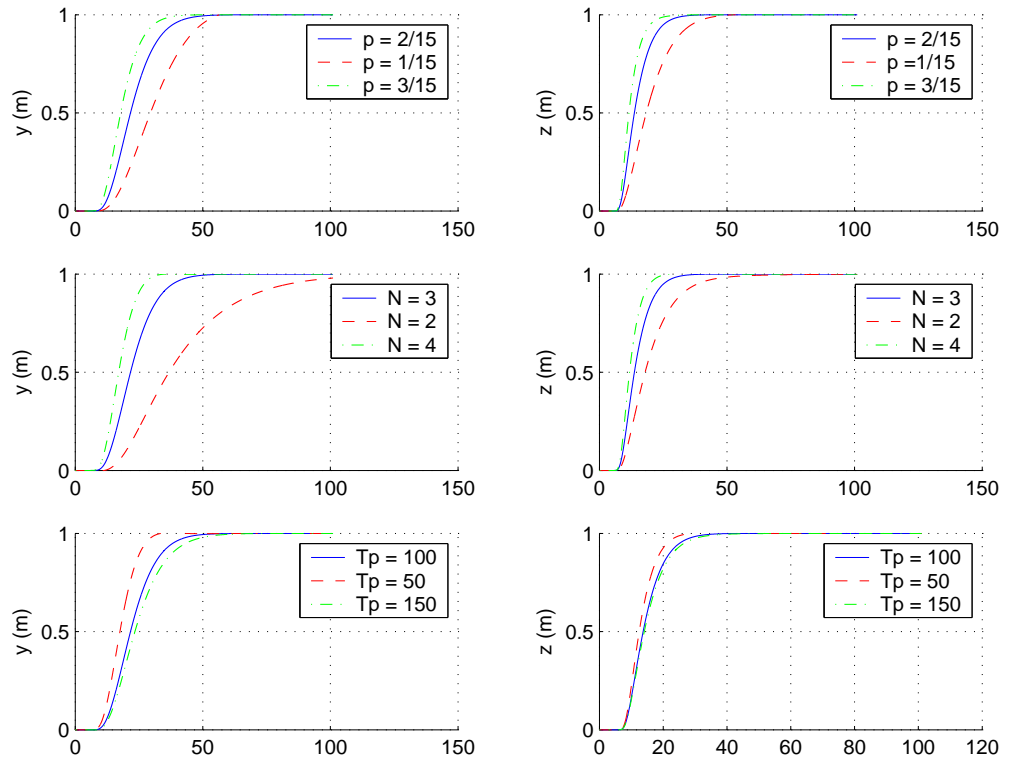


Figure 6.4: Continuous time IMPC: Changing parameters N, T_p, p to tune the response speeds.

Chapter 6. Autonomous Underwater Vehicles

Case 3: Effects of the sampling time interval on the system

The purpose of this simulation is to investigate the effect of the sampling time interval on the system response. The tuning parameters T_p , p , and N are kept constant.

$$T_{p1} = T_{p2} = 100, \quad p_1 = p_2 = 2/15, \quad N_1 = N_2 = 3$$

The sampling time intervals (h) are chosen as follow:

Case2 : $h = 0.01\text{sec}$

Case3 : $h = 0.02\text{sec}$

Case4 : $h = 0.4\text{sec}$

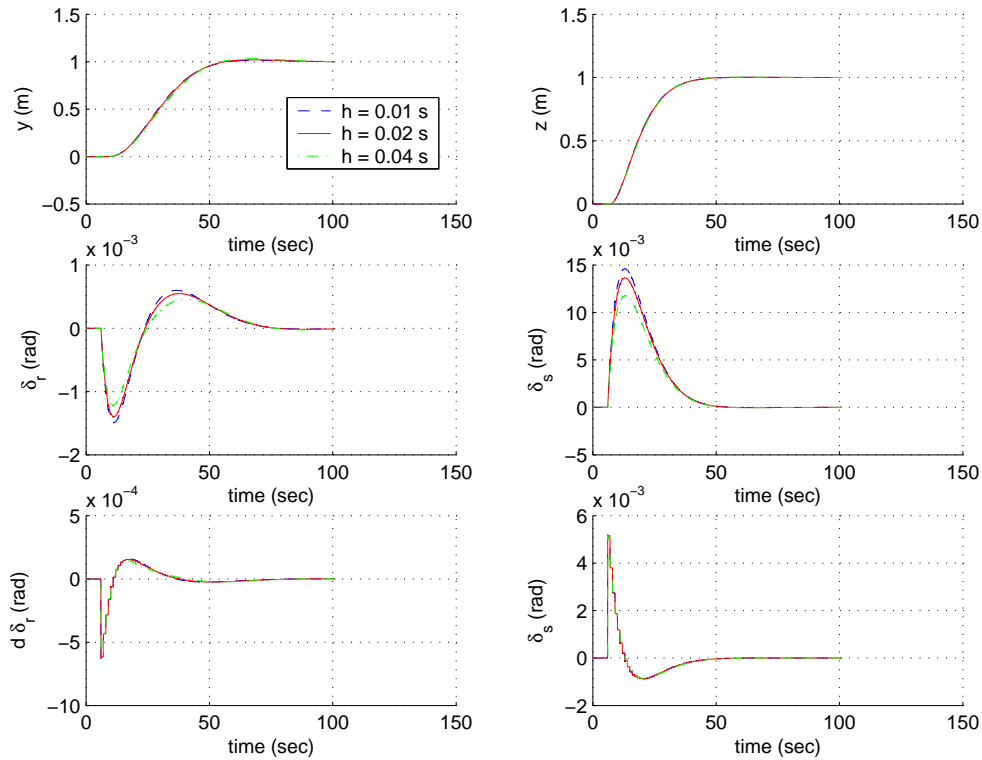


Figure 6.5: Continuous time IMPC: Changing sampling interval h .

6.6 Case Studies

Case 4: Disturbance Rejection

The purpose of the simulation is to investigate how the IMPC approach handles the disturbance of the system at time $t = 5$ second.

The disturbances are added to the inputs of the system. The tuning parameters are $T_{p1} = T_{p2} = 100$, $N_1 = N_2 = 3$, and $p_1 = p_2 = 1/30$.

