การควบคุมแบบออนไลน์ออพติมัลของการผลิตเอทานอลในเครื่องปฏิกรณ์แบบเฟดแบตช์ โดยการใช้ข่ายงานนิวรัลเป็นตัวประมาณค่า

นายณัฐพงศ์ ชมโฉม

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมเคมี ภาควิชาวิศวกรรมเคมี
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2549

ISBN 974-14-3391-3

ON-LINE OPTIMAL CONTROL OF ETHANOL PRODUCTION IN A FED-BATCH REACTOR BY USING NEURAL NETWORK ESTIMATOR

Mr. Natthapong Shomchoam

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Chemical Engineering
Department of Chemical Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2006
ISBN 974-14-3391-3

Thesis Title	ON-LINE OPTIMAL CONTROL OF ETHANOL PRODUCTION IN A FED-BATCH REACTOR BY USING NEURAL NETWORK ESTIMATOR
By	Mr. Natthapong Shomchoam
Field of Study	Chemical Engineering
Thesis Advisor	Amornchai Arpornwichanop, D.Eng.

Accepted by the Faculty of Engineering, Chulalongkorn Univesity in Partial Fulfillment of the Requirements for the Mater's Degree

D.L Lavansiri ..... Dean of the Faculty of Engineering

(Professor Direk Lavansiri, Ph.D.)

THESIS COMMITTEE

Paisan Kittisupakorn ..... Chairman

(Associate Professor Paisan Kittisupakorn, Ph.D.)

Amornchai Arpornwichanop ..... Thesis Advisor

(Amornchai Arpornwichanop, D.Eng.)

Mel Phey ..... Member

(Assistant Professor Muenduen Phisalaphong, Ph.D.)

Soorathep Kheawhom ..... Member

(Soorathep Kheawhom, Ph.D.)

ณัฐพงศ์ ชมโฉม : การควบคุมแบบออนไลน์ออพติมัลของการผลิตเอทานอลในเครื่องปฏิกรณ์แบบเฟดแบตช์โดยการใช้ข่ายงานนิวรัลเป็นตัวประมาณค่า (ON-LINE OPTIMAL CONTROL OF ETHANOL PRODUCTION IN A FED-BATCH REACTOR BY USING NEURAL NETWORK ESTIMATOR) อ. ที่ปรึกษา: อ. ดร. อมรชัย อาภรณ์วิชานพ; 97 หน้า. ISBN 974-14-3391-3.

งานวิจัยนี้เสนอการประยุกต์ใช้การควบคุมแบบออนไลน์ออพติมัลร่วมกับตัวประมาณค่าแบบข่ายงานนิวรัลเพื่อควบคุมเครื่องปฏิกรณ์แบบเฟดแบตช์สำหรับการผลิตเอทานอล เนื่องจากมีการรบกวนที่ไม่ทราบค่าและความผิดพลาดของแบบจำลองทำให้การควบคุมแบบออพติมัลแบบวงเปิดไม่สามารถให้สมรรถนะการควบคุมที่เหมาะสมที่สุดได้เมื่อนำไปประยุกต์ใช้กับกระบวนการจริง เพื่อปรับปรุงสมรรถนะการควบคุม การควบคุมแบบออนไลน์ออพติมัลได้ถูกพัฒนาขึ้นเพื่อแก้ไขโพรไฟล์ที่เหมาะสมที่สุดของสารป้อนเข้าของเครื่องปฏิกรณ์แบบเฟดแบตช์เมื่อได้รับข้อมูลป้อนกลับของระบบ ในงานวิจัยนี้ปัญหาการควบคุมแบบออพติมัลที่ถูกกำหนดขึ้นได้ถูกหาคำตอบโดยใช้วิธีการแบบลำดับขั้นโดยที่โพรไฟล์ของตัวแปรควบคุมถูกแบ่งโดยใช้ฟังก์ชันแบบคงที่แบบเป็นช่วงๆ ข่ายงานนิวรัลได้ถูกนำมาใช้เพื่อประมาณค่าตัวแปรสถานะที่วัดไม่ได้ซึ่งจะถูกใช้เป็นข้อมูลป้อนกลับของระบบ กระบวนการหมักเอทานอลโดย *Saccharomyces cerevisiae* ในเครื่องปฏิกรณ์แบบเฟดแบตช์ได้ถูกเลือกเป็นกรณีศึกษาเพื่ออธิบายถึงวิธีการควบคุมที่นำเสนอ ผลการจำลองแสดงให้เห็นว่าการควบคุมแบบออนไลน์ออพติมัลโดยใช้ข่ายงานนิวรัลเป็นตัวประมาณค่าให้สมรรถนะการควบคุมในเชิงของปริมาณผลิตภัณฑ์เอทานอลที่ต้องการได้ดีกว่าเมื่อเปรียบเทียบกับการควบคุมแบบออพติมัล

ภาควิชา วิศวกรรมเคมี ลายมือชื่อนิสิต.....
สาขาวิชา วิศวกรรมเคมี ลายมือชื่ออาจารย์ที่ปรึกษา.....
ปีการศึกษา 2549

4770285821 : MAJOR CHEMICAL ENGINEERING

KEY WORD : ON-LINE OPTIMAL CONTROL / ETHANOL PRODUCTION / FED-BATCH REACTOR / ARTIFICIAL NEURAL NETWORK

NATTHAPONG SHOMCHOAM : ON-LINE OPTIMAL CONTROL OF ETHANOL PRODUCTION IN A FED-BATCH REACTOR BY USING NEURAL NETWORK ESTIMATOR. THESIS ADVISOR : AMORNCHAI ARPORNWICHANOP, D.Eng. 97 pp. ISBN 974-14-3391-3.

This research presents the implementation of an on-line optimal control with neural network estimator to control a fed-batch reactor for the production of ethanol. Due to the presence of unknown disturbances and model-plant mismatches, an open-loop optimal control may not give the optimal performance when applied to the actual process. To improve the control performance, an on-line optimal control is developed to modify the optimal feed profile of a fed-batch reactor whenever feedback information of the system is available. In this work, the formulated optimal control problem is solved by a sequential method in which the control profile is parameterized by using a piecewise constant function. Artificial neural network is used to estimate unmeasured state variables which are employed as feedback information of the system. The ethanol fermentation process by *Saccharomyces cerevisiae* in a fed-batch reactor is chosen as a case study to demonstrate the proposed control strategy. The simulation results have shown that the on-line optimal control with neural network estimator gives a better control performance in terms of the amount of the desired ethanol product, compared with the off-line optimal control.

Department: Chemical Engineering Student's Signature: Natthapong Shomchoam

Field of Study: Chemical Engineering Advisor's Signature: Amornchai Arpornwichanop

Academic Year: 2006

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to my thesis advisor, Dr. Amornchai Arpornwichanop, for his valuable suggestion and encouraging guidance. Furthermore, I am also grateful to Associate Professor Paisan Kittisupakorn as the chairman of thesis committee, Assistant Professor Muenduen Phisalaphong and Dr. Soorathep Kheawhom as the members of thesis committee for many valuable suggestions.

Special thanks go to the graduate school of Chulalongkorn University for financial support of this research.

I am also thanks to all my friends, process control laboratory members for their friendship and assistance.

Most of all, I would like to express the highest gratitude to my parents, and all of family for their love, inspiration, encouragement and financial support throughout this study.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURE PAGE

FIGURE PAGE

NOMENCLATURES

J	performance index	
K_p	kinetic constant of product	[g/L]
K_s	kinetic constant of substrate	[g/L]
p	product concentration	[g/L]
s	substrate concentration	[g/L]
t	time	[hr]
u	feedrate	[L/hr]
v	liquid volume of the fermentor	[L]
x	cell mass concentration	[g/L]
Y	yield coefficient	
μ	specific growth rate	[hr^{-1}]
η	specific productivity	[hr^{-1}]

CHAPTER I

INTRODUCTION

1.1 Motivation

Due to the increasing of fuel cost, ethanol is used as an alternative to petroleum-derived fuels. Ethanol can be used either as a direct replacement for gasoline or in a blend with gasoline as an oxygenated compound. The advantage of using ethanol as an alternative fuel is to increase combustion efficiency, thus reducing carbon monoxide emission, and to reduce the petroleum-derived fuels demand. In the production of ethanol, ethanol can be obtained by chemical synthesis from petroleum product and by fermentation process from agricultural product. However, the production of ethanol by fermentation process has been interested because the raw materials of the ethanol fermentation process are derived from renewable resources.

In the ethanol fermentation process, the operation of the ethanol fermentation can be generally classified into three modes, i.e., batch, fed-batch, and continuous mode. Among these operation modes, the fed-batch fermentation is preferable choice. Fed-batch fermentation is used to prevent or reduce substrate-associated growth inhibition by controlling the substrate supply. In a fed-batch fermentation process, it starts with some initial volume, microorganisms concentration and substrate concentration in a reactor which is also known as a fermentor. The feed of the substrate is continuously supplied into a reactor while the products remain in the reactor until the end of the operation. For such circumstances, it is found that changing feed rate of the substrate affects the productivity and yield of the desired product. Due to this fact, precise control of the substrate feed at its optimal profile is required for operating the fed-batch reactor efficiently, resulting in high productivity and yield of the desired product.

To control a fed-batch reactor, an optimal control technique is usually employed to determine a feeding policy for the reactor, resulting in the maximum

amount of the desired product at the end of operational time. The optimal feeding profile can be determined by solving an optimal control problem which is formulated based on dynamic models of the system to be controlled. This problem is often referred to a dynamic optimization problem. It is known that achieving the optimal feeding profile of the fed-batch reactor is quite difficult and challenging problem due to the highly nonlinear and complicated dynamic behavior of the reactor. In addition, one of the most important issues to be considered for a model-based control strategy is the existence of unknown disturbances and model–plant mismatches. Under these conditions, the optimal feeding profile which is determined from the off-line calculation may not give the optimal performance when apply to the actual process; the final product may significantly differ from the desired value. To realize this fact, it is necessary to recalculate the optimal feeding profile as an on-line optimal control strategy whenever new feedback information is available.

By performing the on-line optimal control strategy, feedback information obtained from process measurements is used to determine a new optimal feeding profile. However, some process variables such as substrate concentration cannot be directly measured. To cope with this problem, a state estimation from available information is involved with the on-line optimal control strategy. Neural network is found to be the one of various tools that can be used for the estimation of unmeasured process variables since it is considered as an universal approximator that can be used to approximate any arbitrary function.

In this research, the approach based on the idea of an on-line optimal control strategy is developed to modify the optimal feeding profile of the substrate in a fed-batch reactor for ethanol production. The proposed on-line optimal control technique is incorporated with an artificial neural network for the estimation of unmeasured process variables. Simulations are carried out to investigate the performance of the developed control technique under the presence of unknown disturbances and model–plant mismatches.

1.2 Research Objective

The objective of the present work is to develop an on-line optimal control strategy based on an artificial neural network estimator for controlling a fed-batch reactor in the ethanol fermentation process.

1.3 Scope of Research

1. Ethanol fermentation process in a fed-batch reactor is studied by simulations in this work.

2. Off-line optimal control and on-line optimal control strategies are used to control a fed-batch reactor.

3. Artificial neural network is applied to estimate the unmeasured process variables in a fed-batch process.

4. Performance of the off-line optimal controller is compared with the on-line optimal controller.

1.4 Procedure Plan

1. Review a basic of optimal control, artificial neural network and ethanol fermentation process.

2. Determine the optimal feeding profile of ethanol fermentation process in a fed-batch reactor with an off-line optimal control strategy.

3. Design an artificial neural network based estimator to estimate the unmeasured process variables of ethanol fermentation process in a fed-batch reactor.

4. Determine the optimal feeding profile of ethanol fermentation process in a fed-batch reactor with an on-line optimal control strategy by using artificial neural network as an estimator.

5. Compare the simulation results between the off-line optimal control strategy and the on-line optimal control strategy by using artificial neural network as an estimator.

6. Discuss the simulation results and make a conclusion.

This thesis is divided into six chapters:

Chapter I is an introduction of this research. This chapter consists of motivation, research objective, scope of research and procedure plan.

Chapter II presents the literature reviews with an optimal control of a fed-batch reactor, the implementation of an artificial neural network and the control of a fed-batch reactor for the ethanol production**.**

Chapter III describes the theoretical background of the optimal control, the artificial neural network and the biological processes.

Chapter IV presents the optimal control of ethanol production in a fed-batch reactor.

Chapter V presents the application of an artificial neural network as a state estimator and the on-line optimal control strategy by using artificial neural network as a state estimator for the ethanol production in a fed-batch reactor.

Chapter VI presents the conclusions of this research and makes the recommendations for future work.

CHAPTER II

LITERATURE REVIEWS

In biological processes, a fed-batch reactor is widely used in a fermentation process for the production of many products such as pharmaceutical and agricultural products. During the fed-batch operation, an inlet feed of substrate is supplied to the reactor while product generated remains in the reactor until the end of the operation. The benefit of the fed-batch reactor is that changing the inlet feed can affect the productivity and yield of the desired product, and avoiding substrate overfeeding which can inhibit the growth of microorganisms. Then, precise control of the substrate feed rate at its optimal operation is required for operating the fed-batch reactor efficiently. However, it has been accepted that control of a fed-batch reactor is difficult and challenging problem due to their highly nonlinear and complicated dynamic behavior. A practical way for controlling the fed-batch reactor is to track a pre-determined trajectory of a controlled variable so as to optimize an objective function and also known as optimal control procedure.

2.1 Optimal Control of a Fed-Batch Reactor

Optimal control problems of a fed-batch reactor have been solved by using different approaches that can be classified into open-loop and closed-loop control strategies. The open-loop control involves the off-line calculation of the optimal feeding profile that attains the fed-batch targets whereas the closed-loop control involves the calculation that the feedback information of the system is used to recalculate the optimal control problem for new optimal feeding profile.

2.1.1 Off-line Optimal Control Strategies

Open-loop or off-line optimal control of a fed-batch reactor generally involves calculating the substrate feeding profile that provides an optimal state or controlled variable trajectory by using a process model of the system. In addition, the resulting

feeding profile is implemented in an open-loop manner and feedback information during the process operation is not used during process operation. In general, the goal of the control strategy is to drive the controlled or state variables towards a desired value and therefore, the solution involved determining the time-varying substrate feeding profile that maximizes an objective function subject to process dynamic models is obtained.

In the past, optimal control problems can be solved by using the variational approach. In this approach, the optimal solution is obtained from the pontryagins's maximum principle which leads to the two point boundary value problem. However, the presence of many state constraints makes the solution via this approach quite difficult. Therefore, alternative approaches considering the optimal control problem as an optimization problem have been proposed, which can be classified into a simultaneous approach and a sequential approach. In the simultaneous approach, state and control variables are parameterized and the model solution and optimization problem are solved simultaneously. Renfro et al. (1987) proposed simultaneous optimization and solution procedure for systems described by differential algebraic equation using piecewise constant functions for independent variables that combines technologies of Quasi-Newton optimization algorithms and global spline collocation, and applied these algorithms to batch reactor control problem. In addition, Cuthrell and Biegler (1989) applied this approach based on an orthogonal collocation method to parameterize the state and control variables for solving the optimal control of penicillin production in a fed-batch reactor. In contrast to the simultaneous approach, only control variables are parameterized and also known as sequential approach. Following this approach, the model solution and optimization problem are solved sequentially. Banga et al. (1997) proposed the stochastic dynamic optimization algorithm for batch and fed-batch processes. These algorithms are based on a sequential control parameterization strategy. The original dynamic optimization problem is transformed into a constrained nonlinear programming problem using parameterization of the control function. The constrained nonlinear programming problem is then solved using stochastic algorithms such as Integrated Controlled Random Search for Dynamic Systems (ICRS/DS) and Adaptive Randomly Directed Search for Dynamic Systems (ARDS/DS). Such these algorithms were implemented to the production of penicillin and protein in a fed-batch bioreactor. Moreover,

Pushpavanam et al. (1999) considered the optimization of a fed-batch process for alcohol fermentation by a fixed final volume using a Sequential Quadratic Programming (SQP) approach. The entire batch is divided into a series of equally spaced intervals and inlet feed is assumed to be introduced in discrete pulses at the beginning of each interval. The effect of a number of stages on the optimum performance for both free and fixed time cases was reported and the results were compared with that obtained from the minimum principle. In addition, the off-line optimal control is applied to the multiple control variables problem (Kapadi and Gudi, 2004). They considered the problem of determining the optimal feeding profile of simultaneous saccharification and fermentation by using Differential Evolution (DE). The Differential Evolution with augmented Lagrangian including the dynamic penalty method was applied to fed-batch fermentation processes. They claimed that the performance of simultaneous saccharification and fermentation process in terms of the end lactic acid concentration and the productivity for single and multiple feed cases was improved substantially.

2.1.2 On-line Optimal Control Strategies

The limitation of an optimal control is the existence of unknown disturbances and model-plant mismatches which are particularly important in model-based control problems. Due to these conditions, the optimal feeding profile which obtained from the off-line optimal control may not give the optimal performance when applied to the actual process. To realize this fact, it is necessary to recalculate the optimal feeding profile as an on-line optimal control strategy. In the on-line optimal control strategy, the feedback information from the system is used to recalculate the new optimal feeding profile for the remaining batch stages.

A variety of different techniques have been used to solve the on-line optimal control problem in a fed-batch reactor. In 1987, Modak and Lim (1987) analyzed the problem of feedback optimization for the singular control problem of a fed-batch reactor. They considered a general system composed of four differential equations that represent mass balances for the state variables such as substrate, biomass, product, and reactor volume and then formulated the control problem based on singular control theory. In addition, Palanki et al. (1993) considered the problem of

determining an optimal profile in feedback form. They analyzed the optimal control problem from a geometric perspective and introduced the concept of degree of singularity that allows a better characterization of the necessary conditions for optimality. Optimal feedback laws are derived for the singular region of the operation. Such these algorithms were implemented to the yeast fermentation process and simulation results are presented for time-invariant systems and extended to include time-varying systems as well. Moreover, Seki et al. (2002) formulated the nonlinear model predictive control based on a successively linearized nonlinear model and applied it to fed-batch reactor process. These algorithms can prevent thermal runaway of the reactor temperature control. Furthermore, Mahadevan and Doyle III (2003) applied the input elimination based approach to the multiple feed processes. Shrinking horizon is applied with an on-line optimization for maximizing the chloramphenicol cetyltransferase production by optimized the inducer and glucose feed rates.

2.2 Artificial Neural Network

Artificial neural network is a mathematical structure that emerges from the attempt to simulate and understand the working of the human brain. This network involves with the learning process of the interesting system. After artificial neural network has learned what it needs to know, the trained network can be used to perform certain tasks depending on the particular application.

2.2.1 History of an Artificial Neural Network

At the beginning of the chronological overview of an artificial neural network, the work by McCulloch and Pitts essentially featured in the modern age of neural networks. The McCulloch and Pitts's neuron was very simple neuron which had a linear activation function with a threshold value to produce an output. The network was a two-layer network, and there were no training of these neurons. However, the McCulloch and Pitts's neuron model laid the foundation for future developments in neural networks. In 1949, Hebb proposed a learning process which was postulated from neurobiological viewpoint. During the training of network, the information was stored in the connections of the neurons and postulated a learning strategy for

adjustment of the connection weights. This was the first time that a learning rule was presented and allowed for adjustment of the synaptic weights. Later, Rosenblatt developed the original concept of perceptron and demonstrated that perceptron can generalize and learn. The perceptron consisted of neuron-like processing units with linear thresholds, and were arranged in layers similar to biological system. Hebbian learning rule was used for training a network. This rule reinforces active connections only-weight were increased when the outputs are active, and decreased when the outputs are inactive. In 1962, Widrow and Hoff developed the Adaline (adaptive linear element) which was trained by the Least Mean Square (LMS) learning, closely resembles Rosenblatt's perceptron. The Adaline used target value to calculate the prediction error and move the weight values in the direction of negative gradient of the error. Later, in 1974, Werbos proposed the backpropagation algorithm for training multilayer feedforward perceptron. In 1987, Carpenter and Grossberg developed self organizing neural networks based on Adaptive Resonance Theory (ART).

2.2.2 Applications of an Artificial Neural Network

Artificial neural networks are widely applied in modeling, identification and control of unknown nonlinear systems. The main advantage of the using of an artificial neural network is that a highly accurate mathematical model of the system is obtained without the detail of the system and artificial neural network can also deal with the multivariable systems.

Many authors applied the artificial neural network in various systems. Thibault et al. (1990) introduced the use of artificial neural network computational algorithms for dynamic modeling of bioprocesses. The performance of the artificial neural network is compared with an Extended Kalman filter (EKF) and was shown to exhibit comparable performance in case of a continuous stirred tank reactor. In 1991, Breusegem et al. (1991) applied an artificial neural network for on-line prediction of fermentation variables when kinetic change appears during the course of fermentation. They proposed an adaptive algorithm in which sliding window learning schemes are used. In 1992, Massimo et al. (1992) investigated the construction of artificial neural network-based biomass and penicillin estimators for use in industrial fermentations. Their results demonstrated that an artificial neural network could

capture the complex dynamic bioprocess. In addition, Psichogios and Ungar (1992) developed a hybrid model for a fed-batch bioreactor. The hybrid model combined a partial first principles model, which incorporated the available prior knowledge about the process being model, with an artificial neural network which served as an estimator of unmeasured process parameters that are difficult to model from first principles. The training method for the artificial neural network was the error backpropagation algorithm. They found that the hybrid model had better properties than standard black-box neural network model in that it is able to interpolate and extrapolate much more accurately. Furthermore, it was easier to analyze and interpret and required significantly fewer training examples. In 1995, Boslovic and Narendra (1995) applied both the conventional multilayer neural networks and radial basis function networks in an adaptive control scheme, which updates the unknown parameters online, for production of baker's yeast in a fed-batch fermentation process. They considered the set point regulation of the system under no-noise. They found that the conventional multilayer networks gave superior performance over the radial basis function networks and other nonlinear techniques such as the nonlinear adaptive and inverse dynamics controller. Moreover, Lightbody and lrwin (1995) developed a novel nonlinear model control strategy which utilized the nonlinear neural network model of the plant to act as a medium for the estimation of the parameters of the linear discrete-time model. This linear model is then utilized in conjunction with Kalman's method to design the inverse controller, wherein the parameters of this controller are adapted at each sample instant. They used this approach for set point tracking of concentration in a CSTR system, which outperformed the conventional PID control system. In addition, Ramchandran and Rhinehart (1995) used an artificial neural network inverse model to estimate the reflux and the holdup rate, which was incorporated in the Generic Model Control (GMC) strategy to control the top and bottom composition in a distillation column. The GMC technique basically involves incorporating the nonlinear process model directly in the formulation of the control algorithm. This was done for set point tracking and disturbance-rejection cases and the technique was found to be better than the PI controller with feed-forward features. In 1996, Lou and Perez (1996) used the backpropagation algorithm in conjunction with Kalman filtering in order to establish a new self-learning technique of Multi-layers Feed-Forward Neural network (MFFN). They found that this new technique was faster and more stable than the classical backpropagation algorithm for training

MFFN and it was less sensitive to the initial weights and to the learning parameters. In 1999, Lanouette et al. (1999) improved the modeling of complex processes when only small experimental data sets were available. Feed-forward and radial basis function neural networks were used in this problem. In addition, the influence of activation functions, the number of levels in stacked neural networks and the composition of training data set were studied. The study showed that the use of an artificial neural network was a powerful tool for modeling complex processes even when only a small set of data was available for training. A higher number of stacks led only to increase of the confidence level. However, radial basis function presented some weakness for modeling properly a process when data landscape lacked smoothness. Moreover, Shene et al. (1999) designed two different artificial neural networks to predict the state variable such as biomass, substrate and ethanol concentration of Z. mobilis CP4 in batch fermentations. The designed networks were black-box neural networks and the combination of artificial neural network and a mathematical model. Experimental data recorded from batch fermentations carried out under different condition were used to train the network and test its prediction. From the results presented, the error for utilizing the combination of an artificial neural network and a mathematical model was higher than black box neural network. Anyhow, the prediction of using both cases could be carried out using artificial neural networks. In 2000, Aziz et al. (2000) investigated the performance of different types of controllers in tracking the optimal temperature profiles in batch reactor. An artificial neural network was used as an online estimator to estimate the amount of heat release by chemical reaction within the GMC algorithm. The GMC controller coupled with an artificial neural network based heat release estimator was found to be more effective and robust than PI and PID controllers in tracking the optimal temperature profiles. Moreover, Aziz et al. (2003) used an artificial neural network inverse model based control (NN-IMBC) to track the optimal temperature profiles in complex exothermic batch reactor. It was also evaluated through a few robustness tests. Furthermore, the neural network estimator was embedded to strategy as the online estimator to estimate the amount of heat release by the chemical reaction. The NN-IMBC was found to be well performed in the tracking both set point and accommodating changes within its range of training. It also promised robust controller if it is trained with a wide range of reactor temperature covering all possible condition.

