การบูรณาการการเพิ่ม/ลดความร้อนและการระเหยเพื่อปรับปรุงคุณภาพผลิตภัณฑ์ในกระบวนการตกผลึกแบบแบตช์

นายเจษฎา ธัมปสาโท

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมเคมี ภาควิชาวิศวกรรมเคมี
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2554

INTEGRATION OF HEATING/COOLING AND EVAPORATION TO IMPROVE PRODUCT QUALITY IN A BATCH CRYSTALLIZATION PROCESS

Mr. Jedsada Thampasato

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 2011

Thesis Title	INTEGRATION OF HEATING/COOLING AND EVAPORATION TO IMPROVE PRODUCT QUALITY IN A BATCH CRYSTALLIZATION PROCESS
By	Mr. Jedsada Thampasato
Field of Study	Chemical Engineering
Thesis Advisor	Professor Paisan Kittisupakorn, Ph.D.

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

…… Dean of the Faculty of Engineering
(Associate Professor Boonsom Lerdhirunwong, Dr.Ing.)

THESIS COMMITTEE

…….. Chairman
(Assistance Professor Anongnat Somwangthanaroj, Ph.D.)

…….…… Thesis Advisor
(Professor Paisan Kittisupakorn, Ph.D.)

…….. Examiner
(Assistance Professor Kasidit Noothong, Ph.D.)

…….. External Examiner
(Wachira Daosud, D.Eng.)

เจษฎา ธัมปสาโท : การบูรณาการการเพิ่ม/ลดความร้อนและการระเหยเพื่อปรับปรุงคุณภาพผลิตภัณฑ์ในกระบวนการตกผลึกแบบแบตช์. (INTEGRATION OF HEATING/COOLING AND EVAPORATION TO IMPROVE PRODUCT QUALITY IN A BATCH CRYSTALLIZATION PROCESS) อ. ที่ปรึกษาวิทยานิพนธ์หลัก: ศ.ดร.ไพศาล กิตติศุภกร, 119 หน้า.

กระบวนการตกผลึกเป็นกระบวนแยกที่ถูกใช้อย่างแพร่หลายในหลายอุตสาหกรรม เพื่อที่จะผลิตผลิตภัณฑ์ที่มีความบริสุทธิ์สูง ในงานวิจัยนี้ได้นำ การออปติไมซ์พลวัต และข่ายงานนิวรัลมาประยุกต์ เพื่อที่จะปรับปรุงคุณภาพผลิตภัณฑ์ของกรดซิตริก ในส่วนของการทำออปติไมซ์ ปัญหาออปติไมซ์ที่ศึกษาได้แก่ ปัญหาผลิตภัณฑ์สูงสุด และ ปัญหาขนาดผลึกสูงสุด ในงานวิจัยนี้ ข่ายงานนิวรัลถูกออกแบบเพื่อใช้เป็นตัวทำนายค่าอุณหภูมิภายในเครื่องตกผลึก ความเข้มข้นของสารละลาย และอุณหภูมิของแจ็กเก็ต และข่ายงานนิวรัลแบบผกผันถูกพัฒนาเพื่อทำนายค่าเป้าหมายของอุณหภูมิของแจ็กเก็ต โดยการหาโครงสร้างที่เหมาะสมของข่ายงานนิวรัลนั้น ใช้เทคนิคการหาค่าที่น้อยที่สุดของค่าเฉลี่ยของความผิดพลาดยกกำลังสอง ซึ่งในการฝึกสอนข่ายงานนิวรัลจะใช้วิธีเลเวนเบิร์ก มาร์ควอร์ท ในการออกแบบตัวควบคุมนั้นได้ประยุกต์การควบคุมทำนายแบบจำลองร่วมกับข่ายงานนิวรัล และการควบคุมข่ายงานนิวรัลแบบผกผันเพื่อควบคุมอุณหภูมิภายในเครื่องตกผลึก