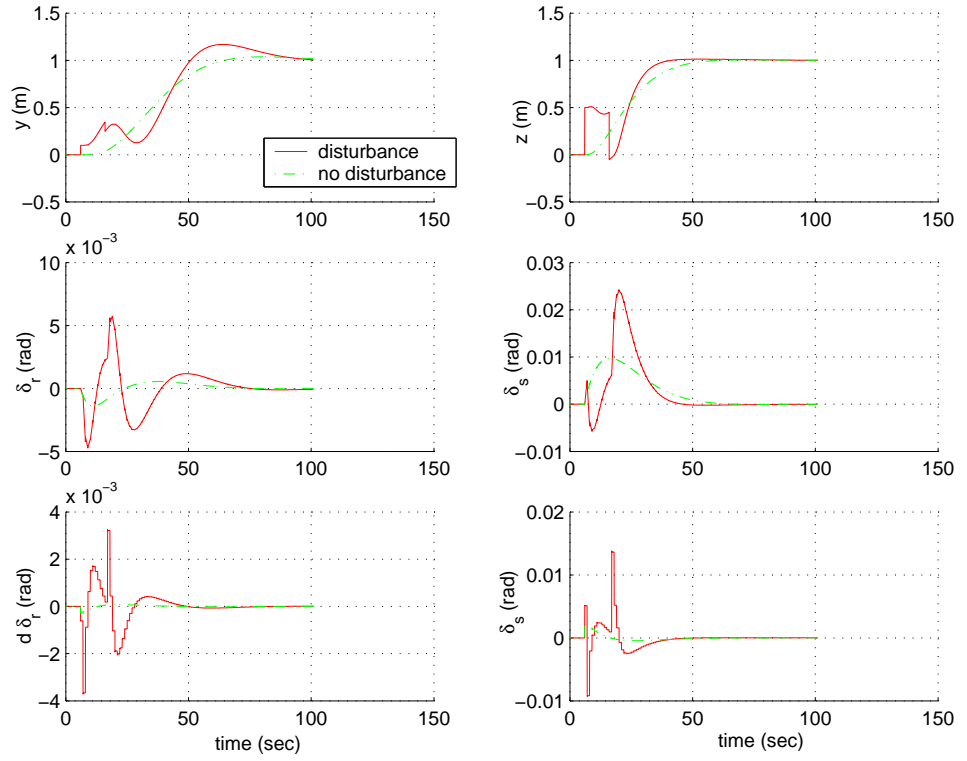


Figure 6.6: Continuous time IMPC: Investigating disturbances with higher observer gain.

Chapter 6. Autonomous Underwater Vehicles

Case 5: Reducing the Effect of the Disturbance

The purpose of this simulation is to show how to reduce the effect of the disturbance on the system response.

The parameters in this case are the same as in case 4, except the pole location parameters are reduced to $p_1 = p_2 = 5/30$. The simulation shows that the disturbance is reduced when changing the pole location parameters.

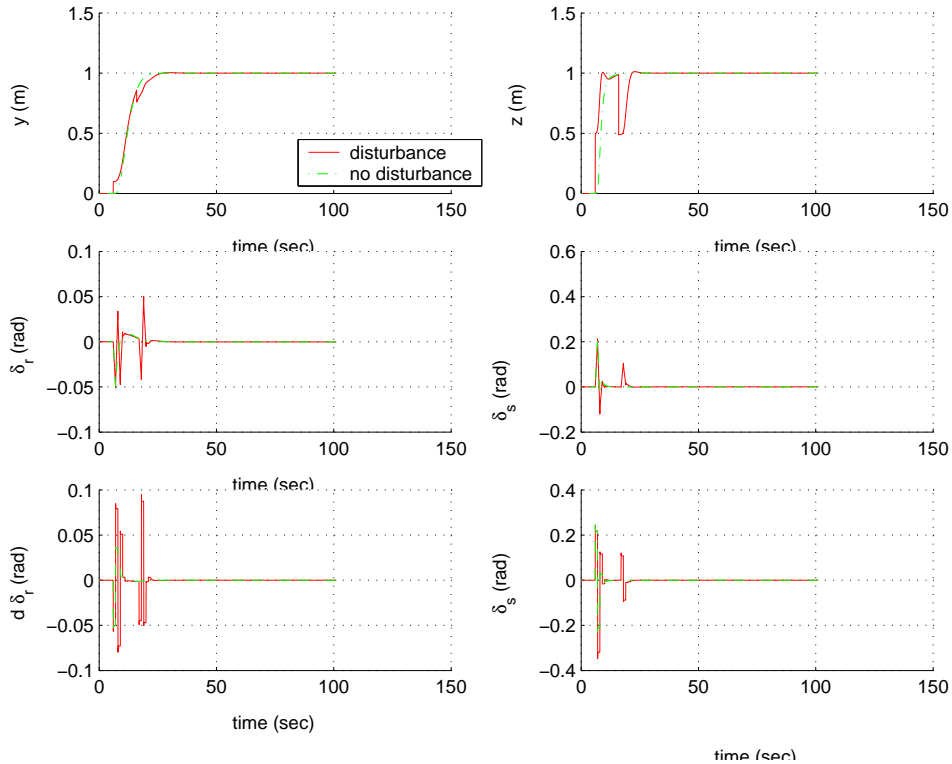


Figure 6.7: Continuous time IMPC: Investigating disturbances with lower observer gain.

Case 6: Constrained Control

The simulation shows how the IMPC handles the constrained problems. The tuning parameters are kept the same as in case 4. The sampling time τ is 0.002 seconds and $\gamma = 0.001$. The set of maximum and minimum limits of $u(\tau_i)$ and $\dot{u}(\tau_i)$ are $[\dot{u}_{1min}, \dot{u}_{2min}, u_{1min}, u_{2min}, \dot{u}_{1max}, \dot{u}_{2max}, u_{1max}, u_{2max}] = [-3*10^{-4}, -6*10^{-4}, -10^{-3}, -5*10^{-5}, 10^{-4}, 3*10^{-3}, 4*10^{-4}, 3*10^{-4}]$. The imposed constraints, where $i = 0, 1, 2$, are:

$$u_{1min} + i\gamma u_{1min} \leq u_1(i\tau) \leq u_{1max} - i\gamma u_{1max}$$

$$u_{2min} + i\gamma u_{2min} \leq u_2(i\tau) \leq u_{2max} - i\gamma u_{2max}$$

$$\dot{u}_{1min} + i\gamma \dot{u}_{1min} \leq \dot{u}_1(i\tau) \leq \dot{u}_{1max} - i\gamma \dot{u}_{1min}$$

$$\dot{u}_{2min} + i\gamma \dot{u}_{2min} \leq \dot{u}_2(i\tau) \leq \dot{u}_{2max} - i\gamma \dot{u}_{2min}$$