2.3 Control of Fed-Batch Reactors for Ethanol Production Processes

Ethanol is the one of important chemical substance which is utilized as a solvent in chemical process, main component in drinking alcohol and remedy and used either as a direct replacement for gasoline or blend with gasoline as an oxygenated compound. Moreover, ethanol is the important biological product which obtains from the fermentation process. Among the various modes of the ethanol fermentation such as continuous, batch and fed-batch modes, the fed-batch operation is the most preferable choice.

The general purpose of an ethanol production process in a fed-batch reactor is to maximize the amount of the desired ethanol product. For the control study of this process, Hong (1986) developed a mathematical model, in which the effects of substrate and product inhibitions are included, for describing the ethanol production. The control problem of maximizing the ethanol concentration in a fed-batch reactor was considered. Kelley's transformations were used to reduce the number of system equations and an analytical expression for the conjunction point between the singular and non-singular arcs for the singular control problem was also presented. The analytical expression enables the feeding policy along the singular arc to be derived in terms of the concentrations of the call mass, substrate, product and liquid volume in the reactor. Moreover, the on-off control for the ethanol production in a fed-batch reactor is suggested by Chen and Hwang (1990). The differential algebraic equation (DAE) of this system which represents the process model is simplified using Kelley's transformations and the input is parametrized over the batch period. Sequential Quadratic Programming (SQP) is used to solve the resulting NLP problem. The advantage of this approach is that it can easily to implement by low cost actuators and easily to maintenance the actuators. In addition, Luus (1993) developed the optimization techniques for the optimal substrate feeding policy of a fed-batch reactor which considered by Chen and Hwang. He presented the Iterative Dynamic Programming (IDP) using a penalty function to provide an easy way of solving a rather difficult optimal control problem. The advantage of this approach is that no transformations and auxiliary variables are required. He claimed that the optimum value for the performance index is 4% better than reported by Chen and Hwang.

Furthermore, Wang and Shyu (1997) developed an optimal feed policy of a fed-batch reactor for ethanol production by introducing additional inequality constraints in the optimization problem to assure optimal solution in a reality region. An updating rule of augmented Lagrange multipliers was introduced to handle inequality constraints so that Iterative Dynamic Programming could be used. In 2004, Soni and Parker (2004) considered the closed-loop control for ethanol production in a fed-batch reactor. The process model is characterized on both a macroscopic reactor scale and a microscopic cellular scale. Nonlinear optimization techniques are used to generate an optimal substrate feed profile for the nominal problem that the ethanol concentration at the end of the batch is maximized. Shrinking-horizon nonlinear quadratic dynamic matrix control is used for closed-loop trajectory tracking and disturbance rejection. Moreover, Xiong and Zhang (2004) considered the modeling and optimal control of a fed-batch process by using a novel control affine feedforward neural network (CAFNN) for a constrained nonlinear optimal control problem. The control affine feedforward neural network offers an effective and simple optimal control strategy by using a sequential quadratic programming (SQP) where the gradient information is computed directly from CAFNN. In addition, Xiong and Zhang (2005) considered the on-line re-optimization of an ethanol production in fed-batch processes by using the neural network discrete-time models which are used to model the fed-batch processes. A modified iterative dynamic programming algorithm based on discrete-time nonlinear model is developed to solve the on-line optimization problem. They claimed that this approach is very effective in addressing the problem of model-plant mismatches.

CHAPTER III

THEORETICAL BACKGROUND

In this chapter, the theoretical background of the optimal control, the on-line optimal control, the artificial neural network and the biological processes are described.

3.1 Optimal Control

Optimal control, also known as a dynamic optimization problem, involves determining a control profile for a dynamic system that optimizes a given performance index. The dynamic system is usually represented by sets of differential and algebraic equations (DAEs) derived from dynamic mass and energy balances, and physical and thermodynamic relations.

A general dynamic optimization problem can be stated as follows:

Find $u(t)$ over $t \in [t_0 , t_f]$ maximizing or minimizing

$$J = \theta[x(t_f)] + \int_{t_0}^{t_f} \phi[x(t),u(t),t]dt \tag{3.1}$$

subject to

$$\frac{dx}{dt} = f[x(t),u(t),t] \tag{3.2}$$

$$x(t_0) = x_0 \tag{3.3}$$

$$h[x(t),u(t),t] = 0 \tag{3.4}$$

$$g[x(t),u(t),t] \leq 0 \tag{3.5}$$

$$x^L \leq x(t) \leq x^U \tag{3.6}$$

$$u^L \leq u(t) \leq u^u \tag{3.7}$$

where J is the performance index or desired objective function, x and u are the vector of state and control variables, respectively, Eq. (3.2) is the system of ordinary differential equations, Eq. (3.3) is the initial condition for Eq. (3.2), Eqs. (3.4) and (3.5) are the equality and inequality algebraic constraints, respectively, and Eqs. (3.6) and (3.7) are the upper and lower bounds on the state and control variables, respectively.

The solution of optimal control problems have been a subject of research for many years. There are several different computational techniques available for solving the optimal control problems. The indirect methods focus on obtaining a solution to the classical necessary conditions for optimality. These methods are also known as variational methods. However, it has been found that these methods result to a two-point boundary value problem which is difficult to solve. Thus, the direct methods which transform the original optimal control problem into a finite-dimensional nonlinear programming problem and solve it directly are proposed. Depending on the degree of discretization, the direct methods can be classified into two general strategies. In the simultaneous methods, the control and state variables are discretized (full discretization) whereas only the control variables are discretized (partial discretization) in the sequential methods.

3.1.1 Variational Methods

These methods are based on the solution of the first order necessary conditions for optimality that are obtained from Pontryagins's Maximum Principle (PMP). According to PMP, the problems of minimizing the objective function J in Eq. (3.1) subject to dynamic constraints represented by Eqs. (3.2) - (3.5) can be reformulated as that of minimizing the Hamiltonian function. These procedures lead to a two-point boundary value problem (TPBVP) that can be solved with different approaches, including single shooting, multiple shooting, invariant embedding or some discretization methods such as collocation on finite elements or finite differences. The limitation of these methods is the complexity in the solution of differential-algebraic equations.

3.1.2 Simultaneous Methods

In the simultaneous methods, both state and control variable profiles are discretized by approximating functions and treated as decision variables in the optimization problem. The process dynamic models and the optimization problems are solved at the same time. These avoid solving the model equations at each of iteration in the optimization algorithm as in the sequential methods. In this approach, the dynamic process model constraints in the optimal control problems are transformed to a set of algebraic equations which is treated as equality constraints in the NLP problem. As a result, the optimal control problems are reduced to a constrained nonlinear optimization problem. To solve this problem, Successive Quadratic Programming (SQP), reduced space SQP, the interior-point approach and the conjugate gradient methods can be used to solve efficiently.

The general algorithm for the simultaneous methods is as follows:

Problem:

$$\min_{x(t),u(t)} \Phi[x(t),u(t),t] \qquad \text{Objective function}$$

s.t.

$$\begin{aligned}
&\frac{dx}{dt} = f[x(t),u(t),t] && \text{Process dynamic equations} \\
&x(0) = x_0 && \text{Initial conditions for states} \\
&h[x(t),u(t),t] = 0 && \text{Equality constraints} \\
&g[x(t),u(t),t] \le 0 && \text{Inequality constraints} \\
&x^L \le x(t) \le x^U && \text{State profile bounds} \\
&u^L \le u(t) \le u^u && \text{Control profile bounds}
\end{aligned} \qquad (3.8)$$

Step 1: Discretize the process states and inputs using any standard collocation method (e.g. orthogonal collocation).

$$x_{K+1}(t) = \sum_{i=0}^{K} x_i \phi_i(t) \qquad \text{where} \quad \phi_i(t) = \prod_{k=0,i}^{K} \frac{(t-t_k)}{(t_i-t_k)} \;,\; x_{K+1}(t_i) = x_i$$

$$u_K(t) = \prod_{i=1}^{k} u_i \psi_i(t) \qquad \text{where} \quad \psi_i(t) = \prod_{k=1,i}^{k} \frac{(t-t_k)}{(t_i-t_k)} \;,\; u_K(t_i) = u_i \tag{3.9}$$

Step 2: Substitute the discrete states and inputs into process dynamic model and obtain the algebraic expression for residuals.

$$\mathrm{R}(t_i) = \sum_{j=0}^{K} x_j \dot{\phi}_j(t_i) - f(x_i, u_i, t_i) = 0$$

$$i = 1, \ldots, K \tag{3.10}$$

$$\text{with } x(0) = x_0$$

Step 3: Substitute the discretized dynamic model into the original optimal control problem.

$$\min_{x(t),u(t)} \Phi[x(t), u(t), t]$$

s.t.

$$\mathrm{R}(t_i) = \sum_{j=0}^{K} x_j \dot{\phi}_j(t_i) - f(x_i, u_i, t_i) = 0 \qquad \text{when} \quad i = 1, \ldots, K$$

$$x(0) = x_0 \tag{3.11}$$

$$h[x(t), u(t), t] = 0$$

$$g[x(t), u(t), t] \le 0$$

$$x^L \le x(t) \le x^U$$

$$u^L \le u(t) \le u^u$$

Step 4: Choose t_i by using orthogonal collocation method.

Step 5: Solve problem given in Step 3 at the t_i chosen in Step 4 using any non-linear programming problem solver such as Successive Quadratic Programming (SQP).

The main advantage of the simultaneous approaches is a capability in handing constraints on state variables. This is because these constraints can be dealt with by including them directly in the optimization problem as additional constraints. However, owing to the discretization on both state and control variables, these lead the simultaneous methods to a large scale optimization problem consisting of a large set of algebraic constraints and decision variables.

3.1.3 Sequential Methods

In the sequential methods, only the control variables are discretized. These techniques are also known as control vector parameterization methods. Typically, a piecewise constant approximation over equally spaced time intervals is made for the inputs. Given the initial conditions and a given set of control parameters, the process model equations are solved with a differential-algebraic equation solver at each of iteration. This produces the value of the objective function, which is used by a nonlinear programming solver to find the optimal parameters in the control parameterization.

The general algorithm of the sequential methods is as follows:

Problem:

$$\underset{u(t)}{\text{Min}}\ \Phi[x(t),u(t),t] \qquad \text{Objective function}$$

s.t.

$$\begin{aligned}
&\frac{dx}{dt} = f[x(t),u(t),t] && \text{Process dynamic equations}\\
&x(0) = x_0 && \text{Initial conditions for states}\\
&h[x(t),u(t),t] = 0 && \text{Equality constraints}\\
&g[x(t),u(t),t] \le 0 && \text{Inequality constraints}\\
&x^L \le x(t) \le x^U && \text{State profile bounds}\\
&u^L \le u(t) \le u^u && \text{Control profile bounds}
\end{aligned} \qquad (3.12)$$

Step 1: Discretize the process inputs using any standard collocation method (e.g. orthogonal collocation).

$$u_K(t) = \prod_{i=1}^{k} u_i \psi_i(t) \quad \text{where} \quad \psi_i(t) = \prod_{k=1,i}^{k} \frac{(t - t_k)}{(t_i - t_k)} \;,\; u_K(t_i) = u_i \tag{3.13}$$

Step 2: Substitute the parameterized inputs into the process dynamic model

$$\begin{aligned} &\frac{dx}{dt} = f[x(t), u_K(t), t] \\ &i = 1, ..., K \\ &\text{with } x(0) = x_0 \end{aligned} \tag{3.14}$$

Step 3: Substitute the modified process dynamic model given by Eq. (3.14) into the problem given by Eq. (3.12). The updated problem statement according to sequential methods is given by Eq. (3.15).

$$\underset{u(t)}{\text{Min}} \; \Phi[x(t), u(t), t]$$

s.t.

$$\begin{aligned} &\frac{dx}{dt} = f[x(t), u_K(t), t] \quad \text{when} \quad i = 1, ..., K \\ &u_K(t) = \prod_{i=1}^{k} u_i \psi_i(t) \quad \text{where} \quad \psi_i(t) = \prod_{k=1,i}^{k} \frac{(t - t_k)}{(t_i - t_k)} \;,\; u_K(t_i) = u_i \\ &x(0) = x_0 \\ &h[x(t), u(t), t] = 0 \\ &g[x(t), u(t), t] \le 0 \\ &x^L \le x(t) \le x^U \\ &u^L \le u(t) \le u^u \end{aligned} \tag{3.15}$$

Step 4: Choose t_i using orthogonal collocation method and evaluate u as a function of time by using Eq. (3.13).

Step 5: Choose initial guess for decision variables and solve the dynamic process model given by Eq. (3.14) in Step 2 for the input obtained in step 4 using any ODE solver.

Step 6: Evaluate the objective function given in Eq. (3.15) using state and control profiles obtained in step 5 and update the values of decision variables using any standard optimization routine such as steepest descent or Quasi- Newton methods. Repeat steps 4 through 6 until convergence.

The main advantage of the sequential methods is that only the control profiles are discretized and considered as the decision variables. The optimization formulated by this approach is a small scale nonlinear programming problem. However, the limitation of these methods is a difficulty to handle a constraint on state variables (path constraints). This is because the state variables are not directly included in the nonlinear programming problem.

3.2 On-line Optimal Control

Due to the presence of unknown disturbances and model-plant mismatches, the off-line calculated optimal control profile may not be optimal when applied to the actual process. If this optimal control profile is still used for the remaining batch stages, the process output at end of the batch will be significantly different from the desired output. As a result, there is a need to re-optimize the control profile for the remaining batch stages. This issue can be overcome by on-line updating the optimal control profile; on-line feedback information obtained from the earlier stage is used to re-calculate the new control profile for the remaining batch stage.

The basic concept of the on-line optimal control is illustrated in Figure 3.1. The batch length is divided into N stage and the control profile is assumed to be constant during each stage. At each current moment k, the process output $y(k+j)$ is predicted over a finite time horizon $j = 1,....,N$. The predicted output values at time k are indicated by $y(k+j|k)$ and the value N is called the prediction horizon. A reference trajectory is defined over the prediction horizon, describing how we want to guide the

process output so as to minimize the tracking error $e(k+j|k) = y_{ref}(k+j|k) - y(k+j|k)$. The control profile is calculated on the basis of a measurement in order to minimize a specified objective function, depending on the predicted output errors. The first element $u(k|k)$ of the optimal control profile is applied to the actual process. All other elements of the calculated control profiles can be forgotten, because at the next sampling instant all time-sequences are shifted, a new output measurement $y(k+1)$ is obtained and the whole procedure is repeated. This leads to a new control input $u(k+1|k+1)$, which is generally different from the previously calculated $u(k+1|k)$.

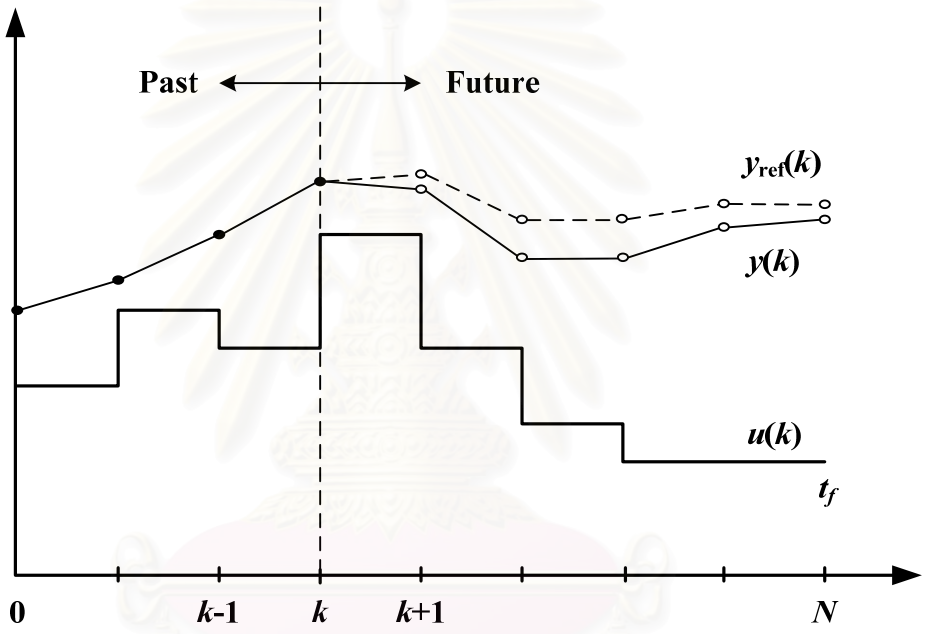

Figure 3.1 On-line optimal control scheme

3.3 Artificial Neural Network

Artificial neural networks are mathematical structures which built from the attempt to emulate the human brain or biological network. These networks involve with the learning process of interesting systems. After artificial neural networks have learned, the trained network can be used to perform certain tasks depending on the particular application. In addition, the artificial neural networks have the ability to learn from their environment and adapt it in an interactive manner similar to the biological counterparts.

3.3.1 Introduction of an Artificial Neural Network

Since the artificial neural network paradigm emerged from the attempt to emulate and understand the working of the human brain or biological neural network. In the nervous system, the brain is the central element of the nervous system which is connected to receptors that shuttle sensory information to it, and delivers action commands to effectors. In addition, the brain is a huge and complicated neural network which consists of about 10^{11} neurons. Each neuron consists of three main components: dendrites, cell body and axon (as shown in Figure 3.2). Dendrites which are branchlike nerve fiber around the neural cell body receive signals from other neurons by the receiving zones, called synapse. The cell body or soma sums the incoming signals which are received from dendrites and sends them to an axon. Axon which is a long fiber-like extension from cell body is the transmit channel of impulses to the other neurons.

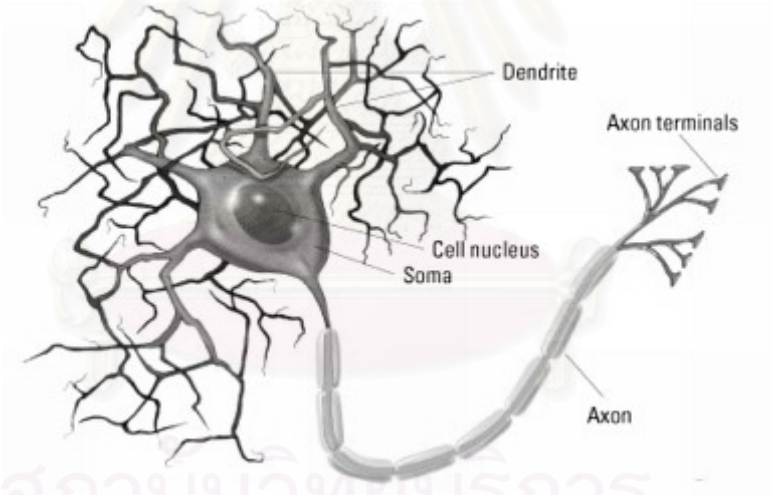

Figure 3.2 Components of biological neural network

The mentioned basic concept of biological neural network lead to research in the area of the mechanism and model of human brain including develop the model to solve complex problems in science and engineering. The first artificial neuron was created in 1943 by McCulloch and Pits. They proposed the model of a simple neuron which seemed appropriate for modeling symbolic logic and its behavior. The McCulloch-Pitts neuron is a simple unit having a linear activation function with threshold value to produce an output. In 1959, Rosenblatt began work on the

perceptron which consisted of neuron-like processing units with linear thresholds, and were arranged in layers similar to biological systems. The perceptron can learn and compute a weighted sum of the inputs, subtract a threshold, and pass one of two possible values out as the result. In addition, Widrow and Hoff developed the models which are called MADALINE (Multiple Adaptive Linear). MADALINE was the first artificial neural network to be applied to a real world problem.

3.3.2 Components of an Artificial Neural Network

The artificial neural network consists of many interconnected artificial neurons or nodes. In each node, there are many components that are used to build an artificial neural network. These components are described as the follow.

3.3.2.1 Weighting Factors

Weighting factors are adaptive coefficients within the artificial neural network that determine the intensity of the input signal as registered by the artificial neuron. An artificial neuron usually receives many inputs for create the network. Each input has its own relative weight which impacts the input on the summation function. Some inputs are made more important than others so that they have a greater effect on the processing element as they combine to produce a neural response. Thus, their weighting factors are greater than the others. These weighting factors can be modified in response to various training sets and according to a network's specific topology**.**

3.3.2.2 Summation Function or Basis Function

The first step in the operation is to compute the sum of all inputs. Mathematically, the inputs and the corresponding weights are vectors which can be represented as (x_1, x_2,…, x_n) and (w_1, w_2,…, w_n). The simplistic summation function is found by multiplying each component of the x vector by the corresponding component of the w vector and then adding up all the products. Moreover, the summation function can be more complex than the simplistic summation function. The inputs and the weighting factors can be combined in many different ways before passing to the transfer function.

The summation function can divided into two common forms:

- Linear Basis Function (LBF)

Linear basis function is a hyperplane-type function which is a first-order basis function. The net value is a linear combination of the inputs and the weighting factors which is shown as the follow.

$$sum = \sum_{j=1}^{n} w_{ij} x_j \tag{3.16}$$

- Radial Basis Function (RBF)

Radial basis function is a hypersphere-type function which involves with the second-order (nonlinear) basis function. The net value which represents the distance to a reference pattern is shown as the follow.

$$sum = \sqrt{\sum_{j=1}^{n} (x_i - w_{ij})^2} \tag{3.17}$$

3.3.2.3 Transfer Function or Activation Function

The results from the summation function, almost always the weighted sum, are transformed to a working output by the transfer function. In the transfer function, the total summation of the inputs and the weighting factors can be compared with some threshold to determine the neural network output. If the summation is greater than the threshold value, the processing element generates a signal. If the summation is less than the threshold, no signal is generated from the transfer function.

The transfer function is generally non-linear function. Linear functions are not useful because the linear transfer functions are limited such as the output is simply proportional to the input. For example, the most common transfer functions which are

the Step function, Ramp function, Linear function, Log-Sigmoid function, Tangent-Sigmoid function, Gaussian function and Arc tangent function are shown in Table 3.1.

Table 3.1 Transfer functions of an artificial neural network

Function	Equation	Characteristic
Step function	$f(x)=\begin{cases}1 & if \quad x>0\\ 0 & otherwise\end{cases}$	Linear
Ramp function	$f(x)=\begin{cases}1 & if \quad x\geq 1\\ x & if \quad x<1\\ -1 & if \quad x<-1\end{cases}$	Linear
Linear function	$f(x)=x$	Linear
Log-Sigmoid function	$f(x)=\frac{1}{1+e^{-x}}$	Non-linear
Tangent-Sigmoid function	$f(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$	Non-linear
Gaussian function	$f(x)=ce^{-x^2}$	Non-linear
Arc tangent function	$f(x)=\arctan(\frac{x}{3.1416})+0.5$	Non-linear

3.3.2.4 Scaling

In the neural networks training, the networks can be made more efficient if scaling processing steps are carried out on the input pattern and target. For example, the backpropagation algorithm is used to train a feed-forward perceptron, if a sigmoid function is used as a non-linear activation function, the saturation limit are 0 and 1. If the training patterns have large values compared to these limits, the non-linear activation functions could be operating almost exclusively in a saturated mode and not allow the network to train. Therefore, the training data should be range-scale to avoid this problem.

3.3.2.5 Output Function

Each processing element or neuron allows one output signal which it may be a output to hundreds of inputs from other neurons. This is just like the biological neurons which there are many inputs and only one output action. Normally, the output is directly equivalent to the transfer function's result.

3.3.2.6 Error Function

In the training of supervised networks, the training procedure requires a measure of the difference between the neural network output values and the target (desired output) values. The difference between the target and output values is so called the error. This error is transformed by the error function to match particular network architecture. For example, the most common error functions are sum square error, mean square errors and mean absolute error.

These error functions are described as the follow.

- Sum Square Error

$$SSE = \sum_{i=1}^{N} (y_i - p_i)^2 \tag{3.18}$$

- Mean Square Error

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (y_i - p_i)^2 \tag{3.19}$$

- Mean Absolute Error

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - p_i| \tag{3.20}$$

which y_i is the network output

p_i is the network target

3.3.2.7 Learning Function

The purpose of the learning function is to modify the variable connection weights on the inputs of each processing element according to some neural based algorithm to achieve a desired result. There are two types of learning algorithm; supervised and unsupervised learning. Supervised learning requires a teacher which may be a training set of data or an observer who grades the performance of the network results. For unsupervised learning, the system must organize itself by some internal criteria designed into the network.

3.3.3 Architecture of an Artificial Neural Network

3.3.3.1 Network Structures

Artificial neural network structure can be divided into common types such as feed-forward networks and feed-back networks.

- Feed-Forward Networks

Feed-forward networks allow the signals travel from input to output one way only. There is no feedback in the network such as the output of any layer does not affect in the same layer. Feed-forward networks tend to be straight forward networks that associate inputs with outputs.

- Feed-Back Networks

Feed-back networks which allow the signals travel in both directions of the network are very powerful networks. These networks are dynamic which their states change continuously until they reach an equilibrium point. They remain at the equilibrium point until the inputs change and a new equilibrium need to be found.

3.3.3.2 Connection Structures

An artificial neural network comprises the neuron and weight building blocks. The behavior of the network depends on the interaction between these building blocks. There are four common types of connections such as feed-forward, feed-back, lateral and time-delayed connections.