ผลการจำลองในส่วนของการทำออปติไมซ์นั้น ขนาดผลึกที่ได้จากปัญหาออปติไมซ์มีขนาดใหญ่กว่าวิธีการลดความร้อน 19% และการระเหย 30% อีกทั้งผลผลิตที่ได้ยังเพิ่มขึ้นมากกว่า 50% โดยข่ายงานนิวรัลที่ใช้จำลองกระบวนการแสดงผลการทำนายได้อย่างถูกต้องแม่นยำ ในส่วนของตัวควบคุมนั้น ความทนทานของตัวควบคุมได้ถูกศึกษา ในกรณีเมื่อค่าพารามิเตอร์มีความผิดพลาดเกิดขึ้น โดยผลที่ได้แสดงให้เห็นว่า ตัวควบคุมทำนายแบบจำลอง หรือ เอ็มพีซี ที่ใช้ข่ายงานนิวรัลร่วมด้วยนั้นมีประสิทธิภาพในการควบคุมสูงที่สุด

ภาควิชา วิศวกรรมเคมี ลายมือชื่อนิสิต .....

สาขาวิชา วิศวกรรมเคมี ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา 2554

5270250621 : MAJOR CHEMICAL ENGINEERING
KEYWORDS : NEURAL NETWORK / BATCH CRYSTALLIZATION / OPTIMIZATION

JEDSADA THAMPASATO : INTEGRATION OF HEATING/COOLING AND EVAPORATION TO IMPROVE PRODUCT QUALITY IN A BATCH CRYSTALLIZATION PROCESS.
ADVISOR: PROF.PAISAN KITTISUPAKORN, Ph.D., 119 pp.

Crystallization processes have been widely used for separation in many fields to provide a high purity product. In this work, dynamic optimization and neural network (NN) have been applied to improve the quality of the product: citric acid. In the dynamic optimization, optimization problems maximizing both crystal yield and crystal size have been formulated. In this work, a neural network forward model has been designed to provide estimations of crystallizer temperature, concentration of solution and jacket temperature as well as a neural network inverse model has been developed to predict jacket temperature set point. The Levenberg Marquadt algorithm has been used to train the networks and optimal neural network architectures have been determined by a mean squared error (MSE) minimization technique. In controller design, neural network direct inverse control (NNDIC) and neural network model predictive control (NNMPC) strategies have been applied to control the crystallizer temperature.

The simulation results have shown that the obtained crystal size from optimization problem is 19% and 30% larger than cooling and evaporation methods, respectively moreover yield increase more than 50%. Both neural network forward and inverse models show good accuracy for the prediction of the system. The robustness of controller is investigated with respect to parameters mismatch. The results have shown that the NNMPC controller provides superior control performance in all case studies.

Department : Chemical Engineering Student's Signature .....
Field of Study : Chemical Engineering Advisor's Signature .....
Academic Year : 2011

ACKNOWLEDGEMENTS

I would like to thank and express my sincere gratitude to my advisor, Professor Paisan Kittisupakorn, for his supervision, inspiration, encouragement, advice, discussion and helpful suggestions throughout the course of this Master Degree study. Furthermore, I also grateful thank to Assistance Professor Anongnat Somwangthanaroj as the chairman, Assistance Professor Kasidit Noothong and Dr. Wachira Daosud as the members of thesis committee.

I am deeply indebted to Dr. Wachira Daosud, Department of Chemical Engineering, Bhurapa University who has helped, suggested, emboldened and resolved about technical obstacles throughout running simulation.

I would like to thank all of my friends and colleagues in the Control and Systems Engineering Research Center, especially Chollaphan Thanomjit for their friendship and support over the years of my study.

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

CONTENTS

Page

Page

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

Page

Page

Page

NOMENCLATURE

A_j	the total heat transfer surface area (m^2)
A_s	the evaporation surface area (m^2)
b	the magma density exponent.
B_p	the primary nucleation rate (no/s.kgH_2O)
B_s	the secondary nucleation rate (no/s.kgH_2O)
C*	the solubility of the solute (kg/kgH_2O)
C_p	the heat capacity of the solution (kJ/kg $^\circ$C)
C_{pj}	the heat capacity of the water in jacket (kJ/kg $^\circ$C)
C_s	the concentration of solute (kg solute/kg H_2O)
F_j	the water flow rate in jacket (m^3/s)
g	the crystal growth rate exponent
G	the crystal growth rate (m/s)
H_{crys}	the heat of crystallization (kJ/kg)
H_{evap}	the heat of vaporization (kJ/kg)
k_g	the crystal growth rate constant (m/s.(kg/kgH_2O)^g)
k_p	the primary nucleation rate constant (m/s.(kg/kgH_2O)^p)
k_s	the secondary nucleation rate constant (m/s.(kg/kgH_2O)^s.(kg/kgH_2O)^b)
k_v	the volumetric shape factor
L	the characteristic crystal length (m)
L_0	the characteristic crystal length of a newly formed crystal (m)
m	the molecular weight (g/mol)
m_i	the i moment of crystal size (no.m^i/kg H_2O)
M	the total mass of solvent at any instant of time (kg)
M_{crys}	the magma density of crystal (kg/kgH_2O)
M_{tot}	the total mass of solution (kg)
n	the number density of the crystals (no/(kgH_2O.m))
p	primary nucleation rate exponent
p_i	the network target