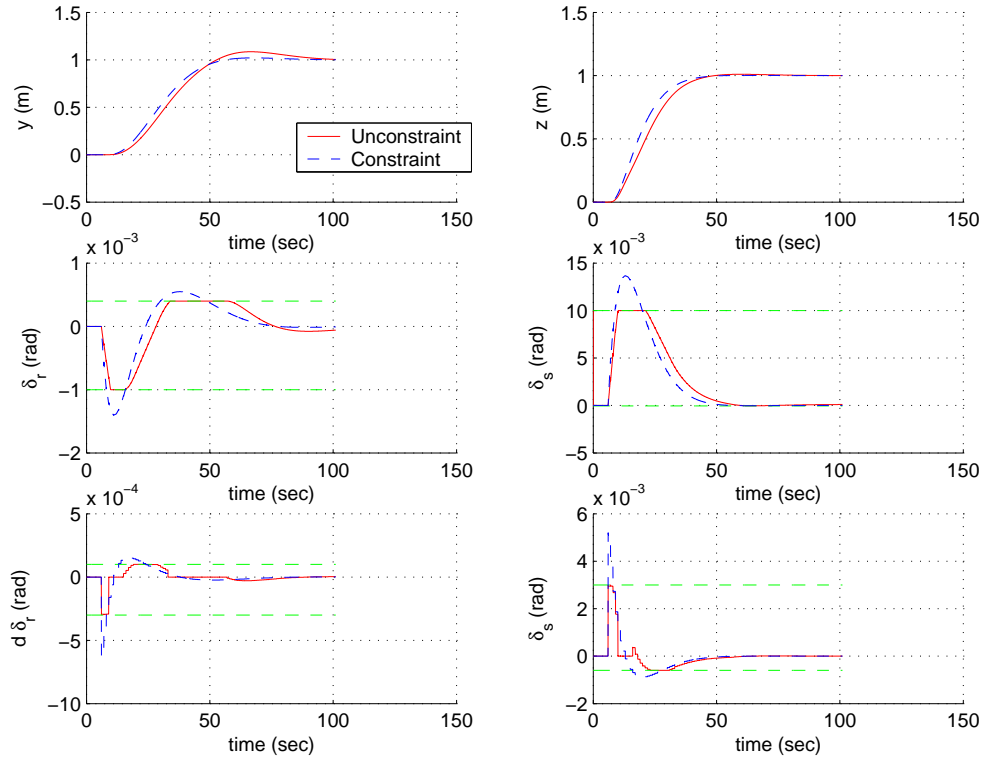


Figure 6.8: Continuous time Intermittent MPC with and without constraints.

Chapter 6. Autonomous Underwater Vehicles

6.6.2 Nonlinear Model Predictive Control

$$\begin{aligned}
 \dot{y} &= \psi U + \phi w \theta & (6.16) \\
 \dot{z} &= -U \theta \\
 \dot{\theta} &= q - \phi r \\
 \dot{\psi} &= \phi q + r \\
 \dot{r} &= Y_{\dot{r}}^{-1} (mUr - my_g r^2 + mz_g qr + \frac{1}{2} \rho CS X_{fin} Ur - X_{\dot{u}} Ur) - Y_{\dot{r}}^{-1} U X_1 u_1 \\
 \dot{q} &= Z_{\dot{q}}^{-1} (-mUq - mz_g q^2 + my_g r q - \frac{1}{2} \rho CS X_{fin} Uq - X_{\dot{u}} Uq + Z_{\dot{q}}^{-1} U X_1 u_2) \\
 y_1 &= y \\
 y_2 &= z
 \end{aligned}$$

The nonlinear dynamic equations of motion of the AUV are derived. The equations in 6 degree of freedom are constructed as in (6.16). The state variables are $[y, z, \theta, \psi, r, q]^T$. The outputs and inputs are $[y, z]^T$ and $[u_1, u_2]^T$, respectively. The NMPC design using Laguerre functions is used to control the system. The tuning parameters are:

$$T_{p1} = T_{p2} = 150, \quad p_1 = p_2 = 1/15, \quad N_1 = N_2 = 3$$

The figure (6.9) shows the movements of the AUV in y , and z directions, together with the control input signals u_1 and u_2 .

6.6 Case Studies

The NMPC using Laguerre function is used to control the system. The tuning parameters are;

$$T_{p1} = T_{p2} = 150, \quad p_1 = p_2 = 1/15, \quad N_1 = N_2 = 3$$

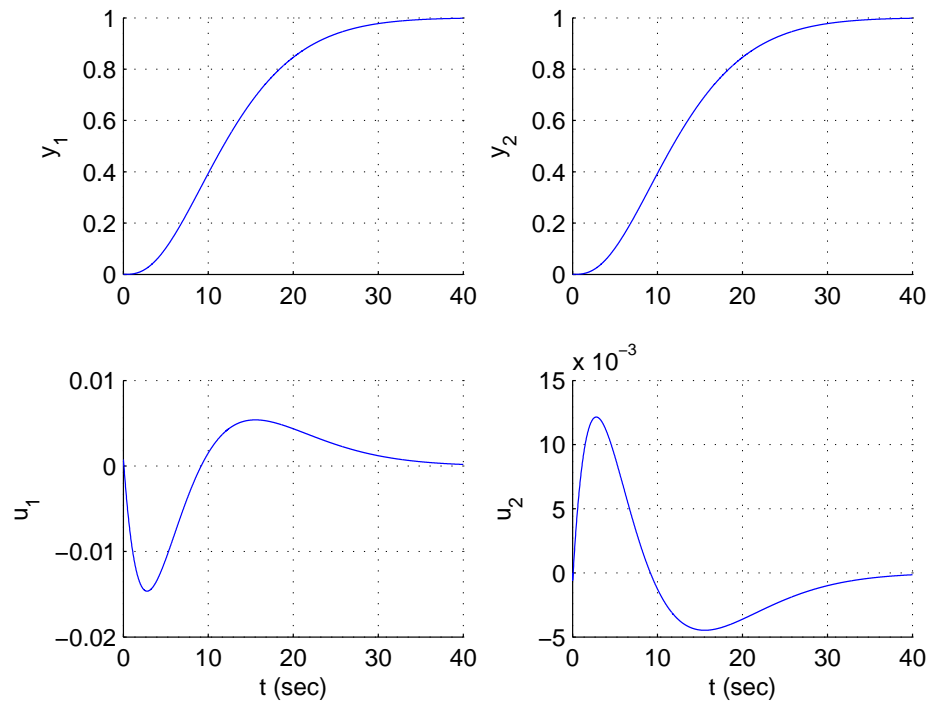


Figure 6.9: Nonlinear MPC to control the AUV

Chapter 6. Autonomous Underwater Vehicles

6.7 Conclusions

The model for the AUV in six degree of freedom works very well. In this thesis, we are interested in the y and z direction, that is why only the y and z are simulated. With this model and the predictive control methods, we can also simulate for the pitch rate q , pitch angle θ , yaw rate r , and yaw angle ψ and all other variables that we want.