- Feed-Forward Connections

  For all the neural network models, the data from neurons of a lower layer are propagated forward to neurons of an upper layer via feed-forward connection networks.

- Feed-Back Connections

  For all the neural network models, the feed-back connections bring the data from neurons of an upper layer back to neurons of a lower layer.

- Lateral Connections

  For all the neural network models, the lateral connections allow the neurons to interact in the same layer.

- Time-Delayed Connections

  Delay elements may be incorporated into the connections to yield temporal dynamics models. They are more suitable for temporal pattern recognitions.

3.3.3.3 Network Layers

The layers of an artificial neural network are divided into three types such as input layer, hidden layer and output layer. The input layer represents the raw

information that is fed into the network. The hidden layer is between the input and output layer. The output layer is the last layer of the networks that depends on the activity of the hidden layers and the weights between the hidden and output layers.

3.3.4 Learning Algorithm of an Artificial Neural Network

Training or Learning means the modifying values of the weighting factor in the interconnections to achieve some target criteria for the output layer. Information is stored and distributed throughout the network via the interconnection weights. Many of learning algorithms are proposed and divided into two types which are shown in Table 3.2.

Table 3.2 Learning algorithms of an artificial neural network

Learning algorithm	
Supervised learning	Unsupervised learning
Perceptron	Additive Grossberg (AG)
Adaline	Adaptive Resonance Theory (ART)
Backpropagation	Continuous Hopfield (CH)
Boltzman Machine (BM)	Learning Matrix (LM)
Associate Reward Penalty (ARP)	Learning Vector Quantizer (LVQ)

3.3.4.1 Supervised Learning

In the supervised learning, training process consists of the input and output data. This data is often referred to as the training set. During the training, the actual output of an artificial neural network is compared to the desired output. Weighting factors, which are usually randomly set to begin, are then adjusted by the network. Thus, the network will produce a closer match between the desired and the actual output in the next iteration. The learning algorithm tries to minimize the current errors of all processing elements. This global error reduction is created over time by continuously modifying the input weights until acceptable network accuracy is reached.

3.3.4.2 Unsupervised Learning

Unsupervised learning which is sometimes called self-supervised learning is limited to networks known as self-organizing maps. These kinds of networks are not in widespread use. These networks use no external influences to adjust their weights. Instead, they internally monitor their performance. These networks look for regularities or trends in the input signals, and makes adaptations according to the function of the network. Even without being told whether it's right or wrong, the network still must have some information about how to organize itself. This information is built into the network topology and learning rules.

3.3.5 Multilayer Feed-Forward Neural Network

Multilayer feed-forward neural network is a one of the most popular artificial neural network architectures which is widely used in the function approximation or modeling any arbitrary system. This type of network is also sometimes called the multilayer perceptron because of its similarity to perceptron networks with more than one layer. Multilayer feed-forward neural network consists of an input layer, one or more internal layers and an output layer. The internal layers are called hidden layers because they only receive internal inputs (inputs from other processing units) and produce internal outputs (outputs to other processing units). The structure of multilayer feed-forward neural network is shown in Figure 3.3.

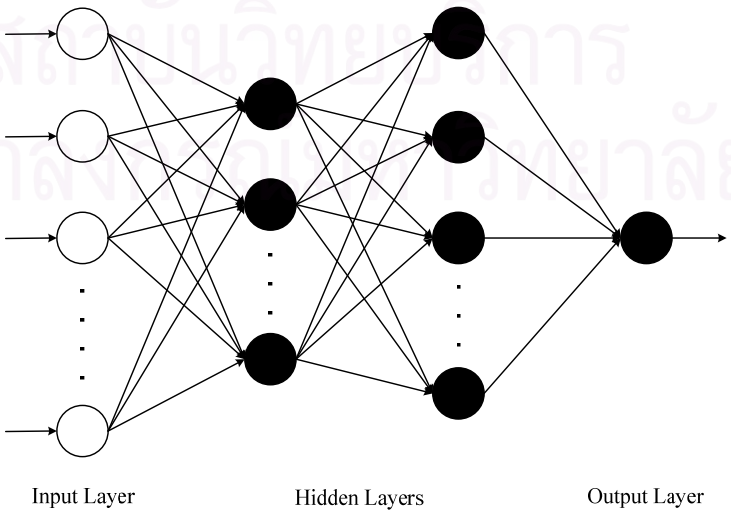

Figure 3.3 The structure of multilayer feed-forward neural network

3.3.6 Design of Artificial Neural Network

3.3.6.1 Structure and Size of Network

No standard procedure has been known to determine the structure and the number of neurons or nodes in the network for any particular application. However, the general procedure for selecting the hidden nodes is to fix an initial size and then check the error tolerance of this structure. If this error satisfies, the training process is stopped. If not, the size and the structure are revised and the whole procedure repeats until it satisfies the tolerance.

3.3.6.2 Data Collection

In utilizing of an artificial neural network, the data set collection is normally split into various sets. One is the training set which is used to train the network weights and normally span the operating region of the model. Later is the testing data set which is used for final validation of the trained network.

The selection of inputs data which is fed into the networks is an important consideration for any particular application. For steady state application, the selection of inputs to the networks basically depends on the relevant variables likely to have an effect on the predicted output variable. For modeling the dynamic behavior of a system, it would not only depend on these relevant variables but also the time history of these variables as well as the time history of the output variables. The knowledge of the system such as the model order is use as the initial guide to decide on the time history.

3.3.6.3 Data Processing

After the data collection, all data should be pre-processed using statistical procedure. Data in the training sets are pre-processed to have zero mean and unit variance. This is necessary to prevent input with large average values in certain dimension.

3.3.6.4 Weight Initialization

The initial weight specification has an effect on the speed and quality of neural network training. The small random number is normally used to initialize the weights of the network so that each connection responds slightly differ during training. If the final prediction does not satisfy the error tolerance during training, the weights are also re-initialized and the identification process is repeated.

3.3.6.5 Training the Network

Training is a procedure to determine the optimal values of the connection weights and bias weights. Training begins by initially assigning arbitrary small random values to the weights. Training proceeds iteratively until a satisfactory model is obtained. In each of iteration, called an epoch, the actual outputs corresponding to all the sets of inputs in the training set are predicted and the weights are adjusted in the direction of the output prediction error is decreased. The weights are incrementally adjusted for every pattern in every of iteration and they gradually converge on the optimal values.

Different network architectures require the different training or learning algorithms. The training times can be significantly reduced by the use of suitable training algorithms. However, backpropagation algorithm remains the mainstay of performing neural network learning.

3.3.6.6 Model Validation

Over-learning, which occurs when the network starts to learn the presented pattern in a point-wise fashion instead of learning the functionality, is a potential problem that can easily occur in process identification. During over-learning, the performance of the network training continues to improve on the learning data set but starts to degrade on the testing set. However, it can be dealt with by proper training and validation.

3.3.6.7 Basic Steps of an Artificial Neural Network Design

There are many procedures of the artificial neural network design but the basic steps of the artificial neural design are summarized as the follow.

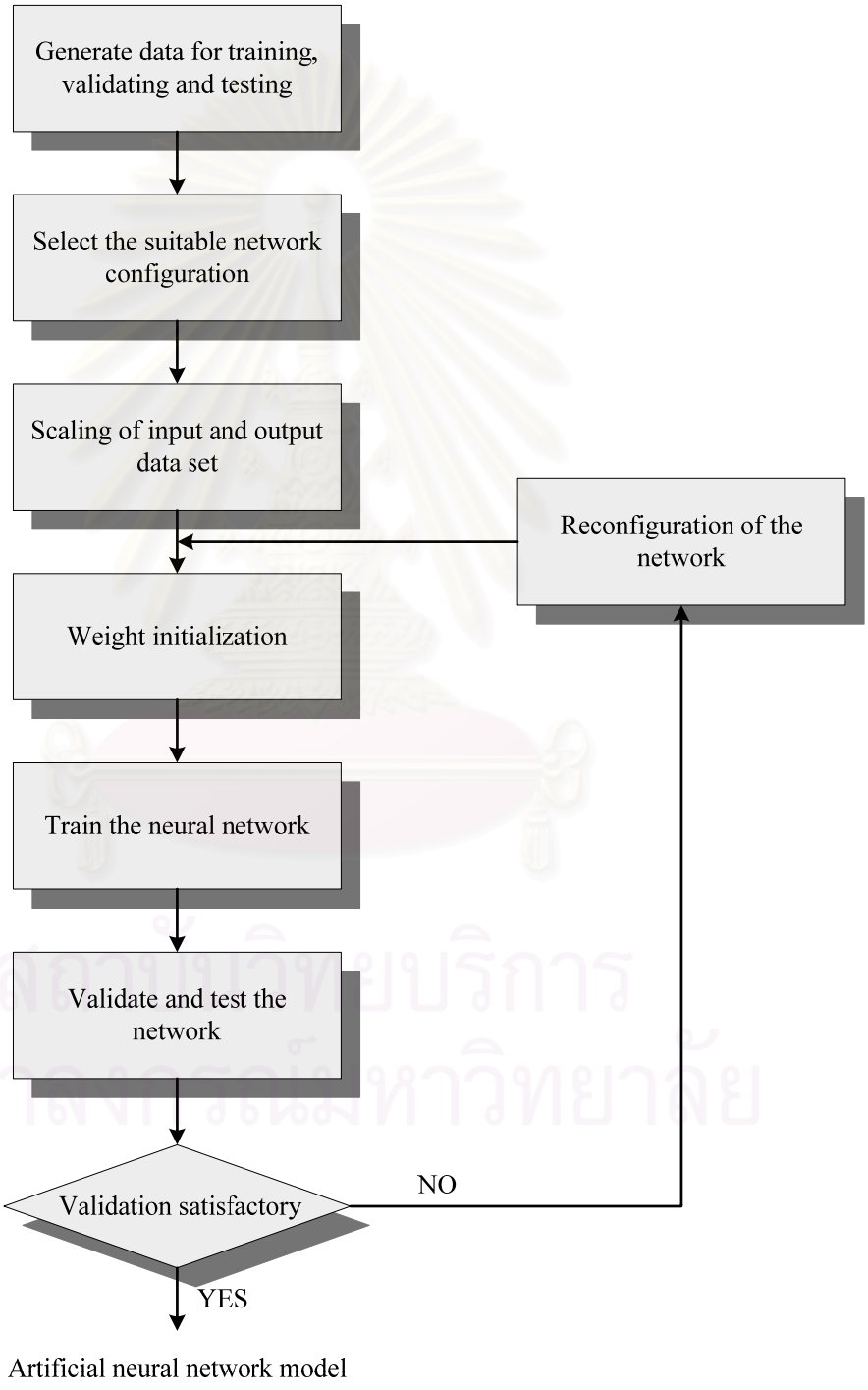

Figure 3.4 Basic steps of an artificial neural network design

3.3.7 Application of Artificial Neural Network

The artificial neural network is widely applied in modeling of the unknown nonlinear systems. The main advantage of the use of artificial neural network is obtaining a highly accurate mathematical model of the system without the detail of the system. The process modeling applications use the artificial neural network to approximate the relationship between the input and output variables. During the process modeling, a number of candidate models are considered and only one model, which is expected to the best prediction of the process outputs with the given process inputs, is selected. The selected model is the one that is expected to have the least prediction error in the future. In addition, artificial neural network is also the universal function approximator that typically works better than the traditional function approximation method for the application of any arbitrary system.

3.4 Biological Processes

In biological processes or bioprocesses, biological systems such as bacteria, yeast, fungi, algae or also animal cells, plant cells or isolated enzymes, are used to convert supplied substrates to desired products. These products can be the organism itself or chemical substances which are utilized in any application.

3.4.1 Bioreactors

For many applications, it is necessary to run the bioprocesses in the well controlled and closed reactors which are widely called fermentor or bioreactor. Several types of bioreactors exist to meet the different requirements of different organisms and products. Some of the most common types are the stirred tank, the bubble column, the airlift, and the packed bed reactor.

The stirred tank reactor is usually a cylindrical vessel equipped with a mechanical impeller, baffles and aeration at the bottom of the vessel. Meanwhile, the bubble column reactor is cylindrical vessel which does not have a mechanical stirrer. Mixing occurs due to the gas flow coming from a sparger at the bottom of the column. The airlift reactor is a bioreactor which also uses the gas flow from a sparger at the

bottom of the reactor to introduce mixing, but has two interconnected parts instead of one cylinder. The supplied gas of the airlift reactor rises in one compartment, the riser, and due to the different density caused by the gas, the medium circulates going down in the other compartment, the downcomer, and rising in the riser. Finally, the packed bed reactor is a bioreactor which filled with solid particles containing the biocatalysts and medium flows past these particles.

3.4.2 Modes of Operation

The bioprocesses can be operated in various operations such as batch, fed-batch, or continuous mode.

In a batch mode, all substrates are available in the initial medium and none is supplied extra during the process operation. The process operation is stopped after the substrate is converted.

In a fed-batch mode, substrates are fed into the reactor during the process operation until the end of the operation. This type of process operation is used very often since it allows relatively high biomass or product concentrations to be achieved. These may not be achievable in batch mode because the total amount of needed substrate would be so much that the initial substrate concentration would be strongly inhibiting the process.

In a continuous mode, substrate is fed continuously and product is also achieved continuously. In the common type of continuous mode with growing cells, substrate is fed into the bioreactor at a constant rate and the suspension is removed at the same rate This way the culture will run into a steady state where the growth of the organism equals the dilution by the feeding.

In the current work, the stirred tank reactor with the fed-batch operation mode is studied for the ethanol production process.

3.4.3 Model of Fed-Batch Processes

3.4.3.1 Mass Balance

The macroscopic models of bioprocesses are based on mass balances:

$$Accumulation = Conversion + Transport \tag{3.21}$$

So, assuming ideal mixing, we can write for the bioreactor:

$$(\dot{CV}) = rV + \sum_f F_f C_f \tag{3.22}$$

where dots above the symbols denote the time derivatives, r are the conversion rates per unit volume and F are fluxes of feeds into and out of the bioreactor, with concentrations C_f in these feeds.

The total mass of the compounds in the fermentation suspension is balanced. Besides the concentration of the compounds, the volume of the suspension has to be regarded. This is especially important for fed-batch fermentations where the volume generally changes significantly during the process.

$$(\dot{CV}) = \dot{C}V + \dot{V}C \tag{3.23}$$

Then, the balance in Eq. (3.22) can be rewritten as

$$\dot{C} = r + \sum_f \frac{F_f C_f}{V} - \sum_f \frac{F_f C}{V} \tag{3.24}$$

with

$$\dot{V} = \sum_f F_f \tag{3.25}$$

3.4.3.2 Kinetics

The kinetic parameter of the above equation is the conversion rates *r* which is a function of the concentrations for all relevant substances: biomass, substrate and product. One common type of kinetic equations which is probably the most often used equation to describe biomass growth is the Monod kinetics.

For the Monod kinetics, which is based on the Michaelis-Menten type enzyme kinetics (Michaelis and Menten, 1913) is described as the follow.

$$r = \frac{v_{\max} C_s C_c}{K_m + C_s} \tag{3.26}$$

where C_s is the concentration of substrate, C_c is the catalyst concentration and K_m is the Michaelis-Menten constant, which is a binding constant, equal to the concentration needed to achieve an enzyme specific conversion rate of half the maximal specific conversion rate $v_{\max}$. For more detailed explanation of the theoretical background of this equation and similar enzyme kinetic models, see for instance Biselli (1992) and Roels (1983).

For biomass growth on one limiting substrate, the Monod kinetics can be written analogously as:

$$r_x = \mu_{\max} \frac{C_s C_x}{K_s + C_s} \tag{3.27}$$

$$r_s = - \frac{r_x}{Y_{xs}} \tag{3.28}$$

where C_s is the substrate concentration, C_x is the biomass concentration, μ_{max} is the maximal specific growth rate, K_s is the Monod-constant analogous to the Michaelis-Menten constant and Y_{xs} is the yield of biomass production on substrate, which is the amount of biomass which is produced from one unit of substrate.

The Monod equation is probably the most often used equation to describe biomass growth. Some alternative equations have also been proposed, amongst which the Blackman equation (Dabes et al., 1973), the equations by Konak (1974) and a logistic equation (Bona and Moser (1997).

CHAPTER IV

APPLICATION OF OPTIMAL CONTROL FOR ETHANOL PRODUCTION IN A FED-BATCH REACTOR

In this chapter, the mathematical model of ethanol production process in a fed-batch reactor is presented. Moreover, the implementation of the optimal control strategy for controlling the ethanol production process in a fed-batch reactor is studied.

4.1 Mathematical Model of Ethanol Production Process in a Fed-Batch Reactor

Ethanol is the one of important chemical substance which obtains from the fermentation process. Among the various modes of the fermentation such as continuous, batch and fed-batch modes, the fed-batch mode is the most preferable alternative to implement. The general purpose of an ethanol production process in a fed-batch reactor is to maximize the amount of the desired ethanol product at the end of the operation. For the control study of this process, Hong (1986) is the first who developed a mathematical model, in which the effects of substrate and product inhibitions are included, for describing the ethanol production by *Saccharomyces cerevisiae* in the fed-batch culture.

In the present study, the mathematical model developed by Hong (1986) for describing the ethanol fermentation in a fed-batch reactor is used in the simulation studies. The schematic diagram of the fed-batch reactor is illustrated in Figure 4.1. The mathematical model of the ethanol production which consists of differential and algebraic equations is given as follows.

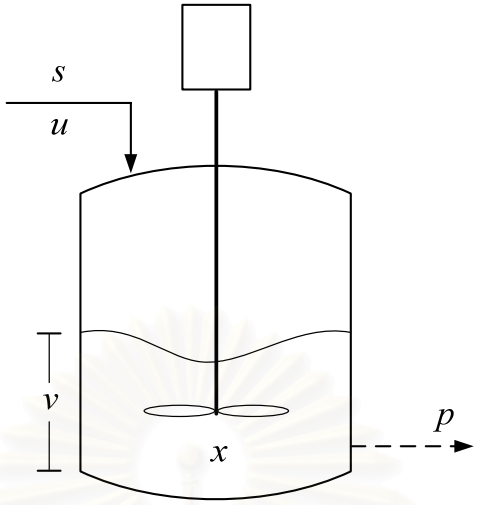

Figure 4.1 Schematic diagram of the fed-batch reactor

$$\frac{dx}{dt} = \mu x - \frac{x}{v}u \tag{4.1}$$

$$\frac{ds}{dt} = -\frac{\mu x}{Y} + \frac{(s_0 - s)}{v}u \tag{4.2}$$

$$\frac{dp}{dt} = \eta x - \frac{p}{v}u \tag{4.3}$$

$$\frac{dv}{dt} = u \tag{4.4}$$

$$\mu = \frac{\mu_0 s}{(1 + \frac{p}{K_p})(K_s + s)} \tag{4.5}$$

$$\eta = \frac{\eta_0 s}{(1 + \frac{p}{K_p'})(K_s' + s)} \tag{4.6}$$

where x, s and p are the concentration of cell mass, substrate, and product (ethanol), respectively, v is the liquid volume within the reactor, μ is the specific growth rate, η is the specific productivity, Y is the yield coefficient, s_0 is the concentration of the substrate feed, and u is the feeding rate of the fed-batch reactor which is the only manipulated variable in this process.

For the process model considered, the rate of substrate feed into the fed-batch reactor (u) is constrained by $0 \leq u$ (L/h) ≤ 12, the volume of the reactor is limited by the 200 liters, the operating time of the fed-batch reactor is fixed to be 63 hr, and the yield coefficient is assumed to be constant of 0.1. In addition, the values of initial conditions and kinetic parameters of this process are described in Table 4.1.

Table 4.1 Initial conditions and kinetic parameters of ethanol production process

Initial conditions		Kinetic parameters	
$x(0)$ =	1 g/L	K_p =	16.0 g/L
$s(0)$ =	150 g/L	K_s =	0.22 g/L
$p(0)$ =	0 g/L	K_p' =	71.5 g/L
$v(0)$ =	10 L	K_s' =	0.44 g/L
μ_0 =	0.408 hr^{-1}		
η_0 =	1 hr^{-1}		

4.2 Optimal Control of Ethanol Production in a Fed-Batch Reactor

The aim of an optimal control is to determine a control profile minimizing (or maximizing) a given objective function subject to process constraints. With the optimal control policy, the controlled system is driven from an initial state to a final desired state in an optimal way.

4.2.1 Problem Formulation

In the current work, two major optimal control problems related to ethanol production in a fed-batch operation are studied. Maximum amount of desired ethanol product and Minimum operational time problems are studied to determine an optimal feed profile of the substrate. The obtained optimal feed profile has to satisfy the specified objective function and other desired process constraints. Such optimal control problems can be described as follows.

4.2.1.1 Maximum Ethanol Production Problem

In this type of the optimal control problem, the objective is to determine the optimal feed profile that maximize the amount of the desired ethanol product for a given fixed terminal time subject to bounds on the reactor volume and the substrate feed rate. This problem can be stated as follows.

Find the substrate feed rate $u(t)$ over $t \in [t_0, t_f]$ for maximizing

$$J = p(t_f) \times v(t_f) \tag{4.7}$$

subject to

$$\frac{dx}{dt} = \psi[x(t), u(t), t] \tag{4.8}$$

$$x(t_0) = x_0 \tag{4.9}$$

$$0 \leq v(t)\ (\mathrm{L}) \leq 200 \tag{4.10}$$

$$0 \leq u(t)\ (\mathrm{L/h}) \leq 12 \tag{4.11}$$

where t_f denotes the terminal time of the operation, $p(t_f)$ and $v(t_f)$ are the final ethanol concentration and the final liquid volume in the reactor respectively, and J is the performance index which is the amount of the final ethanol product. Eq. (4.8) is the

system of ordinary differential equations which are described by Eq. (4.1) through (4.4), Eq. (4.9) is the initial conditions for Eq. (4.8), Eqs. (4.10) and (4.11) are the upper and lower bounds on the liquid volume of the reactor and the substrate feed rate respectively.

4.2.1.2 Minimum Operation Time Problem

The purpose of this optimal control problem is to determine the optimal feed profiles which give the desired final ethanol product in minimum operation time, thus the performance index is the final time whereas the amount of desired final ethanol product is defined as a terminal constraint. The formulation of this problem can be shown as follows.

Find the substrate feed rate $u(t)$ over $t \in [t_0, t_f]$ for minimizing

$$J = t_f \tag{4.12}$$

subject to

$$\frac{dx}{dt} = \psi[x(t), u(t), t] \tag{4.8}$$

$$x(t_0) = x_0 \tag{4.9}$$

$$0 \leq v(t)\ (\mathrm{L}) \leq 200 \tag{4.10}$$

$$0 \leq u(t)\ (\mathrm{L/h}) \leq 12 \tag{4.11}$$

$$p(t_f) \times v(t_f) = pv^* \tag{4.13}$$

where pv^* is the amount of desired ethanol product at the end of the operation and t_f is the final operation time. Eq. (4.13) is the terminal constraint of this minimum operation time problem.

4.2.2 Simulation Results

In the current work, the sequential approach is applied for solving these optimal control problems. Typically, a piecewise constant approximation over equally spaced time intervals is made for the control variables in the sequential approach. To hold the inequality constraint, a penalty function method is applied in this work.

4.2.2.1 Maximum Ethanol Production

All simulation results given here are based on the optimal control problem in the case of maximum ethanol production. The objective is to find the optimal feed rate profile which maximizes the amount of final ethanol product in the fixed operation time problem.

Nominal Case

In the nominal case, the specified final batch time (t_f) is fixed at 63 hr, the batch length is divided in to 10 equal stages and the control variable profiles are piecewise constant. The simulation results obtained from solving the optimal control problem by using the sequential approach are shown in Figures 4.2 and 4.3. In Figure 4.2, the optimal feed profile of substrate slowly increases and close to zero at the end of the operation, signifying the liquid volume in the reactor does not exceed the reactor volume. In Figure 4.3, the dotted line, dashed line and solid line represent the cell mass, substrate and product (ethanol) concentration respectively. At the end of the operation, the final liquid volume is 200 liters, the final ethanol concentration is 103.73 g/L and the performance index is 20,747.

Moreover, the performance index from the proposed method is compared with the performance index from the IDP method (Luus, 1993), ICRS/DS method (Banga et al., 1997) and CAFNN method (Xiong and Zhang, 2004) as shown in Table 4.2. It can be seen that the proposed method give a better control performance compared to the ICRS/DS and CAFFNN methods. It is noted that although the IDP method provides the best value of performance index, its computational time is longer than that used in the proposed method.

Table 4.2 The results of different solution methods of optimal control problem

Method	Performance index
IDP (Luus, 1993)	20,841
ICRS/DS (Banga et al., 1997)	20,715
CAFNN (Xiong and Zhang, 2004)	20,627
Present study	20,747

Effect of Time Interval

In previous section, the optimal control problem is formulated as a fixed final time with equally spaced time interval of 6.3 hr. In this section, the effect of the time interval on the control performance is studied. Figure 4.5 shows the control response of the optimal feed rate profile computed using the time interval of 3.15 hr which shown in Figure 4.4. It is found that the performance index obtained from the decreasing time interval is 20,841 which greater than that obtained in the nominal case. In addition, the obtained performance index equals to that obtained from the IDP method (Luus, 1993) which is claimed for a global optimum of this process. This can be explained by that as the length of time intervals decreases, the approximated optimal profile with piecewise constant policy is closer to the actual optimal profile.

Effect of Switching Time

In this section, the effect of the switching time is studied. Non-uniform control vector parameterization (free switching time) is applied in order to improve the control performance of the optimal control in this process. In Figure 4.6, the optimal feeding profile obtained from the optimal control based on non-uniform control vector parameterization is illustrated. The control response of the obtained optimal feed profile is shown in Figure 4.7. With the non-uniform control vector parameterization, the optimal feed rate does not reach the upper limit and its control performance is

20,798 which is slightly higher than that obtained from using the uniform control vector parameterization (fixed switching time) as shown in Figure 4.2.

Finally, the simulation results based on the optimal control problem in the case of maximum ethanol production are summarized in Table 4.3. This points the effectiveness of the proposed method to improve the control performance of this ethanol production process.

Table 4.3 Comparison of the simulation results obtained from optimal control in the case of maximum ethanol production

Case studies	Performance index
Optimal control in the nominal case	20,747
Optimal control in the case of decreased time interval	20,841
Optimal control in the case of free switching time	20,798

4.2.2.2 Minimum Operation Time

The simulation results presented here are based on the optimal control problem in the case of minimum operation time. The objective of this case is to find the optimal feed rate profile which minimize the batch time of the operation subject to a terminal constraint on the desired amount of final ethanol product.

In this case, the batch length is divided in to unequal stages and the control variable profiles are piecewise constant. The terminal constraint which is the desired amount of final ethanol product is defined as 20,750. The optimal feed rate profile and the switching time are optimized while minimizing the final batch operation time. The simulation results of this case are shown in Figures 4.8 and 4.9. In Figure 4.8, the optimal feed profile obtained in this case gives the desired amount of final ethanol product equal to that obtained in the nominal case (Figure 4.2) but the final operation time of this case is 58.86 min. It can be seen that the minimum operation time can

reduce the final batch time compared to the nominal case resulting to the reducing of the operation cost.

4.2.3 Conclusions

The optimal control of a fed-batch reactor for ethanol production has been studied in this work. The solution of the optimal control problem is computed using a sequential model solution and optimization method. Two types of the optimal control problem related to the ethanol production process in a fed-batch reactor (maximum ethanol production and minimum operation time) were considered. In the maximum ethanol production problem, the effects of a time interval and switching time are investigated in order to improve the control performance. From the simulation results, it can be seen that decreasing the time interval and using the non-uniform control vector parameterization (free switching time) provide a better control performance compared to the nominal case. In addition, the minimum operation time is also studied in order to improve the control performance of this process.

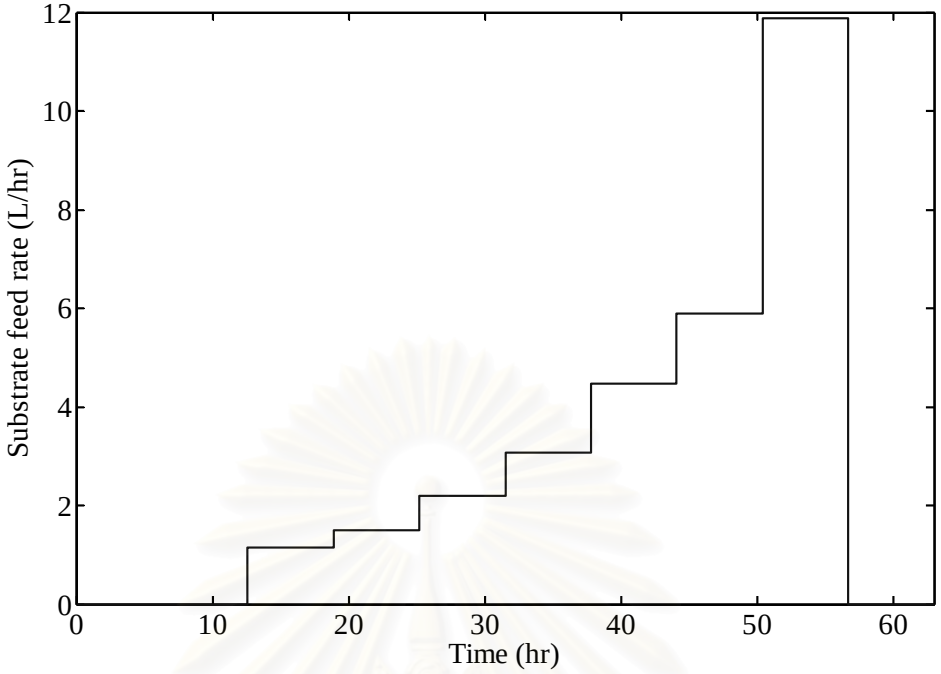

Figure 4.2 Optimal feed profile in the nominal case

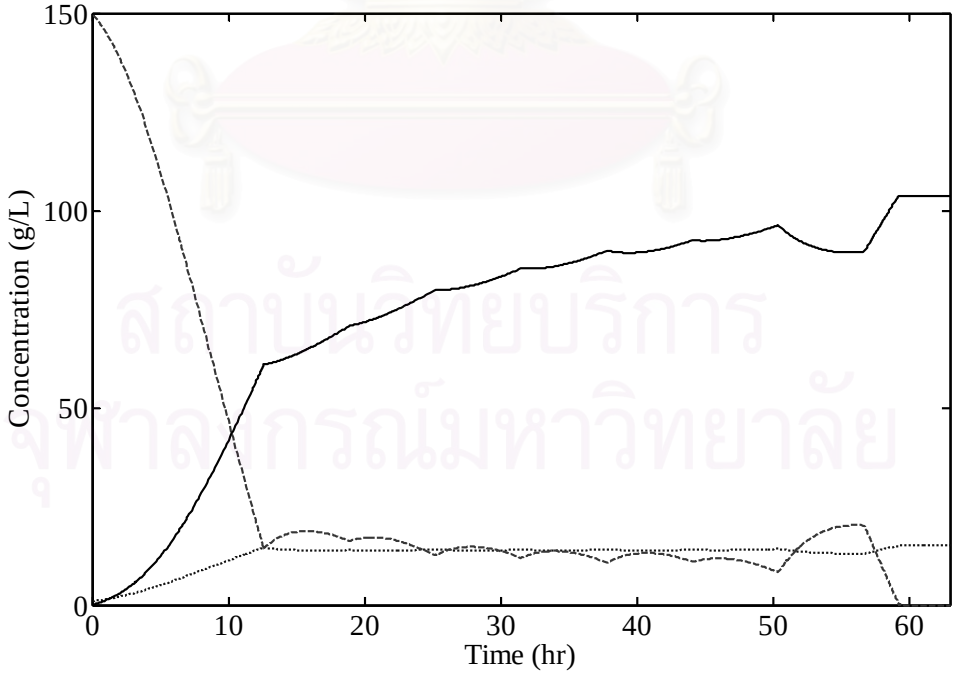

Figure 4.3 Concentration profile in the nominal case

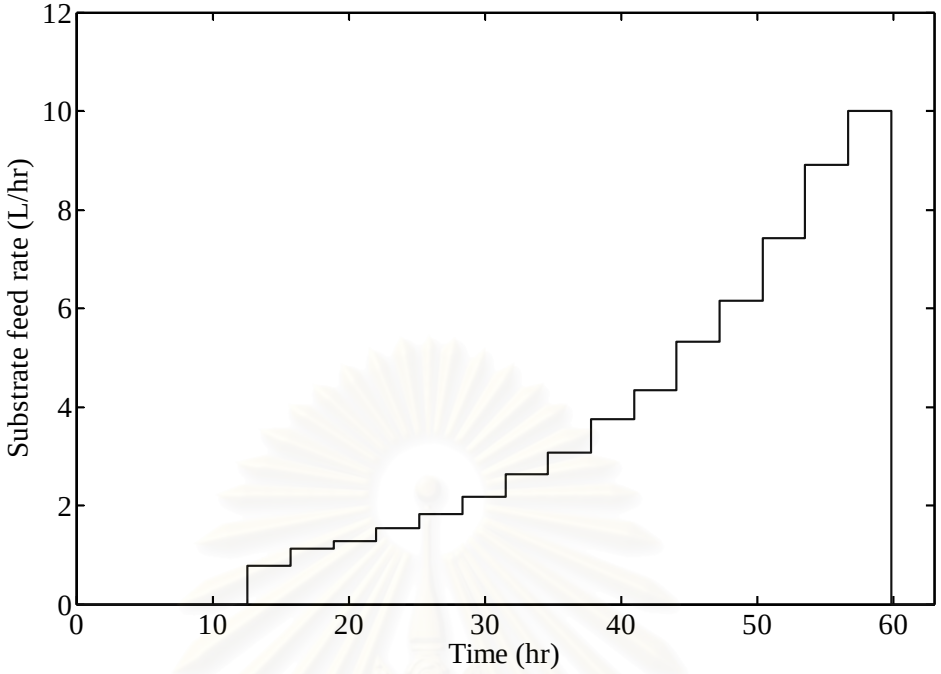

Figure 4.4 Optimal feed profile in the case of decreased time interval

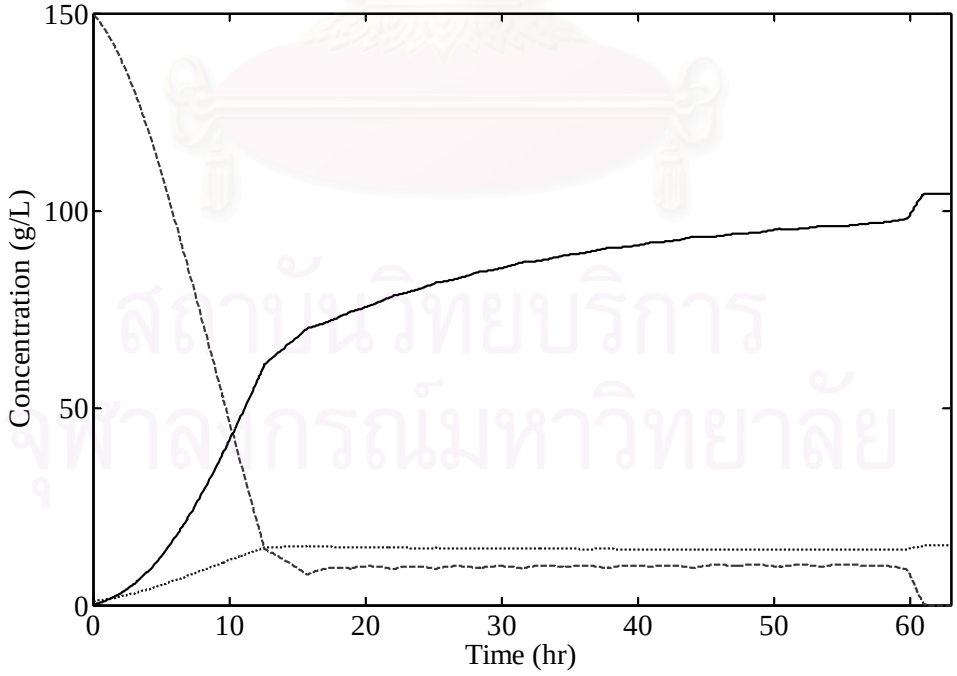

Figure 4.5 Concentration profile in the case of decreased time interval

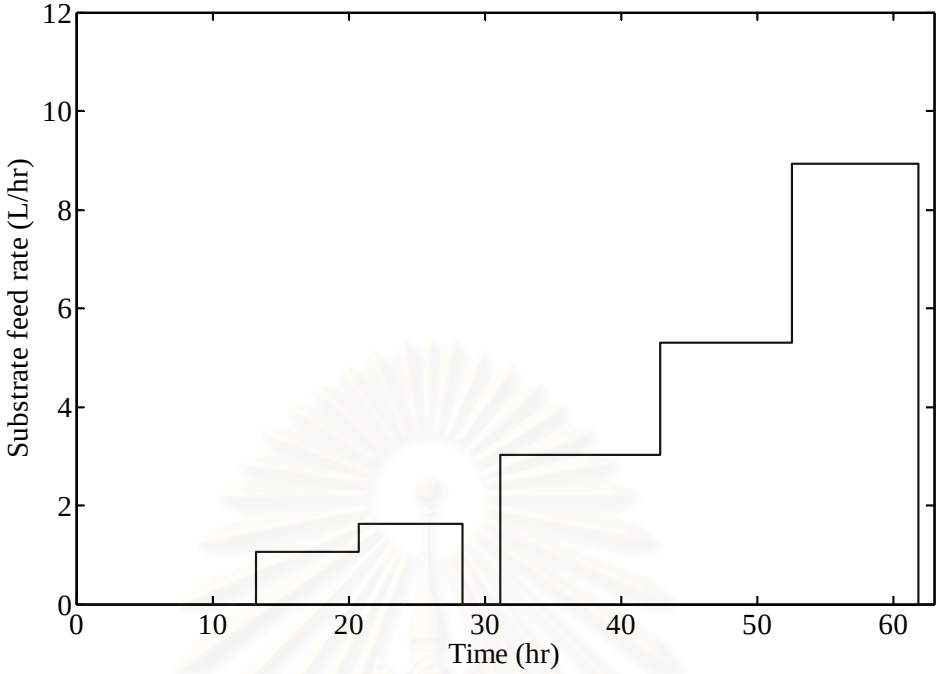

Figure 4.6 Optimal feed profile in the case of free switching time

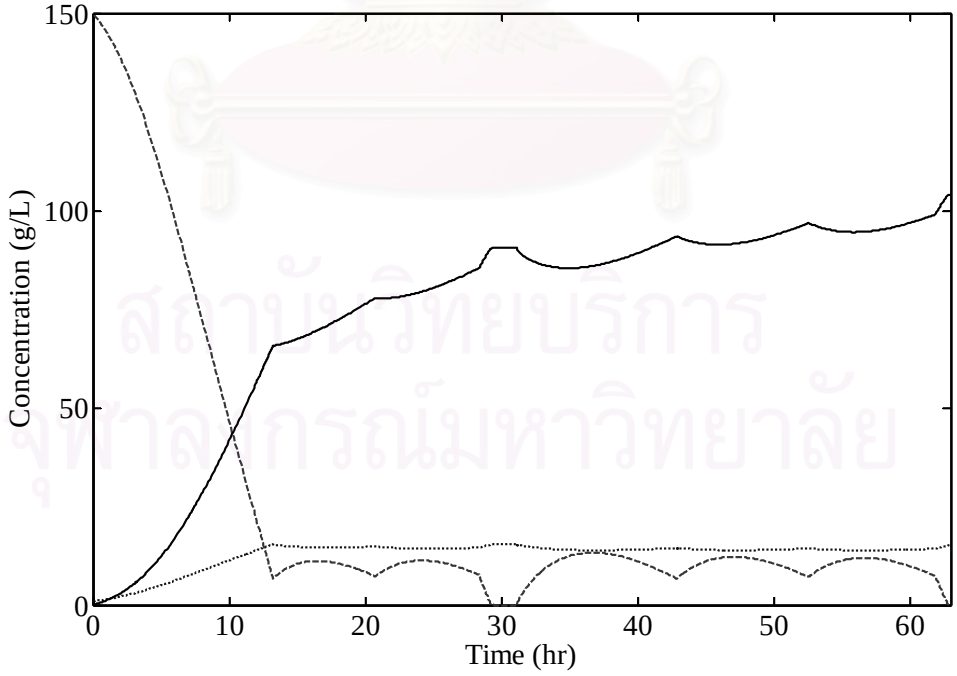

Figure 4.7 Concentration profile in the case of free switching time

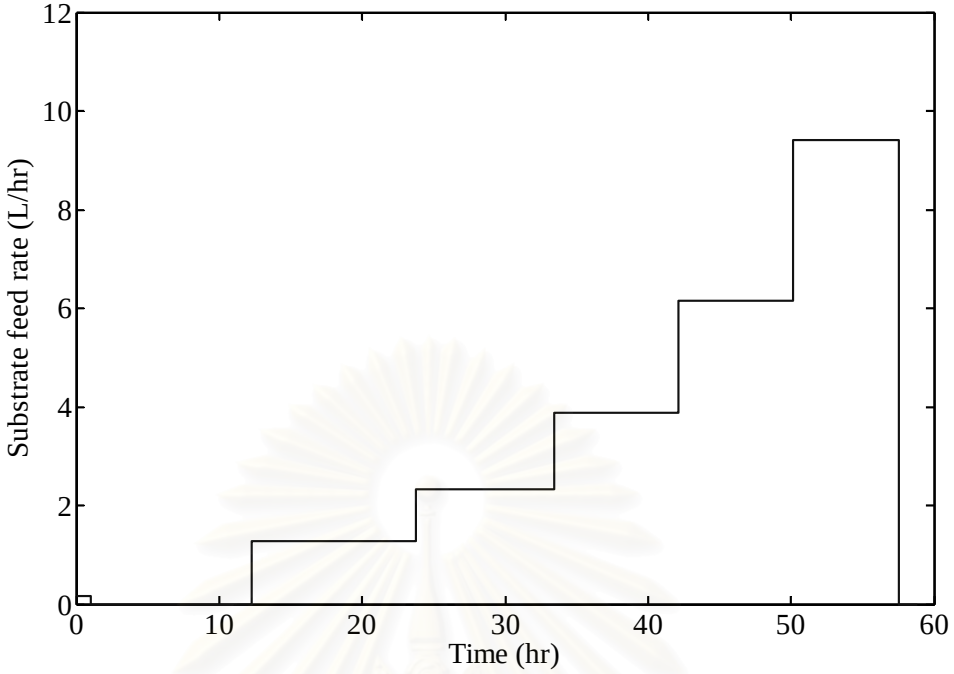

Figure 4.8 Optimal feed profile in the case of minimum operation time

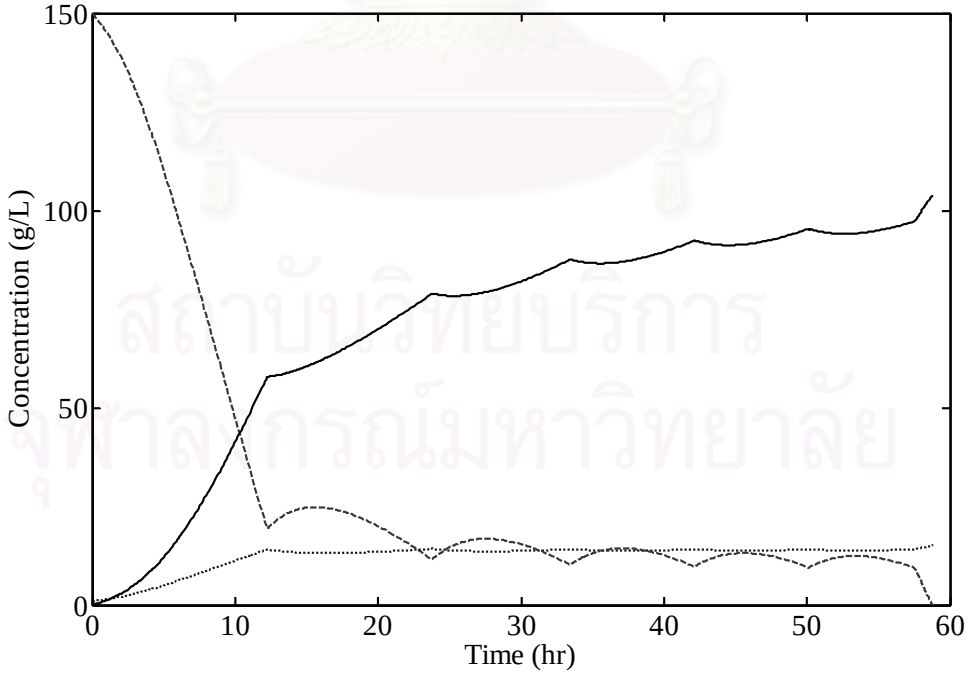

Figure 4.9 Concentration profile in the case of minimum operation time

CHAPTER V

ON-LINE OPTIMAL CONTROL WITH NEURAL NETWORK ESTIMATOR

This chapter describes the implementation of an on-line optimal control with neural network estimator to a fed-batch reactor. The mathematical model of ethanol production in CHAPTER IV is used here to demonstrate the control performance of an on-line optimal control with neural network estimator. Furthermore, the application of an artificial neural network as a state estimator is presented.

5.1 Neural Network Estimator

In chemical industrial processes, some process variables cannot be directly measured or are difficult to measure. Due to this fact, a state estimator is proposed to cope with this problem. Artificial neural network is found to be the one of various tools that can be used to estimate the unmeasured process variables since it is considered as an universal approximator that can approximate any arbitrary function.

5.1.1 Neural Network Training

In the current work, a multilayer feed-forward neural network was applied as a state estimator to determine the unmeasured process variable which is the substrate concentration of the ethanol production process in a fed-batch reactor. For the design of neural network, the data sets for the network training are divided into three different sets; training sets, validation sets and testing sets which are generated by using a random sequence of the inlet feed rate $u(t)$. The training sets, validation sets and testing sets are composed of 60, 30 and 10 percent of the data generated sets respectively. In addition, these data sets are normalized by Z-score Standardization for achieving a good performance of neural network model. Levenberg-Marquardt

Backpropagation algorithm (as shown in APPENDIX A) with the early stopping mechanism is used to train this multilayer feed-forward neural network. The Mean Square Error (MSE) is used as the criterion for the network selection and also for the stopping weights and biases adjustment.

The Mean Square Error for the various architectures of a neural network is presented in Table 5.1. The architecture which gives the minimum value of the Mean Square Error is considered for application as a neural network estimator. The optimum architecture of a neural network for the substrate concentration estimation is (8-5-5-1) which is shown in Figure 5.1. In this Figure, the input layer is composed of eight nodes which are the substrate feed rate (*u*), the cell mass concentration (*x*), the ethanol concentration (*p*) and the liquid volume in the reactor (*v*) at time *t* and *t*-1. The both hidden layers are composed of five nodes with the log-sigmoid transfer function and one output node with a linear transfer function in the output layer which is the estimating value of the substrate concentration at time *t*.

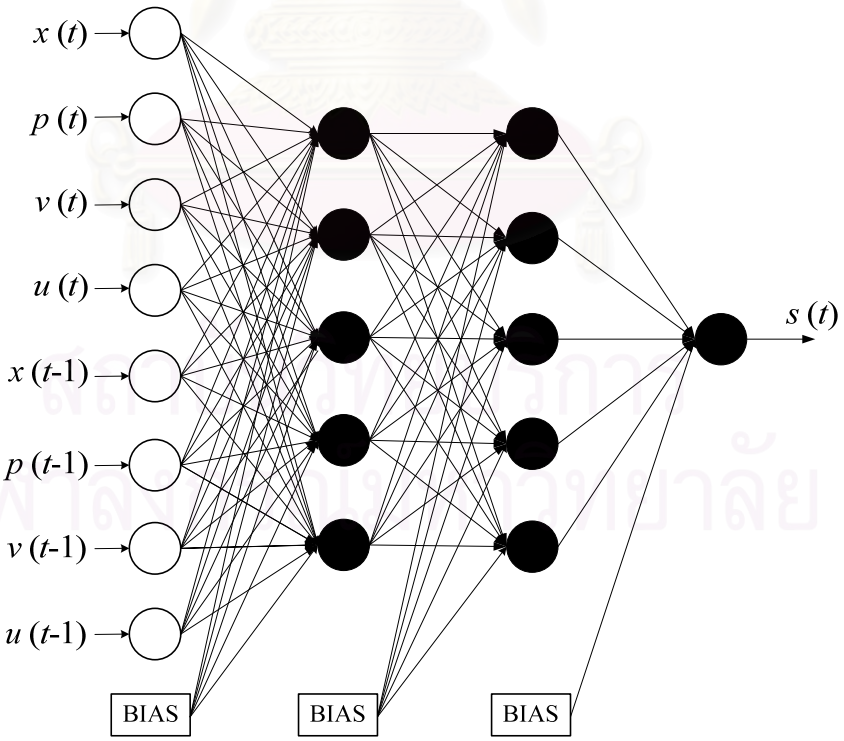

Figure 5.1 Optimum architecture of neural network estimator

Table 5.1 The mean square error for the various architectures of the neural network

Nodes in 1st hidden layer	Nodes in 2st hidden layer	Mean Square Error (MSE)
3	0	11.669
3	3	9.299
3	5	9.290
3	7	10.231
3	9	8.876
5	0	12.042
5	3	6.525
5	5	3.479
5	7	13.007
5	9	7.954
7	0	12.026
7	3	9.519
7	5	9.599
7	7	5.446
7	9	8.772
9	0	8.208
9	3	4.952
9	5	11.520
9	7	7.210
9	9	12.278

5.1.2 Design of a Neural Network Estimator

To design an estimator, the weights and biases of the optimum neural network architecture are used to design the neural network estimator. The input data which are the substrate feed rate (*u*), the cell mass concentration (*x*), the ethanol concentration (*p*) and the liquid volume in the reactor (*v*) at time *t* and *t*-1 are scaled up with mean and its standard deviation as described in Eq. (5.1). The neural network estimator output value which is the substrate concentration is rescaled to find the value in the original units as described in Eq. (5.2).

$$\text{Input}_{\text{scaleup}} = \frac{\text{Input}_{\text{actual}} - \text{Mean of Input}}{\text{Std. of Input}} \tag{5.1}$$

$$\text{Output}_{\text{actual}} = \text{Output}_{\text{scaleup}} \times \text{Std. of Output} + \text{Mean of Output} \tag{5.2}$$

From the optimum neural network in Figure 5.1, the neural network estimator output is defined as a_1^4, which is an output value of the first node in the forth layer, and is described in Eq. (5.3)

$$a_1^4 = f_1^4\left(\sum_{k=1}^{5}\left(w_{1k}^4 a_k^3 \right) + b_1^4 \right) \tag{5.3}$$

where a_k^3 is an output value of the *k* node in the third layer and is defined in Eq. (5.4)

$$a_k^3 = f_j^3\left(\sum_{k=1}^{5}\left(w_{jk}^3 a_k^2 \right) + b_j^3 \right) \tag{5.4}$$

where *j* is the sequence of node in the layer and a_k^2 is an output value of the *k* node in the second layer which is defined in Eq. (5.5).

$$a_k^2 = f_j^2\left(\sum_{k=1}^{8}\left(w_{jk}^2 a_k^1 \right) + b_j^2 \right) \tag{5.5}$$

From Eq. (5.5), the a_k^1 is the neural network input after scale up process. The *f* and *b* in Eqs. (5.3) to (5.5) are the transfer functions and biases of each node in each layer respectively. The scaled up parameters which are the mean and standard deviation of each variable are showed in Table 5.2. The transfer functions of the optimum neural network in each layer are showed in Table 5.3.