This chapter once again confirms the performances of the Intermittent MPC and Nonlinear MPC. From these method, the linear as well as nonlinear process plant can be controlled by the MPC methods.

Chapter 7

Conclusion

7.0.1 Conclusions

The thesis has made new contribution in term of new methodologies such as Intermittent Model Predictive Control approach, two different Nonlinear Model Predictive Control approaches.

The Intermittent MPC approach is very useful for the applications which have fast sampling time. The method inherits all the merits of the original MPC. It also has the advantage of the Intermittent Control method. The method can handle well the constrained problems. Its performance is validated by many case studies.

The Nonlinear Model Predictive Control approaches are used to optimize the nonlinear process plant. In this thesis, the NMPC approaches are applied to control the systems that have nonlinear inputs, nonlinear outputs, unknown constant parameters and disturbances. The approaches produce reliable solutions and fast computational speed. These NMPC approaches also show the ability in handling constrained problems.

Several Quadratic Programming methods are reviewed. The advantages and drawbacks of each methods are pointed out. The Active Set method and Hildreth's Quadratic Programming method are suggested to use when the constraints were changed during simulation. The Interior Point Primal Dual method should be used when the applications have large scale and constant constraints.

Chapter 7. Conclusion

7.0.2 Future Works

There are some questions, which are addressed to guide to the possible future works, include:

- The Intermittent Control method and Nonlinear Model Predictive Control design using Laguerre functions can be combined in order to form the continuous time nonlinear intermittent model predictive control. The new approach may handle the nonlinear plants which operate in fast sampling environment.
- The Interior Point Primal Dual method can be applied to improve the performance of the MPC. The nonlinear MPC can be used for that applications.
- The Active Set Quadratic Programming methods are able to handle the nonlinear medium scale constraints. The new QP methods, which can handle the large scale and nonlinear constraints, should be applied to the nonlinear MPC.

Appendix A

Appendix

A.1 Chapter 3: Intermittent Continuous-time MPC Design to Online Simulation

This section introduces the matlab code to use in the embedded simulink.

```
function y = fcn(H, delta_op, h)

[Ap, Bp, Cp, Dp] = TwoIO(H);
%here comes my multivariable state space model

mb1=2; nb1=2;m1=2;n1=2; p1=1/15; p2=1/15; N1=3; N2=3; Tp1=100;
Tp2=100; p=[p1 p2]; N=[N1 N2]; Tp=[Tp1 Tp2];
[In,Cinf,Setwei]=cmpe_trm();
% [m1,n1]=size(Cp);
%extended state space model
A_e=zeros(n1+m1,n1+m1); A_e(1:n1,1:n1)=Ap;
A_e(n1+1:n1+m1,1:n1)=Cp; B_e=zeros(n1+m1,nb1); B_e(1:n1,:)=Bp;
C_e=zeros(m1,n1+m1); C_e(:,n1+1:n1+m1)=eye(m1,m1);
op1=2*max(p); %poles of the observer
Po=[ -0.1333 -0.6333 -1.1333 -1.6333];\
J = [.428 -.2502 1.6791 -.428; -.3151 .2817 -.4466 1.593]';
xp=zeros(n1,1); % initialize the state variable for the process

nsim = 100; ndelay = 5; r1=[zeros(1,10) ones(1, 100) ];
r2=[zeros(1,10) ones(1, 100) ]; X_hat=zeros(n1+m1,1);
```

Appendix A. Appendix

```

dis=ones(nsim+50,1);
h = .02; % sampling rate
delta_op = 4; % optimised open-loop interval
sample = floor(delta_op/h);% number of sample
num = 3; % number of predictive taken in the open-loop
% Calculate Laguerre matrices
A01=-p1*eye(N1,N1); for ii=1:N1
    for jj=1:N1
        if jj<ii
            A01(ii,jj)=-2*p1;
        end
    end
end
A02=-p2*eye(N2,N2); for ii=1:N2
    for jj=1:N2
        if jj<ii
            A02(ii,jj)=-2*p2;
        end
    end
end
L0=sqrt(2*p1)*ones(N1,1);
% L(1)...L(num)
L1 =zeros(num,num); for k1 = 1:num
    temp = A01*k1*h;
    temp1 = eye(3) + temp + temp^2/2 + temp^3/6 + temp^4/24;
    L1(:,k1)= temp1*L0;
end
L2 = zeros(num, sample-num-1); if(sample-num-1 >0)
    for k2 =1:(sample-num-1)
        temp2 = A01*(num+1)*h;
        temp3 = eye(3) + temp2 + ...
            temp2^2/2 + temp2^3/6 + temp2^4/24;
        L2(:,k2)= temp3*L0;
    end
    L11 = [L0, L1, L2];
else
    L11 = [L0, L1];
end
L12 = L11;
%%%%%%
[nn1,mm]=size(Bp); tam = Ap*h; tam1 = eye(nn1) + tam + tam^2/2 +
tam^3/6 + tam^4/24; Ap_delta=(tam1-eye(nn1))/h; B01=eye(nn1,nn1);
nm=1; for kk=1:500;
    nm=nm*(kk+1);
    B01=B01+(Ap*h).^kk/nm;
end
Bp_delta=B01*Bp;
%%%%%%
[nn2,mm2]=size(B_e); tam2 = A_e*h; tam3 = eye(nn2) + tam2 +