Table 5.2 The scaled up parameters of the neural network estimator

Variables		Scaled up Parameters	
		Mean	Standard Deviation (Std.)
Input # 1	$x(t)$	11.306	5.106
Input # 2	$p(t)$	55.955	29.439
Input # 3	$v(t)$	199.440	111.920
Input # 4	$u(t)$	5.999	3.508
Input # 5	$x(t\text{-}1)$	11.302	5.109
Input # 6	$p(t\text{-}1)$	55.932	29.450
Input # 7	$v(t\text{-}1)$	199.340	111.930
Input # 8	$u(t\text{-}1)$	5.998	3.509
Output	$s(t)$	37.978	52.093

Table 5.3 The transfer functions in each layer of the neural network estimator

Layer	Variables	Transfer Function	Equation
Layer # 2	f_j^2	Log-sigmoid	$f_j^2(x) = \frac{1}{1+e^{-x}}$
Layer # 3	f_j^3	Log-sigmoid	$f_j^3(x) = \frac{1}{1+e^{-x}}$
Layer # 4	f_j^4	Linear	$f_j^4(x) = x$

Tables 5.4 to 5.6 show the weights and biases of the optimum neural network architecture for the second, third and fourth layer respectively. The w_{jk}^{i} and b_{j}^{i} are defined for the weights of output from node *k* in layer *i*-1 to node *j* in layer *i* and the bias of node *j* in layer *i* respectively. From Eqs. (5.1) to (5.5), the scaled up parameters in Table 5.2, the transfer functions in Table 5.3 and the weights and biases in Tables 5.4 to 5.6, the neural network estimator is accomplish as described in APPENDIX B.

To test the performance of the neural network estimator, the comparison of the actual and neural network estimated values of the substrate concentration is shown in Figure 5.2. In this figure, the solid lines represent the actual values which are simulated from the actual model, the dotted lines represent the estimated values which are obtained from the neural network estimator and the percentage relative error is only 6.64 %.

Table 5.4 Weights and biases in the second layer of the optimum neural network

Weighting Factors (w_{jk}^{i})				
w_{11}^{2} = 1.4195	w_{12}^{2} = -0.0687	w_{13}^{2} = -0.0299	w_{14}^{2} = -0.0001	w_{15}^{2} = -0.0284
w_{21}^{2} = 1.4177	w_{22}^{2} = -6.7633	w_{23}^{2} = -0.0510	w_{24}^{2} = 0.0004	w_{25}^{2} = 0.0008
w_{31}^{2} = -2.1423	w_{32}^{2} = -0.5977	w_{33}^{2} = 3.3221	w_{34}^{2} = 0.0552	w_{35}^{2} = -5.5415
w_{41}^{2} = -1.4638	w_{42}^{2} = -0.2673	w_{43}^{2} = 2.5690	w_{44}^{2} = 0.0278	w_{45}^{2} = -2.9306
w_{51}^{2} = -2.7440	w_{52}^{2} = -0.4310	w_{53}^{2} = -0.4392	w_{54}^{2} = -0.0078	w_{55}^{2} = 0.1563
Weighting Factors (Continued)				Bias (b_{j}^{i})
w_{16}^{2} = 0.0794	w_{17}^{2} = 0.0265	w_{18}^{2} = 0.0004		b_{1}^{2} = 0.4113
w_{26}^{2} = 0.0295	w_{27}^{2} = 0.0084	w_{28}^{2} = 0.0015		b_{2}^{2} = -10.969
w_{36}^{2} = -2.2685	w_{37}^{2} = -1.6976	w_{38}^{2} = -0.0756		b_{3}^{2} = -0.7745
w_{46}^{2} = -1.3661	w_{47}^{2} = -1.8356	w_{48}^{2} = -0.0501		b_{4}^{2} = -0.5188
w_{56}^{2} = 1.0148	w_{57}^{2} = -0.2332	w_{58}^{2} = -0.0139		b_{5}^{2} = -1.6070

Table 5.5 Weights and biases in the third layer of the optimum neural network

Weighting Factors (w_{jk}^{i})				
w_{11}^{3} = -5.0598	w_{12}^{3} = 3.3669	w_{13}^{3} = 0.4493	w_{14}^{3} = -0.9277	w_{15}^{3} = 0.4949
w_{21}^{3} = -3.0971	w_{22}^{3} = -6.3727	w_{23}^{3} = 0.3162	w_{24}^{3} = 1.5130	w_{25}^{3} = -5.0696
w_{31}^{3} = 4.1235	w_{32}^{3} = 3.8922	w_{33}^{3} = -1.1180	w_{34}^{3} = -4.2215	w_{35}^{3} = 4.6330
w_{41}^{3} = 6.5760	w_{42}^{3} = 0.6943	w_{43}^{3} = -1.1201	w_{44}^{3} = -1.1562	w_{45}^{3} = 0.9267
w_{51}^{3} = 4.6893	w_{52}^{3} = -5.7701	w_{53}^{3} = -0.5034	w_{54}^{3} = 5.2684	w_{55}^{3} = -2.5007
Bias (b_{j}^{i})				
b_{1}^{3} = 5.2857				
b_{2}^{3} = 4.9311				
b_{3}^{3} = -4.2589				
b_{4}^{3} = 1.3860				
b_{5}^{3} = 4.2222				

Table 5.6 Weights and biases in the fourth layer of the optimum neural network

Weighting Factors (w_{jk}^{i})	Bias (b_{j}^{i})
w_{11}^{4} = 4.4528	b_{1}^{4} = 3.2266
w_{12}^{4} = -1.3831	
w_{13}^{4} = -0.7837	
w_{14}^{4} = -3.1384	
w_{15}^{5} = -2.7548	

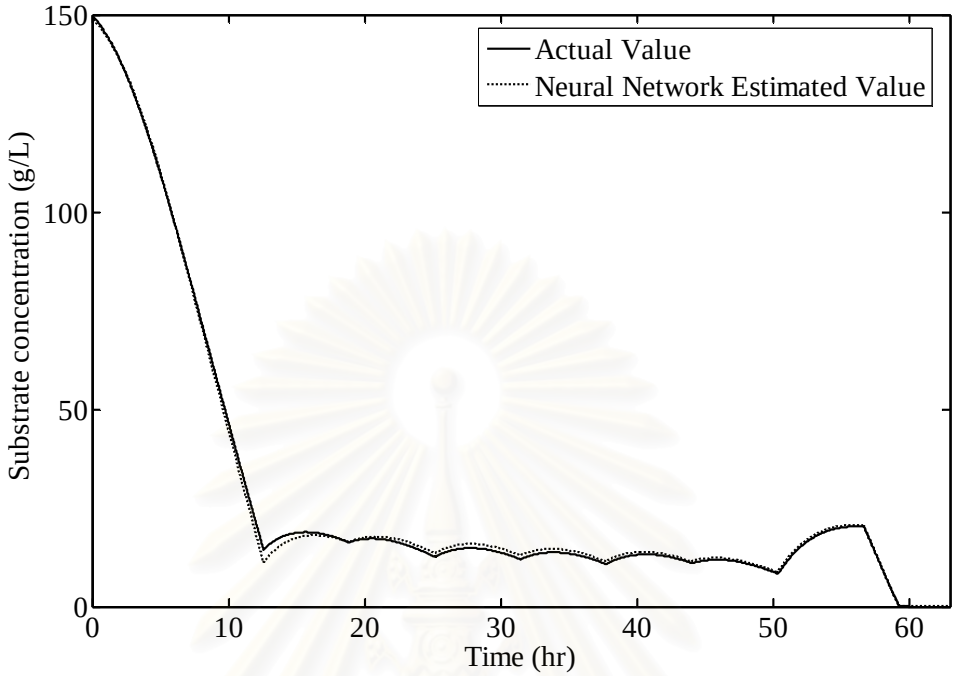

Figure 5.2 The comparison of the actual and neural network estimated values of the substrate concentration

5.2 On-line Optimal Control with Neural Network Estimator for Ethanol Production in a Fed-Batch Reactor

Due to the presence of unknown disturbances and model-plant mismatches, the off-line calculated optimal control profile may not give the optimal performance when applied to the actual process. If this optimal control profile is still used to control the process, the process output at the end of the operation will be significantly different from the desired output. Therefore, the on-line optimal control with neural network estimator that the unmeasured feedback information during the earlier stage is estimated by the neural network and is also used to re-calculate the control profile of the remaining batch stage is proposed in this work.

5.2.1 Formulation of On-line Optimal Control with Neural Network Estimator

In this section, the on-line optimal control with neural network estimator is proposed to deal with the limitation of the optimal control strategy. The mathematical model in section 4.1 is used to demonstrate the control performance of the proposed method. The control structure of the on-line optimal control with neural network estimator for a fed-batch reactor is illustrated in Figure 5.3. In this figure, it is assumed that the substrate concentration is an only unmeasured variable. The measured variables consist of the substrate feed rate (u), the cell mass concentration (x), the ethanol concentration (p) and the liquid volume in the reactor (v). Artificial neural network is employed as a state estimator to estimate the substrate concentration with the measured variable. The feedback information which consists of the cell mass concentration, the ethanol concentration, the liquid volume in the reactor and the estimated substrate concentration are used to update the information from the system. The optimal control computes the new control profile for the remaining batch stage by solving the optimal control problem. This procedure is repeated for the next sampling time until the end of the operation.

The basic algorithm of an on-line optimal control with neural network estimator is summarized as the follows.

Step 1: Specify the objective function and the initial states of the system.

Step 2: Calculate the control profile that maximizing the objective function over the batch stage.

Step 3: Implement the initial value of the controls.

Step 4: Measure the measured outputs and estimate the unmeasured outputs and go back to Step 2 in order to calculate the new control profile for the next sampling time based on the new feedback information.

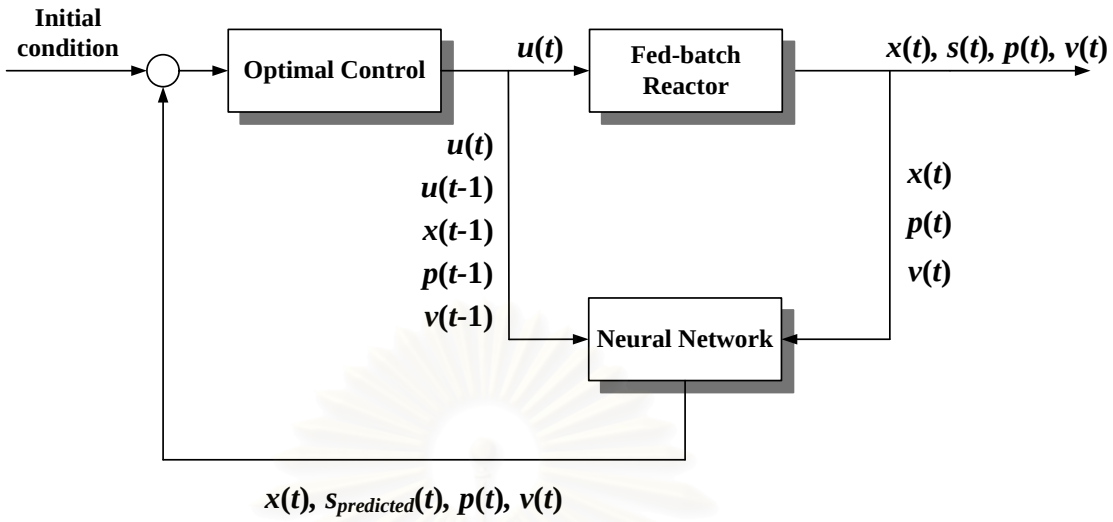

Figure 5.3 Structure of an on-line optimal control with neural network estimator

5.2.2 Simulation Results

This section demonstrates the implementation of the on-line optimal control with neural network estimator for the ethanol production process in a fed-batch reactor. The control performance of the on-line optimal control with neural network estimator is compared with that of the optimal control strategy which is described in CHAPTER 4. In these simulation studies, the specified final batch time (t_f) is fixed at 63 hr and the frequency of updated control action is chosen to be 6.3 hr. Therefore, the number of the future controls is equal to 10.

To test the control performances, the on-line optimal control with neural network estimator and the optimal control strategy are then tested in the cases of the disturbance rejection and the model-plant mismatch.

5.2.2.1 Disturbance Rejection Case

To test the control performances in the disturbance rejection case, a 20% change in the substrate feed concentration (s_0) are applied to the system.

For the case of a 20% increase in the substrate feed concentration, the results show that the on-line optimal control with neural network estimator (Figure 5.4) gives the better control performance compared with the optimal control strategy (Figure 5.5). In these figures, the optimal feed profile (Figure 5.5) gives the control performance equal to 18,528 but equal to 14,536 when applied to the actual process because of the effect of disturbance. To realize this fact, the optimal feed profile (Figure 5.4) is also applied to improve the control performance of this system.

For the case of a 20% decrease in the substrate feed concentration, it can be seen that the on-line optimal control with neural network estimator also gives the better control performance compared with the optimal control strategy as illustrated in Figures 5.6 and 5.7.

Table 5.7 summarizes the control results of the on-line optimal control with neural network estimator and the optimal control strategy in term of the performance index which is an amount of the desired ethanol product. They clearly indicate that the control performances of the on-line optimal control with neural network estimator are greater than that of the optimal control strategy in all case studies.

5.2.2.2 Model-Plant Mismatch Case

To test the control performances in the model-plant mismatch case, a change in the yield coefficient (Y), maximum production rate (η_0), maximum growth rate (μ_0) and kinetic constant (K) are applied to the system.

Figures 5.8 to 5.11 show the results of the model-plant mismatches in the yield coefficient (Y) with the 20% increase and 20% decrease respectively. In the both cases, the on-line optimal control with neural network estimator gives the better control performances compared with the optimal control strategy. This is because the effect of the model-mismatch is compensated by the feedback information in the on-line optimal control with neural network estimator.

For the model-plant mismatches of the maximum production rate (η_0) and maximum growth rate (μ_0), the 20% increase and decrease of the maximum production rate and maximum growth rate are applied to this system. The control responses of these cases are illustrated in Figures 5.12 to 5.19. The control performances of the on-line optimal control with neural network estimator and the optimal control strategy are shown in Table 5.8. From this table, it clearly shows that the control performances of the on-line optimal control with neural network estimator are greater than that of the optimal control strategy in all case studies.

Moreover, the effects of model-plant mismatches of the kinetic constant (K) in this process are also studied. The on-line optimal control with neural network estimator also gives the better control performance compared with the optimal control strategy as shown in Table 5.8.

Table 5.7 Comparison of the control performance of the on-line optimal control with neural network estimator and optimal control strategy in the disturbance rejection case

Case studies	Performance Index	
	On-line optimal control with NN	Optimal Control
+ 20% s_0	19,403	14,536
- 20% s_0	18,388	16,335

Table 5.8 Comparison of the control performance of the on-line optimal control with neural network estimator and optimal control strategy in the model-plant mismatch case

Case studies	Performance Index	
	On-line optimal control with NN	Optimal Control
+ 20% Y	19,767	15,959
- 20% Y	18,363	14,706
+ 20% η_0	20,242	15,170
- 20% η_0	18,194	16,442
+ 20% μ_0	18,453	14,730
- 20% μ_0	18,846	13,831
+ 20% K_s	19,082	17,866
- 20% K_s	18,274	17,917
+ 20% K_s'	18,278	17,179
- 20% K_s'	19,066	17,765
+ 20% K_p	17,286	14,796
- 20% K_p	18,158	15,272
+ 20% K_p'	19,394	18,192
- 20% K_p'	18,295	17,914

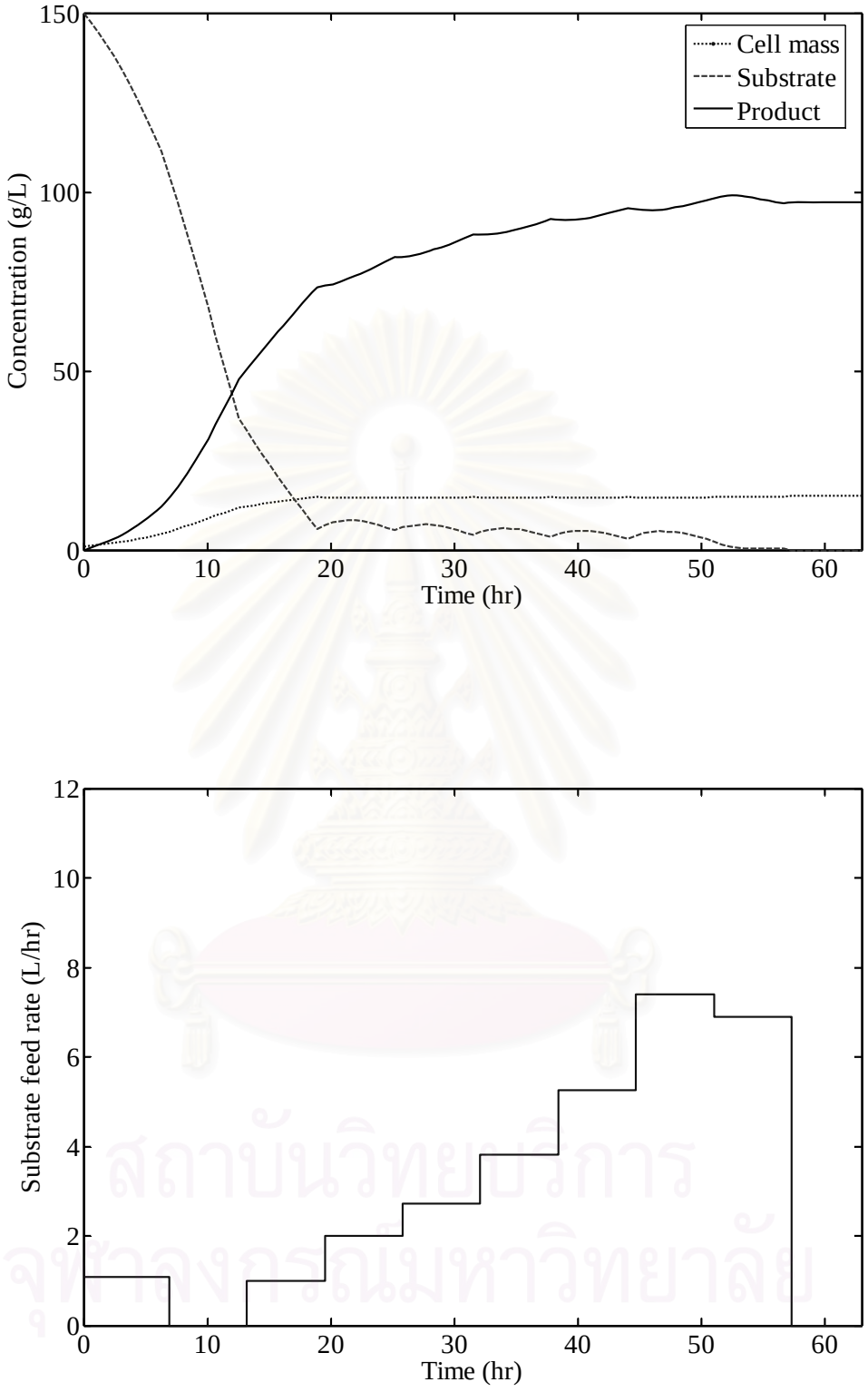

Figure 5.4 Control response of an on-line optimal control with neural network estimator for disturbance rejection with 20% increase of the substrate feed concentration

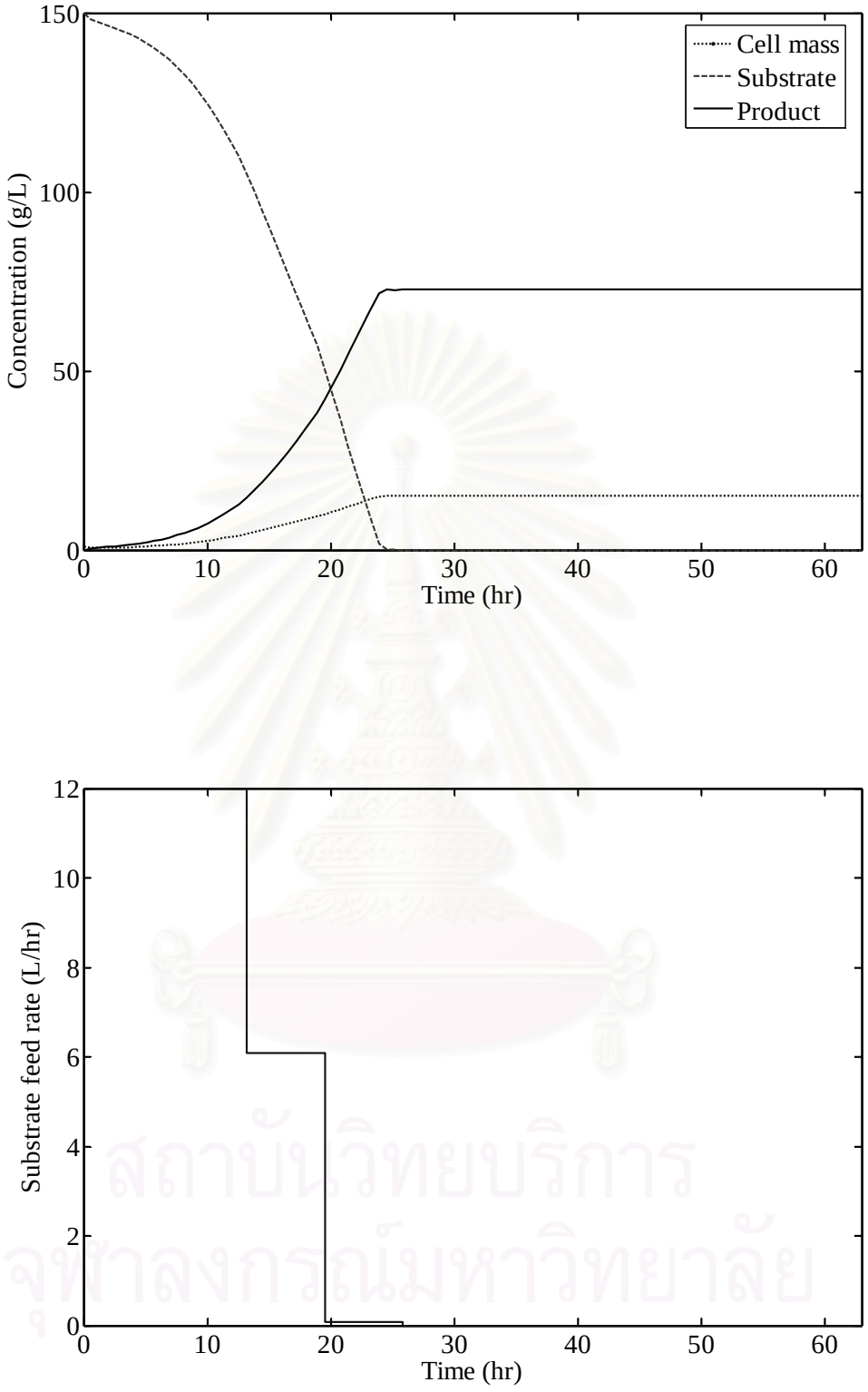

Figure 5.5 Control response of an optimal control for disturbance rejection with 20% increase of the substrate feed concentration

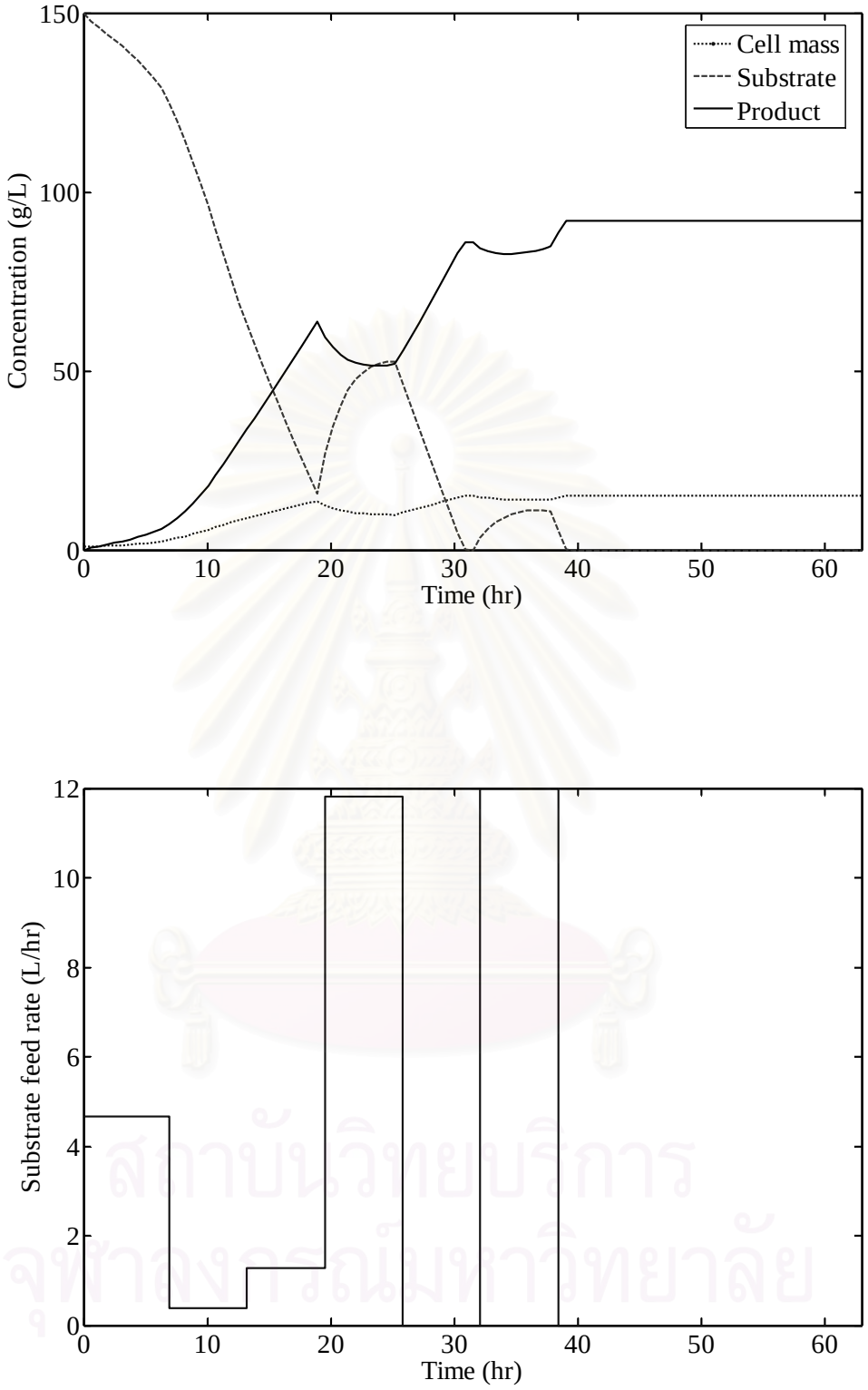

Figure 5.6 Control response of an on-line optimal control with neural network estimator for disturbance rejection with 20% decrease of the substrate feed concentration

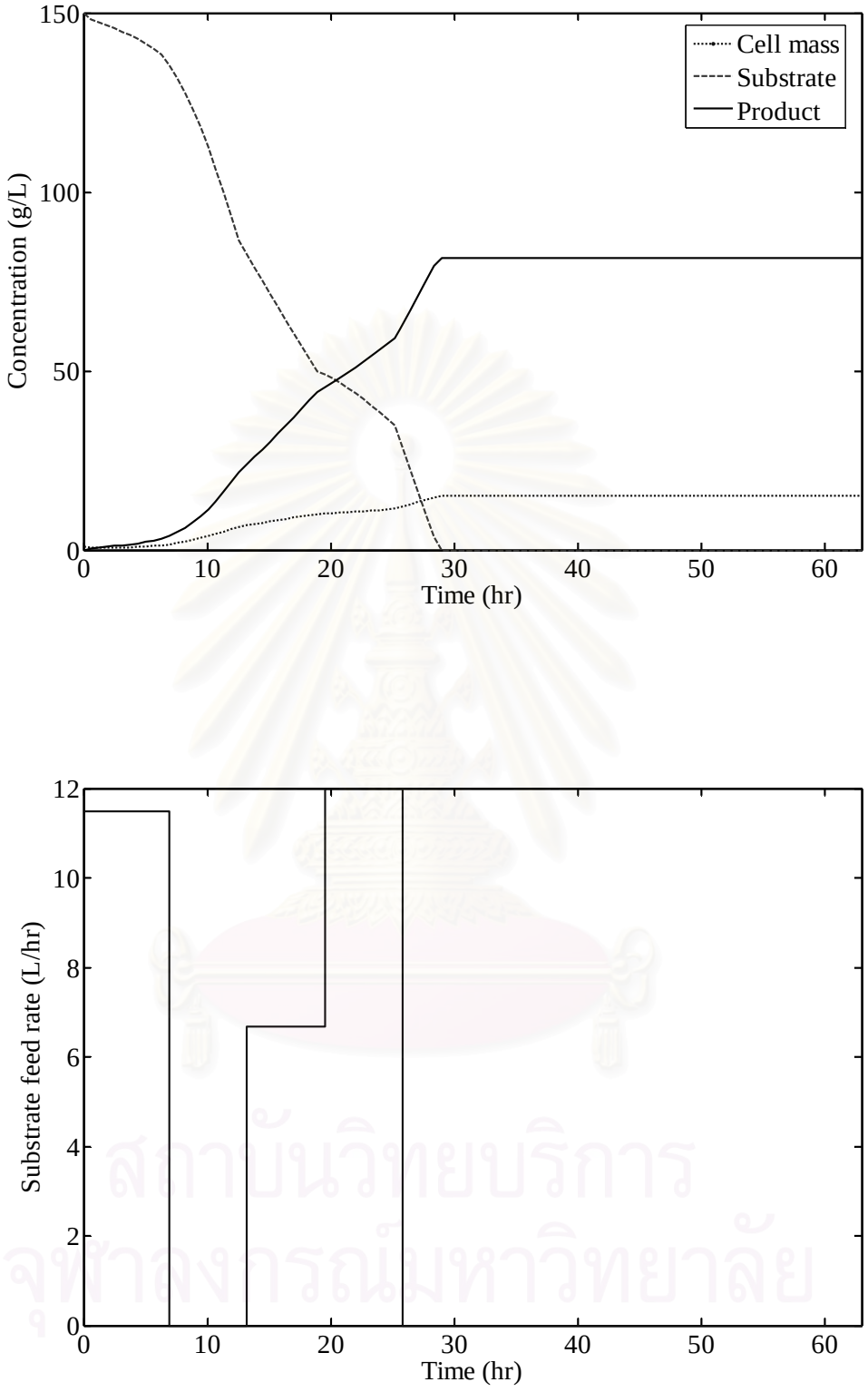

Figure 5.7 Control response of an optimal control for disturbance rejection with 20% decrease of the substrate feed concentration

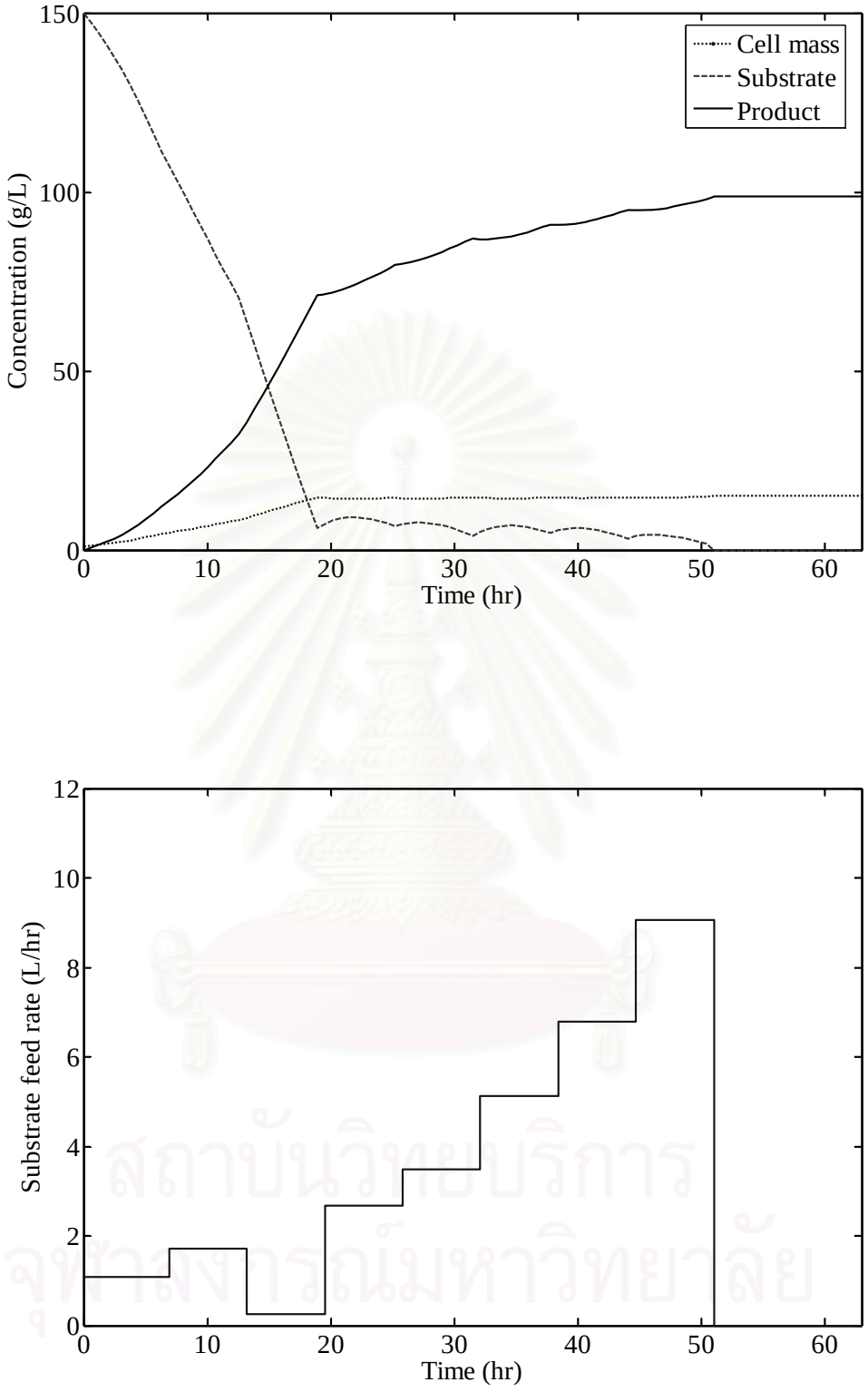

Figure 5.8 Control response of an on-line optimal control with neural network estimator for the case of model-plant mismatch with 20% increase of the yield coefficient

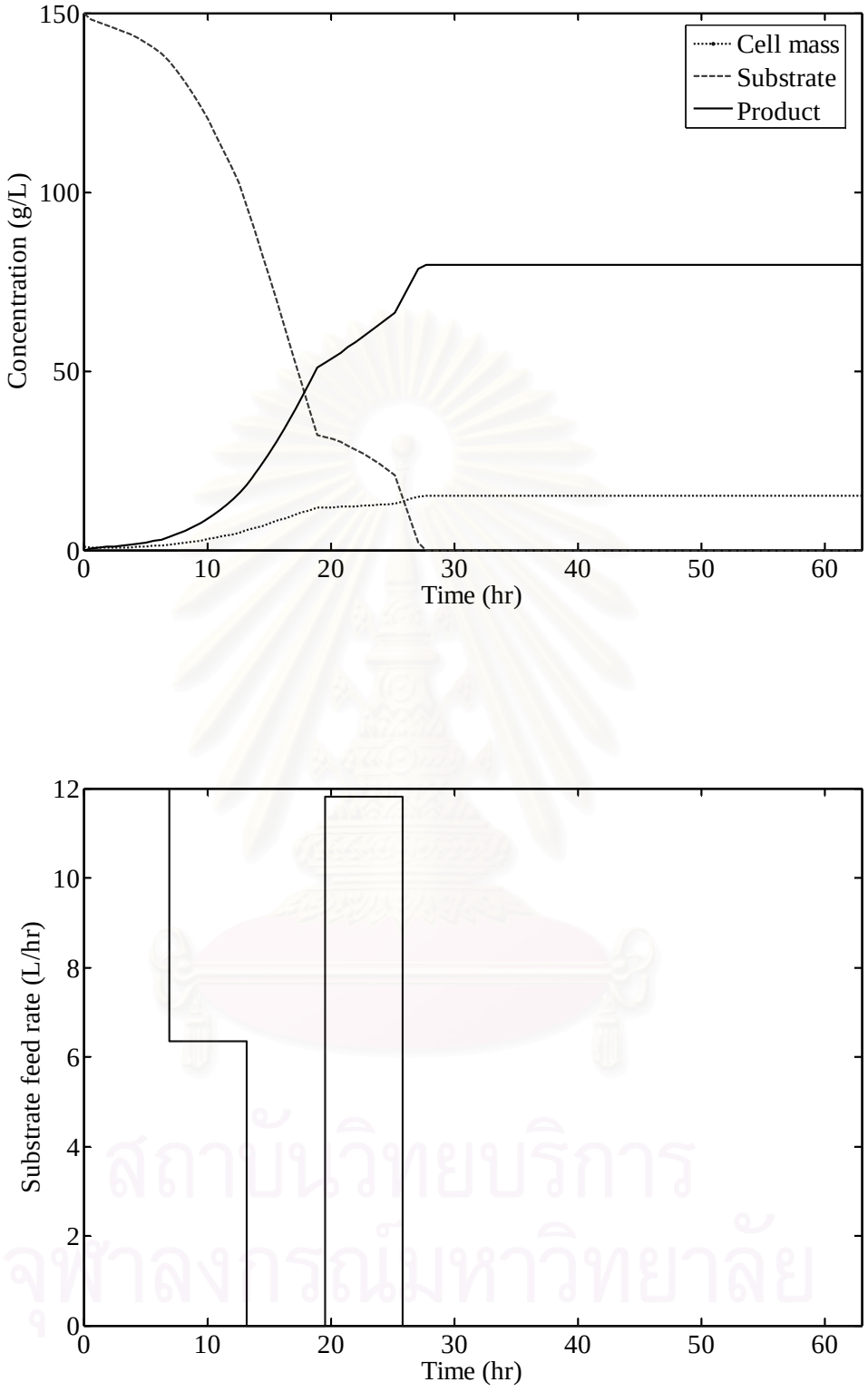

Figure 5.9 Control response of an optimal control for the case of model-plant mismatch with 20% increase of the yield coefficient

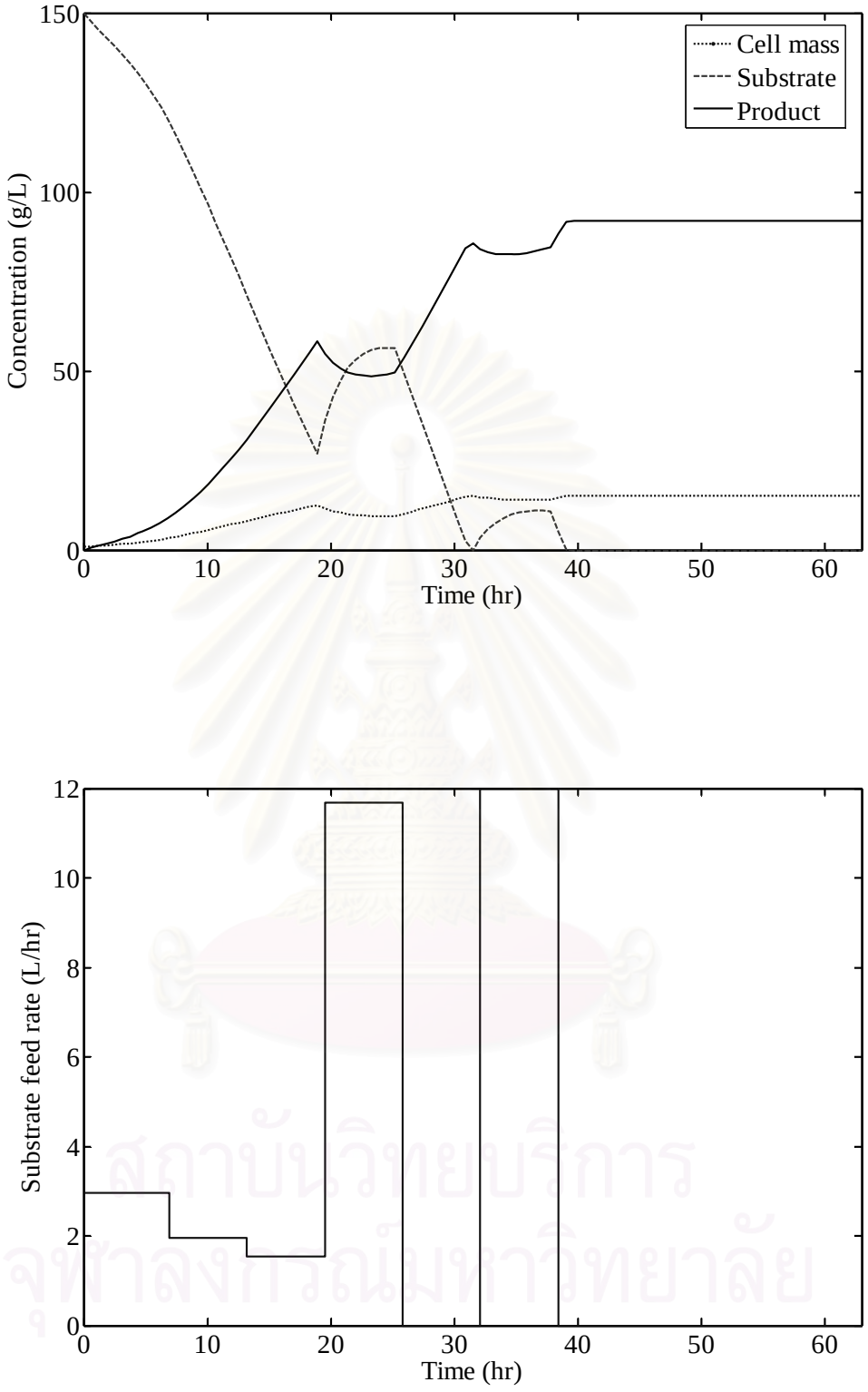

Figure 5.10 Control response of an on-line optimal control with neural network estimator for the case of model-plant mismatch with 20% decrease of the yield coefficient

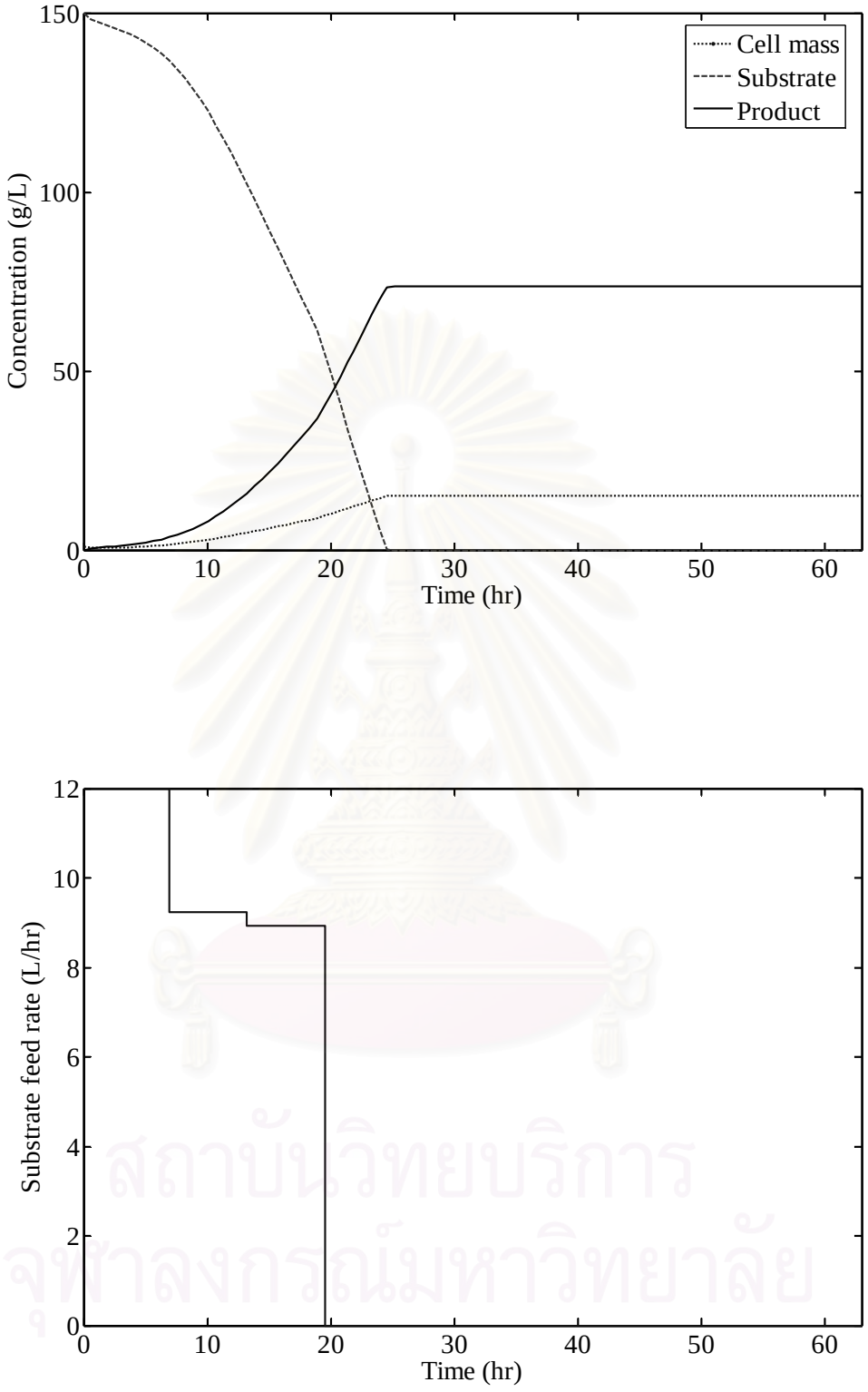

Figure 5.11 Control response of an optimal control for the case of model-plant mismatch with 20% decrease of the yield coefficient