```

A.1 Chapter 3: Intermittent Continuous-time MPC Design to Online Simulation

```

tam2^2/2 + tam2^3/6 + tam2^4/24; A_delta1=(tam3-eye(nn2))/h;
B02=eye(nn2,nn2); nm2=1; for kk=1:500;
    nm2=nm2*(kk+1);
    B02=B02+(A_e*h).^kk/nm2;
end B_e_delta=B02*B_e;
%%%%%%
[nn3,mm]=size(B_e); tam4 = A_e*h; tam5 = eye(nn3) + tam4 +
tam4^2/2 + tam4^3/6 + tam4^4/24; A_e_delta=(tam5-eye(nn3))/h;
B03=eye(nn3,nn3); nm3=1; for kk=1:500;
    nm3=nm3*(kk+1);
    B03=B03+(A_e*h).^kk/nm3;
end
% J_d_delta=B03*J;
%%%%%%
Su=zeros(2,1); t_ini=0; u1 = zeros(1,20198); u2 = zeros(1,20198);
y1 = zeros(1,20198); y2 = zeros(1,20198); y=zeros(m1,1);
y1_temp=zeros(1,sample); y2_temp = zeros(1,sample);
u1_temp=zeros(1,sample); u2_temp = zeros(1,sample); u =
zeros(2,1); udot0 = zeros(2,sample); for i=(ndelay):(nsim+ndelay)
    %Least squares solutions
    eta=(In)\(Setwei*[r1(i) r2(i)]'-Cinf*X_hat);
    udot01 = L11'*eta(1:N1,1);
    udot02 = L12'*eta(N1+1:N1+N2,1);
    % for k =1
    for k = 1:sample
        if(k==1)
            u1_temp(:,1)=u1(sample*(i-ndelay)+1)+udot01(k)*h;
            u2_temp(:,1)=u2(sample*(i-ndelay)+1)+udot01(k)*h;
            udot0(:,1) = [udot01(k,:) udot02(k,:)]';
            Su = Su + h*udot0(:,1);
            u =Su*h;
            xp=(Ap_delta*xp+Bp_delta*u)*h+xp;
            y = Cp*xp;
            X_hat=X_hat+(A_e_delta*X_hat+...
                J*(y-C_e*X_hat))*h+B_e_delta*udot0(:,1)*h;
            y1_temp(1)=y(1);
            y2_temp(1)=y(2);
        else
            u1_temp(:,k) = u1_temp(:,k-1) + udot01(k)*h;
            u2_temp(:,k) = u2_temp(:,k-1) + udot02(k)*h;
            udot0=[udot01 udot02]';
            Su=Su+ udot0(:,k);
            u=Su*h;
            xp=(Ap_delta*xp+Bp_delta*u)*h+xp;

```

Appendix A. Appendix

```

        y = Cp*xp;
        X_hat=X_hat+(A_e_delta*X_hat...
            +J*(y-C_e*X_hat))*h+B_e_delta*udot0(:,k-1)*h;
        y1_temp(k)=y(1);
        y2_temp(k)=y(2);
%         plot(y1_temp)
    end
end
end

function [A, B, C, D] = TwoIO(H);
H = H/100*0.5; % height is 50cm
As = 0.0176; % Cross section area of tank 2, 15cm dia
Av = 4.9e-4/2; % Valve cross section 50%
h0 = 0.05;

A =[-sqrt(9.81/2)*(Av/As)*(1/sqrt(H)) (Av/As)*sqrt(9.81/2/H);
    (Av/As)*sqrt(9.81/2/H) -sqrt(9.81/2)*(Av/As)*(1/sqrt(h0)+1/sqrt(H))];
B =[1/As 0; 0 1/As]; C = [1 0; 0 1]; D =zeros(2);

function [In,Cinf,Setwei]= cmpc_trm();

In= 1e9*[2.6684 -1.7067 0.6543 0.9653 -0.6141 0.2324
-1.7067 1.2582 -0.5709 -0.6141 0.4484 -0.2004
0.6543 -0.5709 0.3433 0.2324 -0.2004 0.1176
0.9653 -0.6141 0.2324 0.5099 -0.3335 0.1346
-0.6141 0.4484 -0.2004 -0.3335 0.2556 -0.1227
0.2324 -0.2004 0.1176 0.1346 -0.1227 0.0803];
Cinf = 1e6*[ 9.8461 3.5697 0.4426 0.1261
-5.9236 -2.1403 -0.2456 -0.0667
2.1481 0.7648 0.0865 0.0205
3.5697 1.8640 0.1261 0.1605
-2.1403 -1.1378 -0.0667 -0.0965
0.7648 0.4379 0.0205 0.0406];
Setwei = 1e5*[ 4.4256 1.2612;
-2.4565 -0.6669;
0.8653 0.2055;
1.2612 1.6054;
-0.6669 -0.9653;
0.2055 0.4058];

```

A.2 Chapter 6: Autonomous Underwater Vehicles

A.2 Chapter 6: Autonomous Underwater Vehicles

A.2.1 Equations for Coefficients

1. Coefficient of pitch moment induced by pitch rate M_q :

$$M_q = M_{qc} + M_{qa} + M_{qf}$$

where:

- Crossflow Drag M_{qc}

$$M_{qc} = -\frac{1}{2}\rho C_{dc} m_q \int_{x_t}^{x_{b2}} 2x^3 R(x) dx - 2x_{fin}^3 \left(\frac{1}{2}\rho S_{fin} C_{dc} m_q\right)$$

- Added mass M_{qa}

$$M_{qa} = -Z_{\dot{q}} U$$

- Fin Lift M_{qf}

$$M_{qf} = -\frac{1}{2}\rho C_{L\alpha} S_{fin} x_{fin}^2 U$$

2. Added mass due to pitch rate $Z_{\dot{q}}$

where:

$$Z_{\dot{q}} = \int_{x_t}^{x_f} x m_a(x) dx - \int_{x_f}^{x_{f2}} x m_{af}(x) dx - \int_{x_{f2}}^{x_{b2}} x m_a(x) dx$$

3. Coefficient of yaw moment due to yaw rate N_r :

$$N_r = N_{urf} + Y_{\dot{r}} U$$

where:

$$Y_{\dot{r}} = -Z_{\dot{q}}$$

$$N_{urf} = -\rho C_{L\alpha} S_{fin} x_{fin}^2$$

4. Added mass moment of inertia due to yaw rate $N_{\dot{r}}$:

$$N_{\dot{r}} = \int_{x_{tail}}^{x_{fin}} x^2 m_a(x) dx - \int_{x_{fin}}^{x_{ffin2}} x^2 m_{af}(x) dx - \int_{x_{fin2}}^{x_{bow2}} x^2 m_a(x) dx$$

Appendix A. Appendix

5. Fin Lift coefficients:

$$M_{uu\delta s} = N_{uu\delta r} = \rho C_{L\alpha} S_{fin} x_{fin}$$

6. Hydrostatic coefficients:

$$M_\theta = -(z_g W - z_b B)$$

$$N_\theta = -(y_g W - y_b B)$$

A.2 Chapter 6: Autonomous Underwater Vehicles

A.2.2 Values of Coefficients

The values of the coefficients are taken from the real AUV named REMUS [94] for modelling.