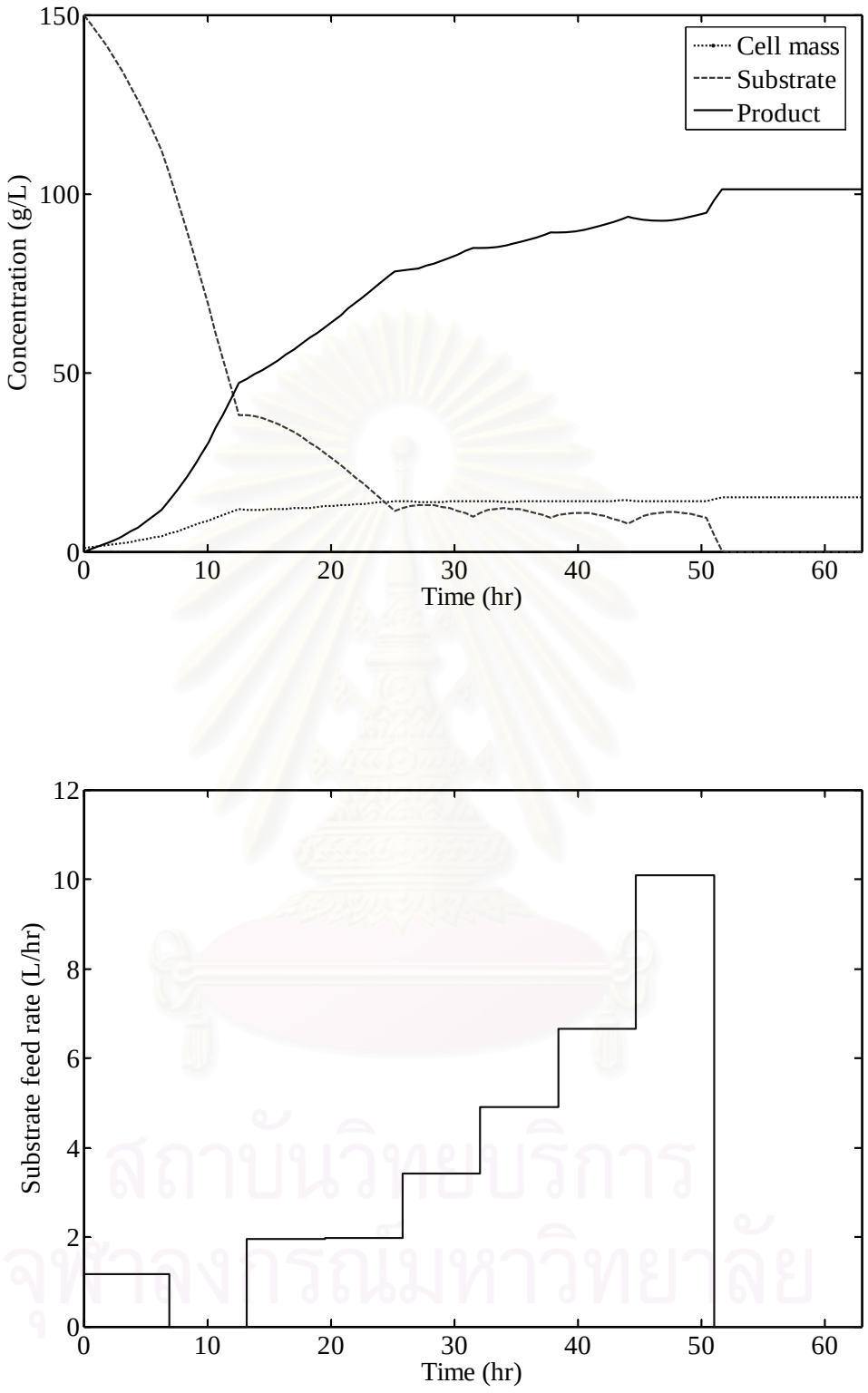

Figure 5.12 Control response of an on-line optimal control with neural network estimator for the case of model-plant mismatch with 20% increase of the maximum production rate

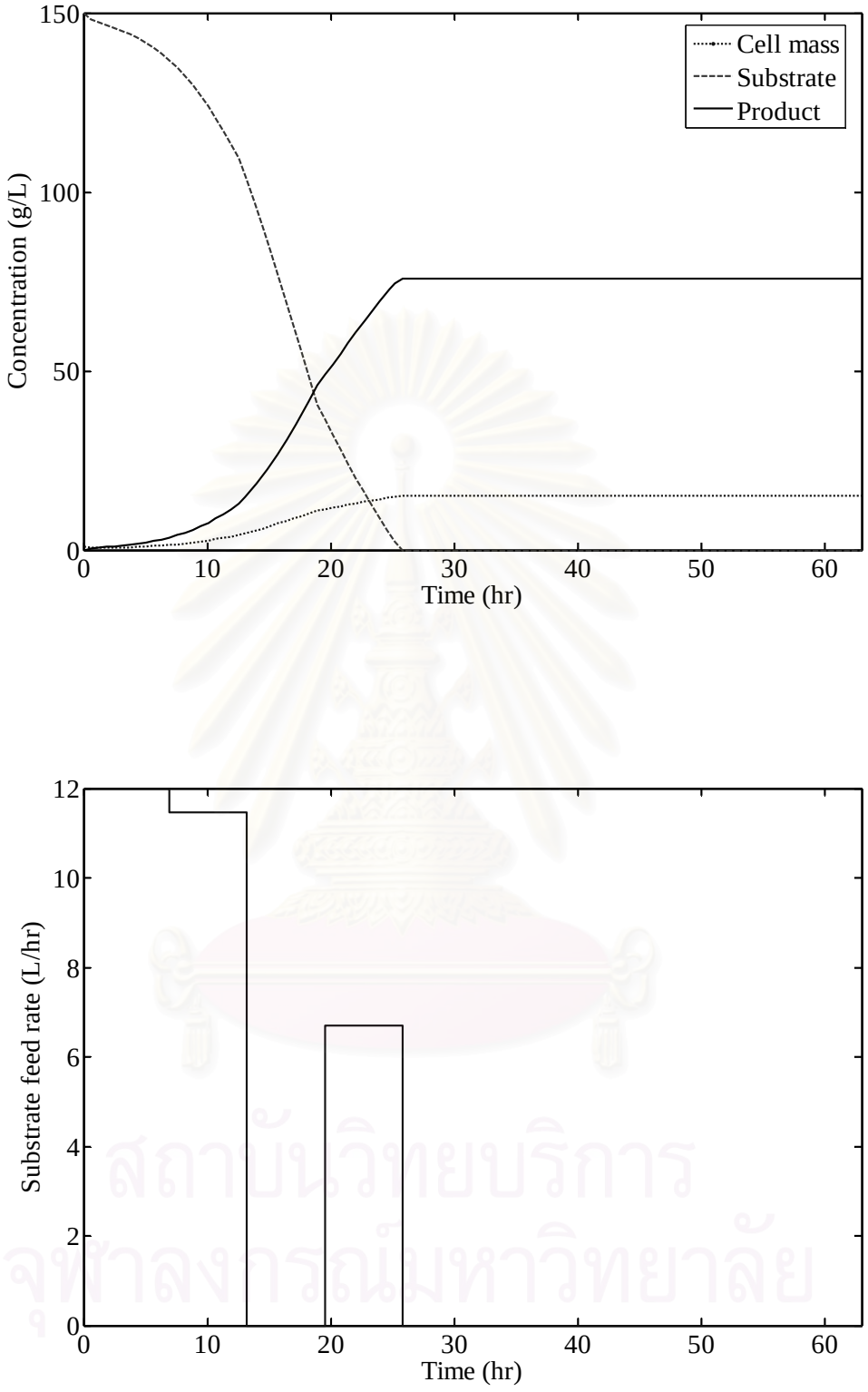

Figure 5.13 Control response of an optimal control for the case of model-plant mismatch with 20% increase of the maximum production rate

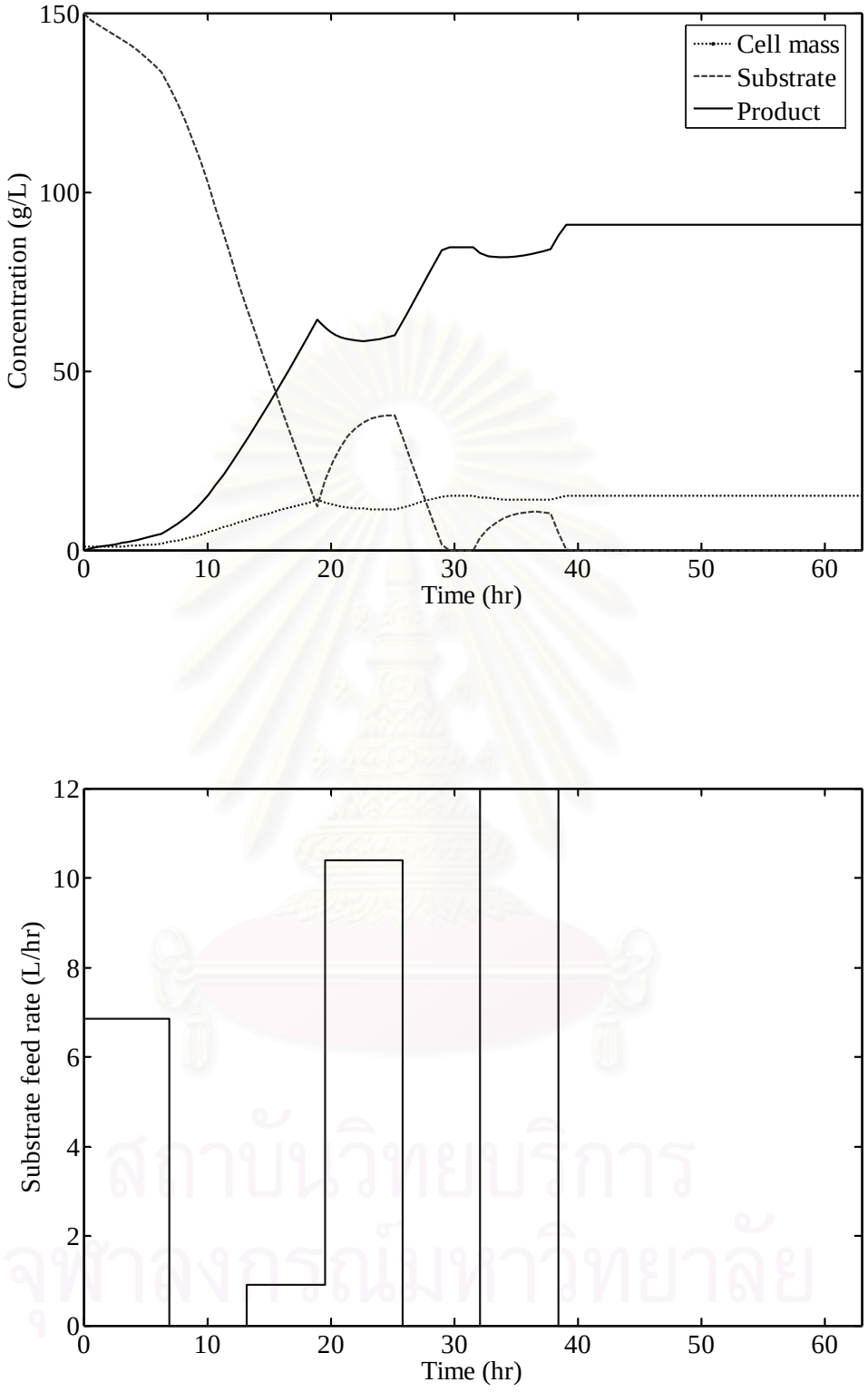

Figure 5.14 Control response of an on-line optimal control with neural network estimator for the case of model-plant mismatch with 20% decrease of the maximum production rate

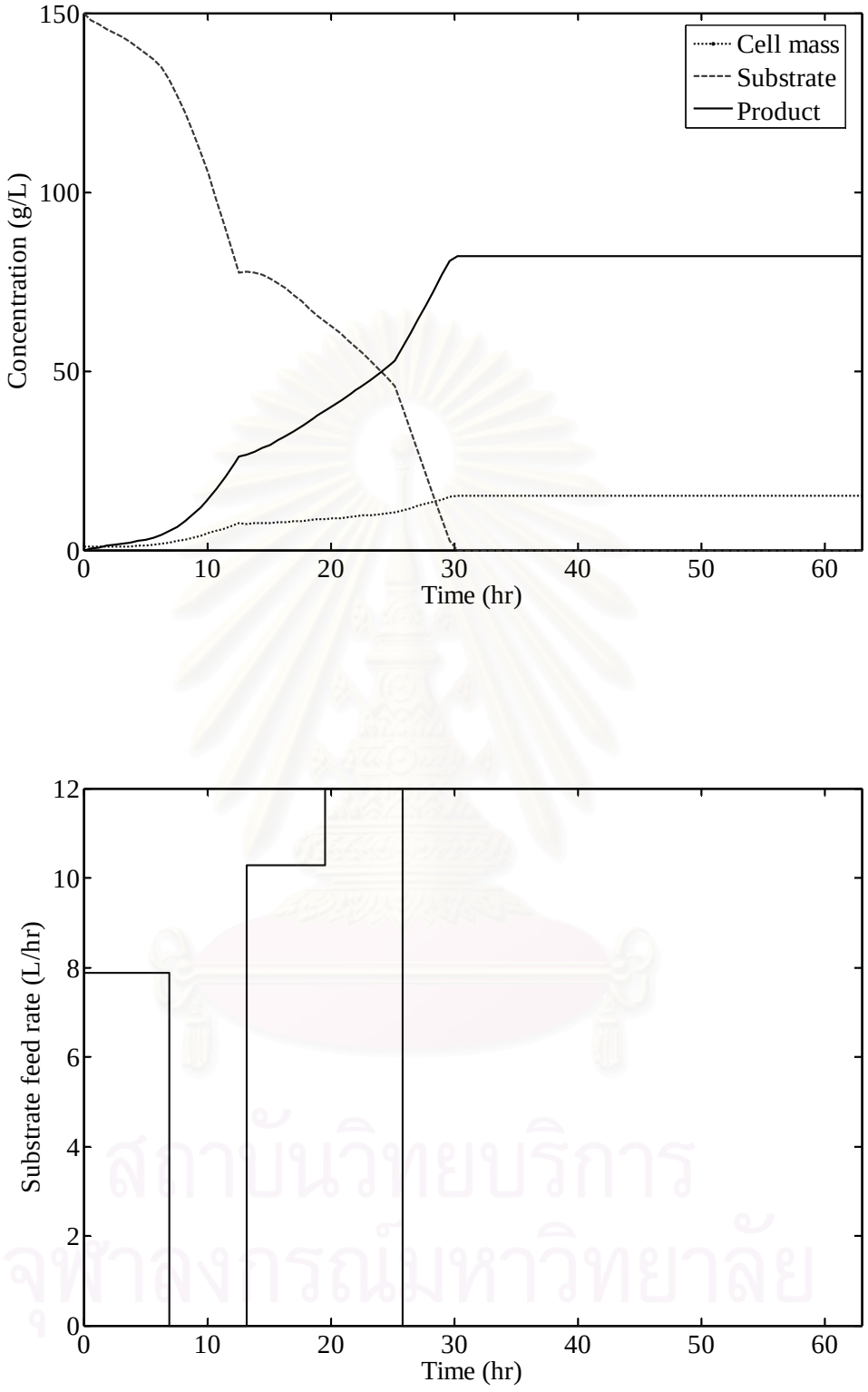

Figure 5.15 Control response of an optimal control for the case of model-plant mismatch with 20% decrease of the maximum production rate

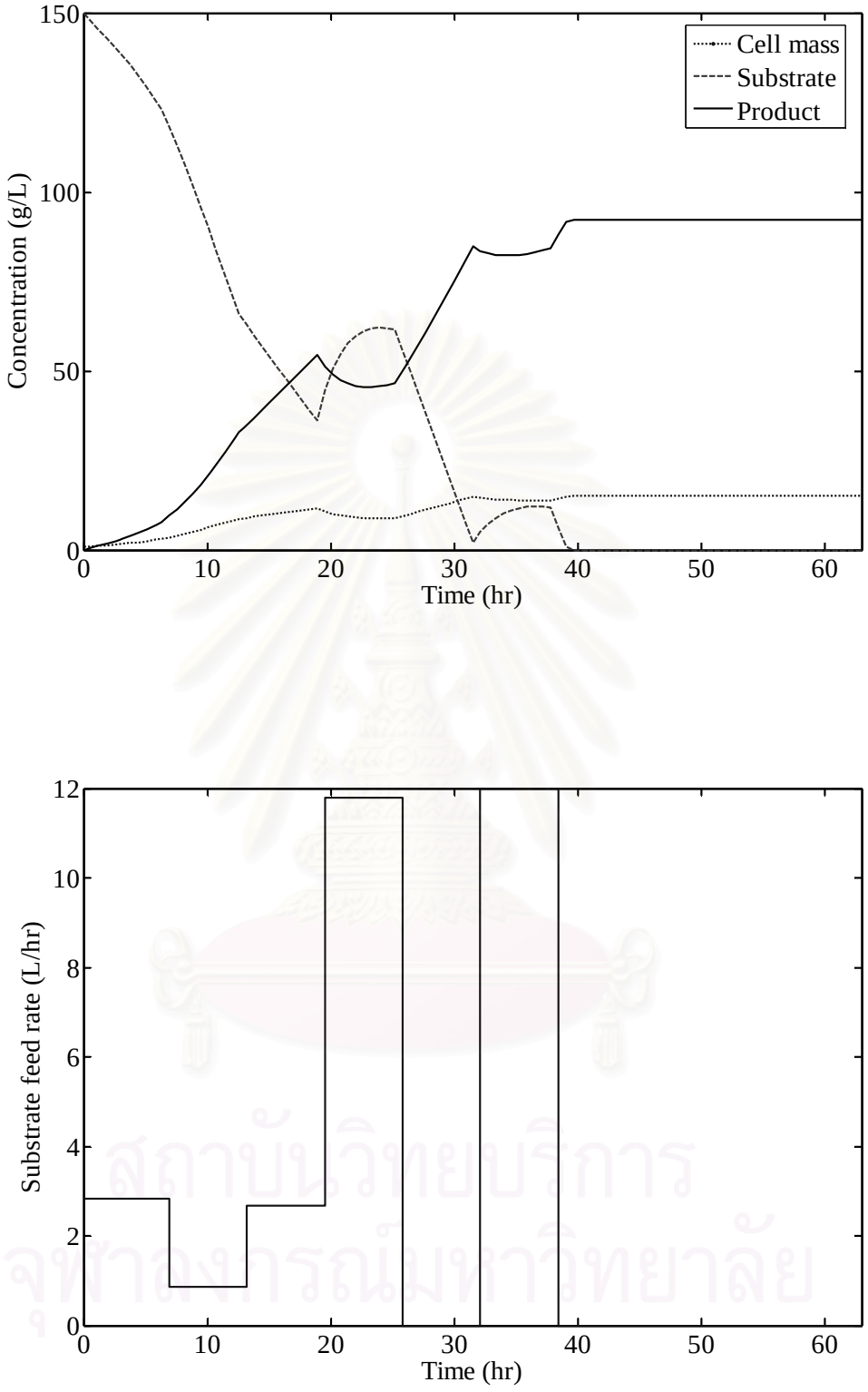

Figure 5.16 Control response of an on-line optimal control with neural network estimator for the case of model-plant mismatch with 20% increase of the maximum growth rate

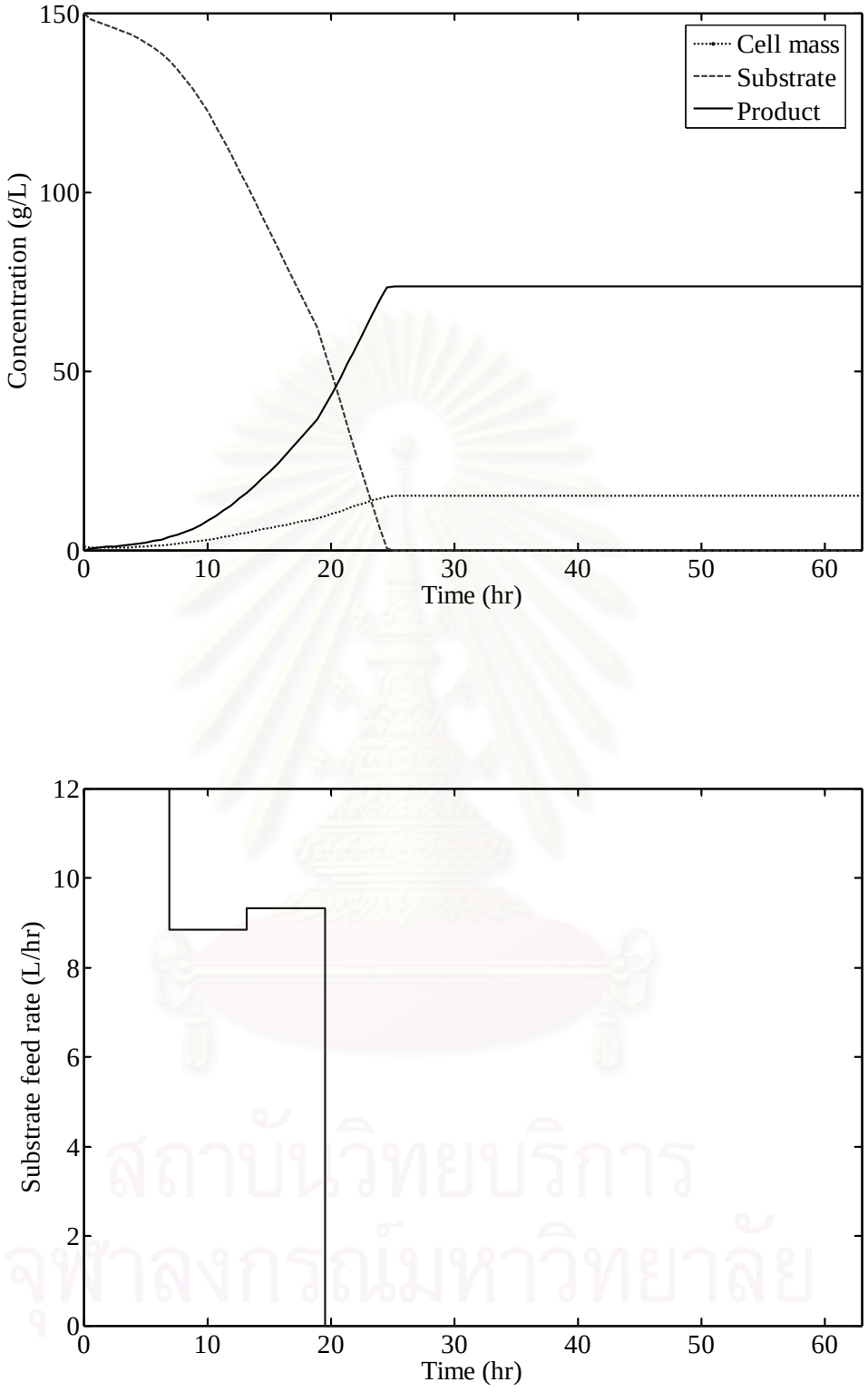

Figure 5.17 Control response of an optimal control for the case of model-plant mismatch with 20% increase of the maximum growth rate

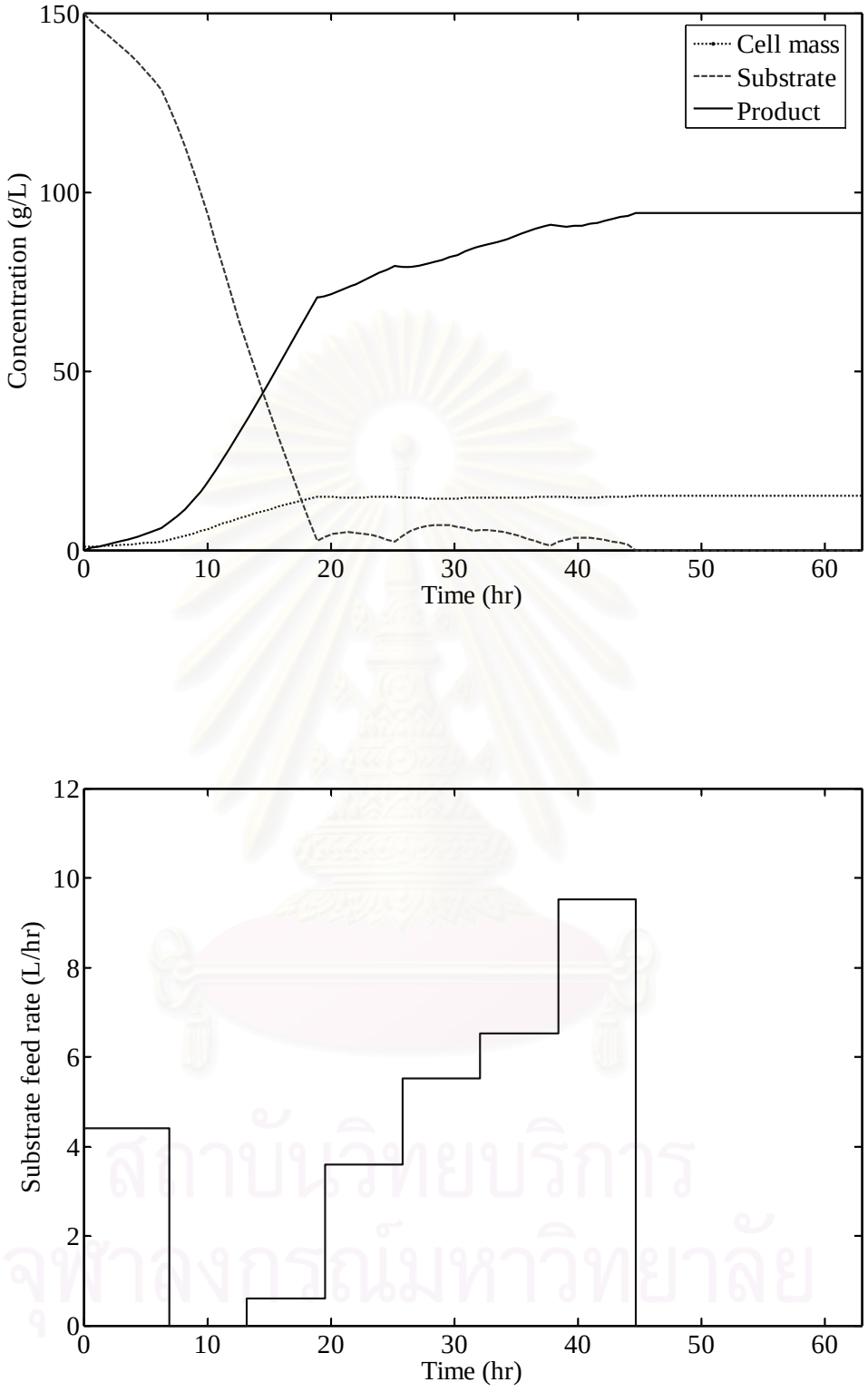

Figure 5.18 Control response of an on-line optimal control with neural network estimator for the case of model-plant mismatch with 20% decrease of the maximum growth rate