Nonlinear force and moment coefficients

Parameter	Values	Units	Description
M_θ	-5.77	$kg.m^2/s^2$	hydrostatic
M_q	-6.87	$kg.m^2/s^2$	pitch moment coefficient
$M_{\dot{q}}$	-4.88	$kg.m^2/s^2$	added mass
$M_{uu\delta s}$	-6.15	kg/rad	fin lift moment
M_{qc}	-2.16	$kg.m^2/s^2$	fin lift
M_{qa}	2.97	$kg.m^2/s^2$	cross flow drag
M_{qf}	-7.68	$kg.m^2/s^2$	added mass
N_r	-9.4	$kg.m^2/s^2$	yaw moment coefficient
$N_{\dot{r}}$	-4.88	$kg.m^2/s^2$	added mass
N_{uui}	-6.15	kg/rad	fin lift moment
N_{urf}	-3.93	kg.m/rad	fin moment coefficient
N_θ	0	$kg.m^2/s^2$	hydrostatic
$Z_{\dot{q}}$	-1.93	kg.m/s	added mass

Appendix A. Appendix

AUV's Parameters

Parameter	Values	Units	Description
x_g	0	m	center of gravity
y_g	0	m	center of gravity
z_g	0.00196	m	center of gravity
x_b	-0.0611	m	center of buoyancy
y_b	0	m	center of buoyancy
z_b	0	m	center of buoyancy
U	1.54	m/s	surge velocity
V	0.5	m/s	sway velocity
w	1.54	m/s	heave velocity
W	309	N	vehicle weight
B	306	N	vehicle buoyancy
ρ	999.3	kg/m ³	fluid density
I_x	0.177	kg/m ²	moment of inertia
I_y	3.45	kg/m ²	moment of inertia
I_z	3.45	kg/m ²	moment of inertia
S_{fin}	$6.65 * 10^{-3}$	m ²	platform area
x_{fin}	-0.819	m	fin height
a_{fin}	5.14	m	fin height
t	0.654	n/a	fin taper ratio
$C_{L\alpha}$	3.12	n/a	fin lift slope
C_{df}	0.558	n/a	fin crossflow drag coefficient
b_{fin}	0.0857	n/a	span

Bibliography

- [1] A. Healey, and D. Lienard, *Multivariable sliding model control for autonomous diving and steering of unmanned underwater vehicles*, IEEE Journal of Oceanic Engineering, Vol 18, No. 3, pp 327 - 339, 1993.
- [2] A. Healey, J. Kim, *Modeling and simulation methodology for reconnaissance in VSM minefields with multiple autonomous vehicles*, Proceedings SPIE Conference, Orlando Florida, 1999.
- [3] A. Jealey, and D. B. Marco, *Command, control, and navigation experimental results with the NPC Aries AUV*, Monterey, CA, Dept. of Mechanical Engineering, Vol 25, 2000.
- [4] A. Healey, *Application of Formation Control for Multiple Vehicle Robotic Minesweeping*. Paper CDC-INV-3103, Proceedings of the IEEE CDC-2001, Orlando, FL, December 2001.
- [5] Adrian Gambier, and Heinz Unbehauen, *Multivariable generalized state space receding horizon control in a real time environment*, Automatica, vol. 35, pp 1787-1797, 1999.
- [6] Antoni Podsiadly, *Continuous time system identification of a food cooker extruder*, Newcastle University, 2000. [http: murray.newcastle.edu.au](http://murray.newcastle.edu.au)
- [7] B. A. Foss, T. A. Johansen, and A.a V. Sorensen, *Nonlinear predictive control using local models - applied to a batch fermentation process*, Control Engineering Practice, Vol. 3, pp 389 - 396, 1995.
- [8] B. W. Bequette, *Nonlinear predictive control using multirate sampling*, Can. J. Chem. Eng. Vol 69, pp 136 - 143, 1991.
- [9] B. Kouvaritakis, M. Cannon, and J. A. Rossiter, *Recent development in generalized predictive control for continuous time system*, International Journal of Control, Vol 72, pp 164 - 173, 1999.

BIBLIOGRAPHY

- [10] C. V. Rao, S. J. Wright, and J. B. Rawlings, *Application of interior point methods to model predictive control*, Journal of optimization theory and applications, Vol 99, No 3, pp 723 - 757, 1998.
- [11] C. E. Garcia, *Quadratic/ Dynamic Matrix Control of nonlinear processes: An application to a batch reaction process*, AIChE Annual Meeting, San Francisco, CA, 1984.
- [12] C. E. Garcia, D. M. Prett, and M. Morari, *Model Predictive Control: theory and practice - a survey*, Automatica, Vol 25, pp 335 - 349, 1989.
- [13] C. R. Cutler and B. L. Ramaker, *Dynamic matrix control - A computer control algorithm*, Presented at the Meeting of the American Institute of Chemical Engineers, Houston, Texas, 1979.
- [14] D. G. Luenberger, *Optimization by vector space methods*, New York, Wiley professional paperback series, 1969.
- [15] D. Ljungquist, and J. G. Balchen, *Recursive prediction error methods for online estimation in nonlinear state space models*, Modeling, Identification, and Control, Vol. 15, pp 109 - 121, 1994.
- [16] D. Q. Mayne, and H. Michalska, *Receding horizon control of nonlinear system*, IEEE Trans. On Autom. Control, vol. 35 no 37, pp 814 - 824, 1990.
- [17] D. Q. Mayne, *Nonlinear model predictive control: An assessment*, Chemical Process Control, Vol 5, 1997.
- [18] D. Q. Mayne, J. Rawlings, C. Rao, and P. Scokaert, *Model predictive Control: satiability and optimality*, Automatica, Vol 36, 2000.
- [19] D. W. Clarke, and L. Zhang, *Long range predictive control using weighting sequence models*, IEE Proc. Pt D, Vol. 34, pp 187 - 195, 1987.
- [20] E. F. Camacho, and C. Bordons, *Model Predictive Control*, Springer Verlag, 1999.
- [21] Edoardo Mosca, *Optimal, predictive and adaptive control*, Prentice-Hall, 1995.
- [22] Eric Ronco, Peter J Gawthrop and David J Hill (1999a). *Open-loop intermittent feedback optimal control: reference manual to an on-line simulation package*. EE-99002, Laboratory for Dynamical Systems and Control, Sydney University.
- [23] Eric Ronco, and David J Hill (1999b). *Open-loop intermittent feedback optimal predictive control: a human movement control model*. EE-99003. Sydney, School of Electrical and Information Engineering, Sydney University.

BIBLIOGRAPHY

- [24] Eric Ronco, T. Arsan and Peter J Gawthrop (1999c). *Open-loop intermittent feedback control: Practical continuous-time GPC*. IEE Proceedings Control Theory Application Vol. 146, No. 5.
- [25] F. Kappel, A. V. Kuntsevich, *An implementation of Shor's r-Algorithm*, Computational optimization and applications, Vol. 15, pp 193 – 205, 2000.
- [26] F. Martinsen, L.T. ZBiegler, B.A. Foss, *A new optimization algorithm with application to nonlinear MPC*, Journal of process control, Vol. 14, 853-865, 2004.
- [27] G. Astfalk, I. Lusitg, R. Marsten, D. Shanno, *The interior point method for linear programming*, IEEE, 1992.
- [28] G. Gattu, and E. Zafiriou, *Nonlinear quadratic dynamic matrix control with state estimation*, Ind. Eng. Che. Res. Vol 31, pp 1096 - 1104, 1992.
- [29] G. Gattu, and E. Zafiriou, *Observer based nonlinear quadratic dynamic matrix control for state space and input/ouput models*, The Canadia Journal of Chem. Eng., Vol 73, 883 - 895, 1995.
- [30] G. R. Srinivas, and Y. Arkum, *A global solution to the nonlinear model predictive control problem using polynomial ARX models*, Computers Chem. Engng, Vol 21, pp 431 - 439, 1997.
- [31] G. W. Stewart, *The decompositional approach to matrix computation*, IEEE, 2000.
- [32] G. C. Goodwin, S. Graebe, and M. Salgado, *Control system design*, Prentice Hall, Englewood Cliffs, NJ, 2000.
- [33] H. Demircioglu, and P. J. Gawthrop, *Continuous time generalized predictive control*, Automatica, Vol 27, No. 1, pp 55 - 74, 1991.
- [34] H. Kondo, T. Ura, *Navigation of the autonomous underwater vehicle for investigation of underwater structures*, Tokyo, Institute of Industrial Science, The University of Tokyo.
- [35] H. Takatsu, I. Toshiaki, and M. Araki, *Future needs for the control theory in industries - report and topics of the control technology survey in Japanese industry*, Journal of Proceedings of Control, vol. 8, no 5-6, pp 369 - 374, 1998.
- [36] L. Ljung, *System Identification: Theory for the user*, Second Edition, Prentice Hall Ptr, 1999.

BIBLIOGRAPHY

- [37] K. O. Temeng, P. D. Schnelle, and T. J. McAvoy, *Model predictive control of an industrial packed bed reactor using neural networks*, Journal of Process Control, Vol 5, pp 19 - 27, 1995.
- [38] M. Cannon and B. Kouvaritakis, *Infinite horizon predictive control of constrained continuous time linear system*, Automatica, Vol. 36, pp 943 - 956, 2000.
- [39] N. L. Ricker, and J. H. Lee, *Nonlinear model predictive control of the Tennessee Eastman challenge process*, Computers Chem. Engng, Vol 19, pp 961 - 981, 1995.
- [40] N. L. Ricker, and J. H. Lee, *Nonlinear modeling and state estimation for the Tennessee Eastman challenge process*, Computers Chem. Engne, Vol 19, pp 983 - 1005.
- [41] L. Wang, *Frequency-sampling filters: an improved model structure for step response identification*, Automatica, Vol 33, pp 939-944, 1997.
- [42] L. Wang and W. R. Cluett, *From plant data to process control: Ideas for process identification and PID design*. Taylor and Francis, London, 2000.
- [43] L. Wang, *Continuous time model predictive control using orthonormal functions*, International Journal of Control 74: 1588 - 1600, 2001.
- [44] L. Wang and P. J. Gawthrop, *On the estimation of continuous time transfer functions*, International Journal of Control, Vol. 74, pp 889-904, 2001.
- [45] L. Wang, *A Tutorial on model predictive control: Using a linear velocity-form model*. Sydney, Department of Electrical and Computer Engineering, University of Newcastle, 2002. (also Proceedings of the 4th Asian Control Conference, Singapore).
- [46] L. Wang, P. J. Gawthrop, C. Chessari, and T. Podsiadly, *Continuous time system identification of a food extruder: experiment design and data analysis*, Proceedings of the 13th IFAC Symposium on System Identification (Rotterdam, The Netherlands), Aug 2003, pp. 639 - 634.
- [47] L. Wang, *Discrete model predictive controller design using Laguerre functions*, Journal of Process Control Vol. 14, pp 131142, 2004.
- [48] J. A. Nelder, and R. Mead, *Computer Journal*, Vol. 7, pp. 309 - 313, 1965.
- [49] J. A. Rossiter, B. Kouvaritakis, and M. J. Rice, *A numerically stable robust state space approach to stable predictive control strategies*, Automatica, Vol. 34, pp 65 - 73, 1998.

BIBLIOGRAPHY

- [50] J. A. Rossiter, *Model based predictive control: A practical approach*, CRC press, Boca Raton, 2003.
- [51] J. E. Dennis, R. B. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Englewood Cliffs, NJ: Prentice Hall, 1993.
- [52] J. H. Lee, M. Morari, and C. E. Garcia, *State space interpretation of model predictive control*, *Automatica*, Vol 30. pp 707 - 717, 1994.
- [53] J. K. Hedrick, A. Girard, *Control of nonlinear dynamic systems: Theory and Applications*, 2005.
- [54] J. J. Keller, *Tracking control of autonomous underwater vehicles*, PhD dissertation, Naval postgraduate school, Monterey, CA, 2002.
- [55] J. M. Maciejowski, *Predictive control; with constraints*, Harlow, New York, Prentice Hall, 2002.
- [56] J. Richalet, A. Rault, J. L. Testud, and J. Papon, *Model predictive heuristic control: Applications to industrial processes*, *Automatica*, Vol. 14, pp 413 - 428, 1978.
- [57] J. W. Eaton, and J. B. Rawlings, *Feedback control of nonlinear processes using online optimization techniques*, *Comput. Chem. Eng.* Vol 1, pp 469 - 479, 1990.
- [58] M. A. Abokowitz, *Stability and motion control ocean vehicle : organization, development, and initial notes of a course of instruction in the subject*, the University of MIT, 1969.
- [59] M. Cannon and B. Kouvaritakis (2000). *Infinite horizon predictive control of constrained continuous time linear systems*. *Automatica* Vol 36, 943-956.
- [60] M. Nikolaou, *Model predictive controllers; A critical synthesis of theory and industrial needs*, *Advances in Chemical Engineering Seriew*, Academic Press, 2001.
- [61] N. J. Ricker, *model predictive control: state of art*, Proc. Fourth International Conference on Chemical Process Control, Padre Island, Texas, pp 271 - 296, 1991.
- [62] N. M. C. de Oliveira, and L. T. Biegler, *An extension of Newton-type algorithms for nonlinear process control*, *Automatica*, Vol 31, pp 281 - 286, 1995.
- [63] N. Karmarkar, *A new polynomial-time algorithm for linear programming*, *Combinatorica*, pp. 373-395, 1984.