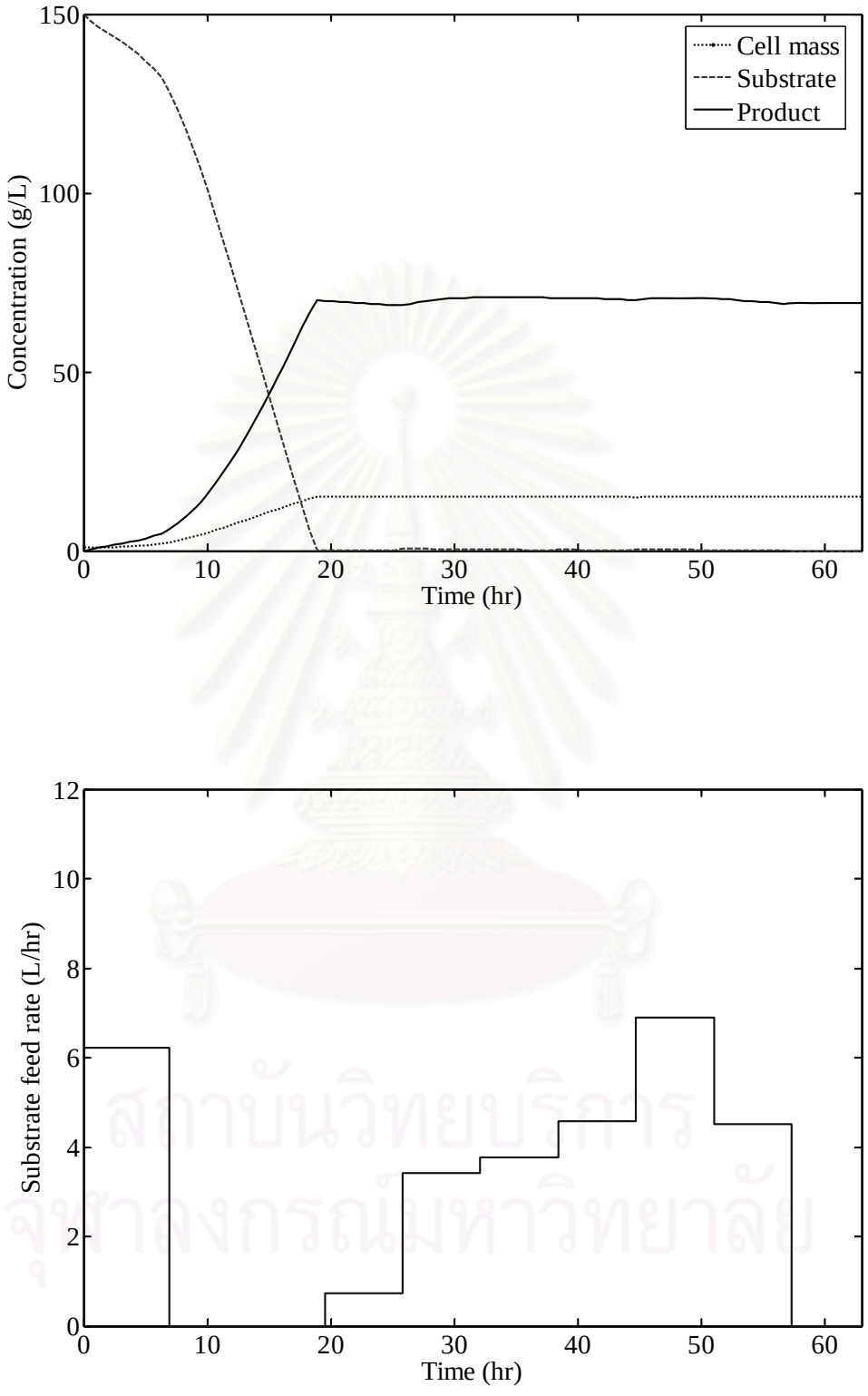

Figure 5.19 Control response of an optimal control for the case of model-plant mismatch with 20% decrease of the maximum growth rate

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

In this research, the control of the ethanol production process in a fed-batch reactor has been studied. The optimal control strategy is applied to obtain the maximum amount of the desired ethanol product at the end of the operation. Due to the existence of unknown disturbances and model-plant mismatches, the off-line calculated optimal profile may not give the optimal performance when applied to the actual process. To realize this fact, the on-line optimal control with neural network estimator is proposed to modify the substrate feed rate profile for improving the control performance.

In the optimal control strategy, the solution of the optimal control problem is computed using a sequential model solution and optimization method. The effects of a time interval and switching time are investigated in order to improve the control performance in term of the amount of the desired product. From the simulation results, it can be seen that decreasing the time interval and using the non-uniform control vector parameterization (free switching time) provide a better control performance compared to the nominal case. Furthermore, the minimum operation time is also studied in order to improve the operation in this process.

Artificial neural network is applied to estimate the substrate concentration which is assumed to be an unmeasured variable of this process. A multilayer feed-forward neural network is trained by Levenberg-Marquardt Backpropagation algorithm. The optimum topology of an artificial neural network (8-5-5-1) is employed as a neural network estimator.

The on-line optimal control with neural network estimator is proposed for deal with the unknown disturbance and the model-plant mismatch which are particularly important in the model-based control problem. The simulation results demonstrate that the on-line optimal control with neural network estimator can improve the control performance compared with the optimal control strategy in the case of disturbance rejection and the model-plant mismatch.

6.2 Recommendations

For the future direction, the proposed on-line optimal control with neural network estimator will be applied to the complicated system such as the complex fermentation and polymerization process. Furthermore, the neural network based estimator for multiple-input and multiple-output (MIMO) is also considered in the future research.

REFERENCES

Arpornwichanop, A., Kittiisupakorn, P. and Mujtaba, I. M. On-line dynamic optimization and control strategy for improving the performance of batch reactors. Chemical Engineering and Processing. 44 (2004): 101-114.

Aziz, N., Hussian, M. A. and Mujtaba, I. M. Performance of different types of controllers in tracking optimal temperature profiles in batch reactors. Comput. Chem. Eng. 24 (2000): 1069-1075.

Aziz, N. and Mujtaba, I. M. Optimal operation policies in batch reactors. Chemical Engineering Journal. 85 (2002): 313-325.

Banga, J. R., Alonso, A. A. and Singh, R. P. Stochastic Dynamic Optimization of Batch and Semi continuous Bioprocesses. Biotechnol. Prog. 13 (1997): 326-335.

Banga, J. R. and Carrasco, E. F. Dynamic optimization of batch reactors using adaptive stochastic algorithms. Ind. Eng. Chem. Res. 36 (1997): 2252-2261.

Banga, J. R., Canto, E. B., Moles, C. G. and Alonso, A. A. Dynamic optimization of bioprocesses: Efficient and robust numerical strategies. Journal of Biotechnology. 117 (2005): 407-419.

Basheer, I. A. and Hajmeer, M. Artificial neural network: fundamentals, computing, design, and application. Journal of Microbiological Methods. 43 (2003): 3-31.

Biegler, L. T. Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. Comput. Chem. Eng. 8 (1984): 243-248.

Breusegem, V. V., Thibault, J. and Cheruy, A. Adaptive neural models for on-line prediction in fermentation. Can. J. Chem. Eng. 69 (1991): 481-487.

Bryson, A. E. and Ho, Y. C. Applied Optimal Control. Washington DC: Hemisphere, 1975.

Chen, C. T. and Hwang, C. Optimal on-off control for fed-batch fermentation processes. Ind. Eng. Chem. Res. 29 (1990): 1869-1875.

Chen, C. T. and Hwang, C. Optimal control computation for differential-algebraic process systems with general constraints. Chem. Eng. Commun. 97 (1990): 9-26.

Cunha, C., Glassey, J., Montague, G., Albert, S. and Mohan, P. An assessment of seed quality and its influence on productivity estimation in an industrial antibiotic fermentation. Biotechnol Bioeng. 78 (2001): 659-669.

Cuthrell, J. E. and Biegler, L. T. Simultaneous optimization and solution methods for batch reactor control profiles. Comput. Chem. Eng. 13 (1989): 49-62.

Edgar, T. F., Himmelblau, D. M. and Lasdon, L. S. Optimization of chemical processes. 2nd edition. Singapore: McGraw-Hill, 2001.

Hagan, M. T., Demuth, H. B. and Beale M. H. Neural Network Design. Boston: PWS Publishing, 1996.

Hong, J. Optimal substrate feeding policy for a fed-batch fermentation with substrate and product inhibition kinetics. Biotechnol. Bioeng. 28 (1986): 1421-1431.

Hussain, M. A. Review of the applications of neural networks in chemical process control-simulation and on-line implementation. Artificial Intelligence in Engineering. 13 (1999): 55-68.

Kapadi, M. D. and Gudi, R. D. Optimal control of fed-batch fermentation involving multiple feeds using differential evolution. Process Biochemistry. 39 (2004): 1709-1721.

Komives, C. and Parker, R. S. Bioreactor state estimation and control. Biotechnology. 14 (2003): 468-474.

Lanouette, R., Thibault, J. and Jacques, L. Process modeling with neural networks using small experimental datasets. Comp. Chem. Engng. 23 (1999): 1167-1176.

Lee, J., Lee, S. Y., Park, S. and Middelberg, A. P. J. Control of fed-batch fermentations. Biotechnology Advances. 17 (1999): 29-48.

Lou, K. N. and Perez, R. A. A new system identification technique using kalman filtering and multilayer neural networks. Artificial Intelligence in Engineering. 10 (1996): 1-8.

Luus, R. Application of dynamic programming to differential-algebraic process systems. Comput. Chem. Eng. 17 (1993): 373-377.

Mahadevan, R. and Doyle III, F. J. On-line Optimization of Recombinant Product in a Fed-batch Bioreactor. Biotechnol. Prog. 19 (2003): 639-646.

Massimo, C. D., Montague, G. A., Willis, M. A., Tham, M. T. and Morris, A. J. Towards improved penicillin fermentation via artificial neural networks. Comput. Chem. Eng. 16 (1992): 283-291.

Modak, J. M. and Lim, H. C. Feedback optimization of fed-batch fermentation. Biotech. Bioeng. 30 (1987): 528-540.

Morari, M. and Lee, J. H. Model predictive control: past, present and future. Computers and Chemical Engineering. 23 (1999): 667-682.

Neeleman, R. and Van Boxtel, A. Estimation of specific growth rate from cell density measurements. Bioprocess Biosystems Eng. 24 (2001): 179-185.

Palanki, S., Kravaris, C. and Wang, H. Y. Synthesis of state feedback laws for end-point optimization in batch processes. Chem. Eng. Sci. 48 (1993): 135-152.

Palanki, S. and Vemuri, J. Optimal Operation of Semi-Batch Processes with a Single Reaction. Journal of Chemical Reactor Engineering. 3 (2005): 1-23.

Psichogios, D. and Ungar, L. H. A hybrid neural network-first principles approach to process modeling. A.I.Ch.E.Journal. 38 (1992): 1499-1511.

Pushpavanam, S., Rao, S. and Khan, I. Optimization of a biochemical fed-batch reactor using sequential quadratic programming. Ind. Eng. Chem. Res. 38 (1999): 1998-2004.

Ramchandran, S. and Rhinehart, R. R. A very simple structure for neural network control of distillation. Journal of Process Control. 5 (1995): 115-128.

Ramirez, W. F. Optimal state and parameter estimation and application to batch fermentation. Chem. Eng. Sci. 42 (1987): 2749-2756.

Rani, K. Y. and Rao, V. S. R. Control of fermenters - a review. Bioprocess Engineering. 21 (1999): 77-88.

Renfro, J. G., Morshedi, A. M. and Asbjornsen, O. A. Simultaneous optimization and solution of systems described by differential/algebraic equations. Comput. Chem. Eng. 11 (1987): 503-517.

Riascos, C. A. M. and Pinto, J. M. Optimal control of bioreactors: a simultaneous approach for complex systems. Chemical Engineering Journal. 99 (2004): 23-34.

Sargent, R. W. H. Optimal control. Journal of Computational and Applied Mathematics. 124 (2000): 361-371.

Sarkar, D. and Modak, J. M. Optimization of fed-batch bioreactors using genetic algorithm: multiple control variables. Computers and Chemical Engineering. 28 (2004): 789-798.

Shene, C., Diez, C. and Bravo, S. Neural networks for the prediction of the state of *Zymomonas mobilis* CP4 batch fermentations. Computers and Chemical Engineering. 23 (1999): 1097-1108.

Smets, I. Y., Claes, J. E., November, E. J., Bastin, G. P. and Van Impe, J. F. Optimal adaptive control of (bio)chemical reactor: past, present and future. Journal of Process Control. 14 (2004): 795-805.

Soni, A. S. and Parker, R. S. Closed-loop control of fed-batch bioreactor: A shrinking-horizon approach. Ind. Eng. Chem. Res. 43 (2004): 3318-3393.

Srinivasan, B., Palanki, D. and Bonvin, D. Dynamic optimization of batch processes I Characterization of the nominal solution. Computer and Chemical Engineering. 27 (2003): 1-26.

Tanvir, M. S. and Mujtaba, I. M. Neural network based correlations for estimating temperature elevation for sea water in MSF desalination process. Desalination. 195 (2006): 251-272.

Thibault, J., Breusegem, V. V. and Cheruy, A. On-line prediction of fermentation variables using neural networks. Biotechnol. Bioeng. 36 (1990): 1041-1048.

Wang, F. S. and Shyu, C. S. Optimal feed policy for fed-batch fermentation of ethanol production by Zymomous mobilis. Bioprocess Eng. 17 (1997): 63-68.

Xiong, Z. and Zhang, J. Modelling and optimal control of fed-batch processes using a novel control affine feedforward neural network. Neurocomputing. 61 (2004): 317-337.

Xiong, Z. and Zhang, J. Neural network model-based on-line re-optimization control of fed-batch processes using a modified iterative dynamic programming algorithm. Chemical Engineering and Processing. 44 (2005): 477-484.

Zuo, K. and Wu, W. T. Semi-realtime optimization and control of a fed-batch fermentation system. Computer and Chemical Engineering. 24 (2000): 1105-1109.

APPENDICES

APPENDIX A

LEVENBERG-MARQUARDT BACKPROPAGATION ALGORITHM

A.1 Backpropagation Learning Algorithm

Backpropagation is the most widely used learning algorithm in an artificial neural network. In this algorithm, the error between neural network predicted output and the actual target is propagated backward from the output layer to the hidden layers and finally to the input layer. The weights and biases are changed in the direction of minimizing the prediction error.

For the multilayer feed-forward neural networks, the output of the first layer becomes the input of the following layer. The equations that describe of this operation are showed as the follows.

$$a^{m+1} = f^{m+1}(W^{m+1} a^{m} + b^{m+1}) \qquad \text{for } m = 0, 1, \ldots, M-1 \qquad \text{(A.1)}$$

where M is the number of layers of the network. The neurons in the first layer receive external inputs:

$$a^{0} = p \qquad \text{(A.2)}$$

The outputs of the neurons in the last layer of the network are considered as the network outputs:

$$a = a^{M} \qquad \text{(A.3)}$$

A.1.1 Performance Index

The backpropagation algorithm for multilayer feed-forward neural networks use the mean square error as a criterion which is shown in Eq. (A.4)

$$\begin{aligned} F(x) &= E[e^2] = E[(t-a)^2] \\ &= E[(t-a)^T (t-a)] \end{aligned} \tag{A.4}$$

where x is the vector of network weights and biases, t and a are the corresponding target output and actual output respectively. The steepest descent algorithm for the approximate mean square error is shown as the follows:

$$w_{i,j}^m(k+1) = w_{i,j}^m(k) - \alpha \frac{\partial \overline{F}}{\partial w_{i,j}^m} \tag{A.5}$$

$$b_i^m(k+1) = b_i^m(k) - \alpha \frac{\partial \overline{F}}{\partial b_i^m} \tag{A.6}$$

where α is the learning rate.

From Eqs. (A.5) and (A.6), the error is an indirect function of the weights in the hidden layers. Therefore, the chain rule is used to determine the error gradient as the following

$$\frac{\partial \overline{F}}{\partial w_{i,j}^m} = \frac{\partial \overline{F}}{\partial n_i^m} \times \frac{\partial n_i^m}{\partial w_{i,j}^m} \tag{A.7}$$

$$\frac{\partial \overline{F}}{\partial b_i^m} = \frac{\partial \overline{F}}{\partial n_i^m} \times \frac{\partial n_i^m}{\partial b_i^m} \tag{A.8}$$

The second term in each of these equations can be easily computed, since the net input to layer m is an explicit function of the weights and biases in that layer:

$$n_i^m = \sum_{j=1}^{S^{(m-1)}} w_{i,j}^m a_j^{m-1} + b_i^m \tag{A.9}$$

Therefore,

$$\frac{\partial n_i^m}{\partial w_{i,j}^m} = a_j^{m-1}, \quad \frac{\partial n_i^m}{\partial b_i^m} = 1 \tag{A.10}$$

Define

$$s_i^m = \frac{\partial \overline{F}}{\partial n_i^m} \tag{A.11}$$

which is the sensitivity of $\overline{F}$ to changes in the *i*th element of the net input at layer *m*, then Eqs. (A.7) and (A.8) can be simplified to

$$\frac{\partial \overline{F}}{\partial w_{i,j}^m} = s_i^m a_j^{m-1} \tag{A.12}$$

$$\frac{\partial \overline{F}}{\partial b_i^m} = s_i^m \tag{A.13}$$

Therefore, the approximate steepest descent algorithm is expressed as the follows.

$$w_{i,j}^m(k+1) = w_{i,j}^m(k) - \alpha s_i^m a_j^{m-1} \tag{A.14}$$

$$b_i^m(k+1) = b_i^m(k) - \alpha s_i^m \tag{A.15}$$

A.2 Training Function

A.2.1 Conjugate Gradient Method

The basic backpropagation method adjusts the weights in the steepest descent direction which the performance function rapidly decreases in the direction of negative of the gradient. It turns out that, although the function decreases most rapidly along the negative of the gradient, this does not necessarily produce the fastest convergence.

In the conjugate gradient method, a search is performed along the conjugate direction that produces generally faster convergence than steepest descent directions. All of the conjugate gradient methods begin with searching in the steepest descent direction (negative of the gradient) on the first iteration.

$$s^0 = -\nabla f(x^0) \tag{A.16}$$

A line search is then performed to determine the optimal distance along the current search direction as the follow.

$$x^1 = x^0 + \alpha^0 s^0 \tag{A.17}$$

Then, the next search direction is determined so that it is conjugate to various search directions. The general procedure for determining the new search direction is to combine the new steepest descent direction with the previous search directions.

$$s^1 = -\nabla f(x^1) + s^0 \frac{\nabla' f(x^1)\nabla f(x^1)}{\nabla' f(x^0)\nabla f(x^0)} \tag{A.18}$$

For the k^{th} iteration, the relation in Eq. A.18 is shown as the follow.

$$s^{k+1} = -\nabla f(x^{k+1}) + s^k \frac{\nabla' f(x^{k+1})\nabla f(x^{k+1})}{\nabla' f(x^k)\nabla f(x^k)} \tag{A.19}$$

A.2.2 Newton's Method

Newton's method is an alternative to the conjugate gradient method for fast optimization. The basic step of Newton's method is

$$x^{k+1} - x^k = -\left[\mathrm{H}(x^k)\right]^{-1} \nabla f(x^k) \tag{A.20}$$

where $\mathrm{H}(x^k)$ is the Hessian matrix (second derivatives) of the performance index at the current values of the weights and biases. Newton's method often converges faster

than the conjugate gradient method. Unfortunately, it is complex to compute the Hessian matrix for the feedforward neural networks.

A.2.3 Levenberg-Marquardt Method

The Levenberg-Marquardt method was designed to approach second order training speed without having the computing of the Hessian matrix. When the performance function has the form of a sum of squares which is typical in training feedforward networks, then the Hessian matrix can be approximated as

$$H = J' J \tag{A.21}$$

and the gradient can be computed as:

$$\nabla f = J' e \tag{A.22}$$

where J is the Jacobian matrix, which contain the first derivatives of the network errors with respect to the weights and biases, and e is a vector of network errors. The Jacobian matrix can be computed through a standard backpropagation technique that is much less complex than the computing of the Hessian matrix.

The Levenberg-Marquardt method uses this approximation to the Hessian matrix in the following Newton like update.

$$x^{k+1} = x^k - \left[J' J + \mu I\right]^{-1} J' e \tag{A.23}$$

When the scalar μ is zero, this is just a Newton's method using the approximate Hessian matrix. When μ is large, this becomes gradient descent with a small step size. Newton's method is faster and more accurate near an error minimum, so the aim is to shift towards Newton's method as quickly as possible. Thus, μ is decreased after each successful step (reduction in performance function) and is increased only when a tentative step would increase the performance function. In this way, the performance function will always be reduced at each of iteration in the algorithm.

APPENDIX B

NEURAL NETWORK ESTIMATOR

Inputs of Neural Network Estimator

$$a_1^1 = \frac{x(t) - 11.306}{5.106}$$

$$a_2^1 = \frac{p(t) - 55.955}{29.439}$$

$$a_3^1 = \frac{v(t) - 199.440}{111.920}$$

$$a_4^1 = \frac{u(t) - 5.999}{3.508}$$

$$a_5^1 = \frac{x(t-1) - 11.302}{5.109}$$

$$a_6^1 = \frac{p(t-1) - 55.932}{29.450}$$

$$a_7^1 = \frac{v(t-1) - 199.340}{111.930}$$

$$a_8^1 = \frac{u(t-1) - 5.998}{3.509}$$

Outputs from the 1st Hidden layer

$$a_1^2 = 1 / (1+\exp(-(+1.4195* a_1^1 -0.0687* a_2^1 -0.0299* a_3^1 -0.0001* a_4^1 -0.0284* a_5^1 +0.0794* a_6^1 +0.0265* a_7^1 +0.0004* a_8^1 +0.4113)))$$

$$a_2^2 = 1 / (1+\exp(-(+1.4177* a_1^1 -6.7633* a_2^1 - 0.0510* a_3^1 +0.0004* a_4^1 +0.0008* a_5^1 +0.0295* a_6^1 +0.0084* a_7^1 +0.0015* a_8^1 -10.9690)))$$

$$a_3^2 = 1 / (1+\exp(-(-2.1423* a_1^1 -0.5977* a_2^1 +3.3221* a_3^1 +0.0552* a_4^1 -5.5415* a_5^1 -2.2685* a_6^1 -1.6976* a_7^1 -0.0756* a_8^1 -0.7745)))$$

$$a_4^2 = 1 / (1+\exp(-(-1.4638* a_1^1 -0.2673* a_2^1 +2.5690* a_3^1 +0.0278* a_4^1 -2.9306* a_5^1 -1.3661* a_6^1 -1.8356* a_7^1 -0.0501* a_8^1 -0.5188)))$$

$$a_5^2 = 1 / (1+\exp(-(-2.7440* a_1^1 -0.4310* a_2^1 -0.4392* a_3^1 -0.0078* a_4^1 +0.1563* a_5^1 +1.0148* a_6^1 -0.2332* a_7^1 -0.0139* a_8^1 -1.6070)))$$

Outputs from the 2nd Hidden layer

$$a_1^3 = 1 / (1+\exp(-(-5.0598* a_1^2 +3.3669* a_2^2 +0.4493* a_3^2 -0.9277* a_4^2 +0.4949* a_5^2 +5.2857)))$$

$$a_2^3 = 1 / (1+\exp(-(-3.0971* a_1^2 -6.3727* a_2^2 +0.3162* a_3^2 +1.5130* a_4^2 -5.0696* a_5^2 +4.9311)))$$

$$a_3^3 = 1 / (1+\exp(-(+4.1235* a_1^2 +3.8922* a_2^2 -1.1180* a_3^2 -4.2215* a_4^2 +4.6330* a_5^2 -4.2589)))$$

$$a_4^3 = 1 / (1+\exp(-(+6.5760* a_1^2 +0.6943* a_2^2 -1.1201* a_3^2 -1.1562* a_4^2 +0.9267* a_5^2 +1.3860)))$$

$$a_5^3 = 1 / (1+\exp(-(+4.6893* a_1^2 -5.7701* a_2^2 -0.5034* a_3^2 +5.2684* a_4^2 -2.5007* a_5^2 +4.2222)))$$

Output of Neural Network Estimator

$$a_1^4 = +4.4528* a_1^3 -1.3831* a_2^3 -0.7837* a_3^3 -3.1384* a_4^3 -2.7548* a_5^3 +3.2266$$

$$s_{predicted}(t) = (a_1^4 \times 52.093) + 37.978$$

VITA

Mr. Natthapong Shomchoam was born in Bangkok on January 6, 1982. He graduated the Bachelor Degree in department of Chemical Technology from Chulalongkorn University in April 2004. After that he studied for Master Degree in Chemical Engineering, Chulalongkorn University in May 2004.