BIBLIOGRAPHY

- [64] N. Z. Shor, *Minimization methods for non-differential functions*, Springer-Verlag: Berlin, Springer Series in Computational Mathematics, Vol. 3, 1985.
- [65] N. Z. Shor, *Monotone modifications of r -Algorithms and their applications*, Cybernetics and Systems Analytics, Vol. 38, No. 6, 2002.
- [66] P. E. Gill, W. Murray, and M.H. Wright, *Practical Optimization*, London; New York: Academic Press, 1981.
- [67] P. Hr. Petkov, N. D. Christov, and M. M. Konstantinov, *Computational methods for linear control systems*, Englewood Cliffs, NJ: Prentice Hall, 1991.
- [68] P. Ridley, J. Fontan, P. Corke, *Submarine dynamic modelling*, Brisbane, Australia, School of mechanical engineering, Queensland University of Technology, 2002.
- [69] Peter J Gawthrop, H. Demircioglu, and L. Siller Alcala, *Multivariable continuous generalised predictive control: A state space approach to linear and nonlinear systems*, Proc. IEE Pt.D: Control Theory and Applications, 1998.
- [70] Peter J Gawthrop, Eric Ronco (1999). *Predictive Pole-placement Control with Linear Models*. Elsevier.
- [71] Peter J Gawthrop (2002). *Physical model-based intermittent predictive control*. 8th IEEE International Conference on Methods and Models in Automation and Robotics, 707-712.
- [72] Peter J Gawthrop (2004a). *Intermittent constrained predictive control of mechanical systems*, Centre for Systems and Control and Department of Mechanical Engineering, University of Galsgow, Scotland.
- [73] Peter J Gawthrop, Liuping Wang (2004b). *Intermittent Predictive Control of an Inverted Pendulum*, submitted to Elsevier.
- [74] P. S. Stanimirovic, N. V. Stojkovic, V. V. K. Vujcic, *Stabilization of Mehrotra's primal dual algorithm and its implementation*, European Journal of Operational Research, Vol 165, pp 598 - 609, 2005.
- [75] Q. Truong, L. Wang, and P. Gawthrop, *Intermittent Model Predictive Control of an Autonomous Underwater Vehicle*, 9th International Conference on Control, Automation, Robotics and Vision (ICCARV), 2006.
- [76] R. A. Bartlett, A. Wachter, and L. T. Biegler, *Active set vs. Interior point strategies for model predictive control*, Preceddings of the American Control Conference, Chicago, 2000.

BIBLIOGRAPHY

- [77] R. Fletcher, *A general quadratic programming algorithm*, Journal of the Institute of Mathematics and its Applications, Vol 7, pp 76 - 91, 1971.
- [78] R. Fletcher, *Practical methods of optimization*, 2nd Edition, Chichester, Wiley, 2000.
- [79] R. Gessing *State Space and Internal Models for Linear Quadratic Regulator Design*, Preprints of IV-th DYCOMANS Workshop, Zakopane, Poland, pp. 7-13, 1997.
- [80] R. Gessing *Continuous-Time Linear-Quadratic Regulator with Output Feedback*, Proceedings of the American Control Conference, Chicago, USA, pp.877-881, 2000.
- [81] R. Marino, P. Tomei, *Nonlinear control design: geometric, adaptive, and robust*, New York, Prentice Hall,1995.
- [82] S. B. J. Jang, and H. Mukai, *Control of constrained multivariable nonlinear process using a two phase approach*, Ind. Eng. Chem. Res. Vol. 26, pp 2106 - 2114, 1987.
- [83] S. G. Nash, A Sofer, *Linear and nonlinear programming*, New York, McGraw Hill, 1996.
- [84] S. L . Shah, *A tutorial introduction to constrained long range predictive control*, Pulp and Paper Canada, Vol. 94, pp57-63, 1995.
- [85] S. Mehrotra, *On the implementation of a primal dual interior point method*, SIAM Journal of Optimization, Vol. 2, No. 4, pp 575 - 601, 1992.
- [86] S. J. Qin, and T.A. Badgwell, *An overview of industrial model predictive control technology*. In Preprints of Chemical Process Control - CPC V, California, 1996.
- [87] S. J. Qin, T. A. Badgwell, *A survey of industrial model predictive control technology*, Control Engineering Practice, Vol 11, Issue 7, pp 733-764, July 2003.
- [88] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM Publications Philadelphia, Pennsylvania, 1997.
- [89] S. Smith, L. Wang, C. Chessari, and J. Chen, *Continuous time model predictive control of a food extruder*, prepared for Food Science Australia, 2000.
- [90] Saint-Donat, J. N. Bhat, and T. J. McAvoy, *Neural net based model predictive control*, Int. J. Control, Vol 54, pp 1453 - 1468, 1991.

BIBLIOGRAPHY

- [91] T. A. Johansen, and B. A. Foss, *Operating regime based process modeling and identification*, Computers Chem. Engng, Vol 21, pp 159 - 176, 1997.
- [92] T. E. Peterson, Y. A. Hernandez, and F. J. Schork, *Nonlinear predictive control of a semi batch polymerization reactor by an extended DMC*, Proc. American Control Conference, pp 1534 - 1539, 1989.
- [93] T. E. Peterson, Y. A. Hernandez, and F. J. Schork, *A nonlinear DMC algorithm and its application to a semibatch polymerization reactor*, Chem. Engng Sci, Vol 47, pp 737 - 753, 1992.
- [94] T. Prestero, *Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle*, PhD dissertation, Massachusetts Institute of Technology, 2001.
- [95] T. S. Schei, and T. A. Johansen, *Nonlinear model based/ model predictive control with constraints and with/ without nonlinear observer*, Sintef Automatic Control.
- [96] V. Peterka, *Predictor based self tuning control*, Automatica, Vol. 20, pp 39 - 50, 1985.
- [97] W. C. Li, and L. T. Biegler, *Process control strategies for constrained nonlinear systems*, Ind. Eng. Chem. Res, Vol 27, pp 1421 - 1433, 1998.
- [98] Y. Nakamori, K. Suzuki, and T. Yamanaka, *A new design of a fuzzy model predictive control system for nonlinear processes*, In Fuzzy Engineering Toward Human Friendly System, T. Terano, M. Sugeno, M. Mukaidono, and K. Shigemasu, IOS Press, Amsterdam, pp 788 - 799, 1992.
- [99] W. C. Li, and L. T. Biegler, *Multistep, Newton-type control strategies for constrained, nonlinear process*, Chem. Eng. Res. Des. Vol 67, pp 562 - 577, 1989.
- [100] W. Naeem, *Model predictive control of an autonomous underwater vehicle*, UK, Department of Mechanical and Marine Engineering, the University of Plymouth.
- [101] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, *Numerical recipes in C++: The art of scientific computing*, Cambridge University Press, 2002.
- [102] Y. W. Lee, *Statistical Theory of Communication*, John Wiley and Sons, New York, 1960.
- [103] Autonomous Underwater Vehicles Resources, 2004, <http://www.cacs.louisiana.edu/kimon/AUV>