P_a	the partial pressure in the crystallizer (kPa)
P_v	the saturation vapor pressure (kPa)
Q_{evap}	the evaporation rate (kg/s)
R	the gas constant (m^3.Pa/K.mol)
s	secondary nucleation rate exponent
S	the supersaturation ratio
T	the crystallizer temperature (°C)
T_j	the jacket temperature (°C)
T_{jsp}	the jacket temperature set point (°C)
U	the overall heat transfer coefficient (W/m^2 °C)
V_j	the jacket volume (m^3)
w	synaptic weight
x	input vector
y_i	the network output

GREEK SYMBOLS

ΔC	the concentration driving force (kg/kgH_2O)
σ_s	the relative supersaturation
u	value of the internal potential
ρ_j	the density of water in jacket (kg/m^3)
α	the evaporation coefficient
ρ_c	the density of crystals (kg/m^3)

SUPSCRIPTS

sp	set point
j	jacket

ACRONYM

DIC	direct inverse control
IAE	Integral absolute error
MPC	model predictive control
MSE	mean square error
NN	neural network

CHAPTER I

INTRODUCTION

1.1 Introduction

Batch crystallization which is an important separation and purification unit is widely used in many fields such as food, chemical and pharmaceutical to produce high value-added specialty chemicals with high purity, desired Crystal Size Distribution (CSD) and shape. In general, a batch cooling crystallization is one of the most method to apply in industrial processes owing to it operates easily. Cooling crystallization, supersaturation is generated by cooling, is applied to the solution which the solubility of solute depend on significantly temperature. On the other hand, evaporative crystallization, supersaturation can be created by evaporate, is more favorable than cooling crystallization if the solubility of solute depend on slightly temperature. With the object of enhancing the productivity of batch crystallization process, there are a number of researches focused on optimization and controlling both of batch crystallization by cooling and evaporation processes (Mukhopadhyay and Epstein, 1980; Neelakantan and Mukesh, 1979; Fagervik et al., 1988; Sowul and Epsteln, 1981; Fevotte et al., 1990; Mesbah et al., 2010). For examples, Paengjuntuek et al. (2008) studied the implementation of a dynamic optimization integrated with a nonlinear control strategy of batch crystallization process for the production of potassium sulfate. A generic model control (GMC) was used for control the crystallizer temperature following the desired profile and compared with conventional PI control techniques. Mesbah et al. (2010) presented a model-based control approach for optimal operation of a seeded fed-batch evaporative crystallization. The evaporative crystallization was operated at isothermal and vacuum pressure.

Moreover, combining heating/cooling and evaporation in batch crystallization is also the main topic of interest due to it can produce higher supersaturation, higher crystal yield and larger crystal size than all conventional crystallization. This method

is only suitable for the system where the solubility varies significantly with temperature (Choong and Smith, 2004). The combined evaporative crystallization with other process presents several advantages such as reduce equipment requirements, time cycle and saving cost of separation and recovery in some case. Choong and Smith (2004) optimized and compared batch heating/cooling evaporative crystallization with the other crystallizations such as batch evaporative, semi-batch evaporative and batch cooling crystallizations. The results showed that the heating/cooling evaporative crystallization provided larger average crystal size 35%, 12% and 8% than crystallization by batch cooling and unseeded constant evaporation, batch seeded evaporative and semi-batch evaporative, respectively. In addition, heating/cooling evaporative crystallization increased the crystal yield to 47% higher than all modes of crystallization. Nevertheless, the process control to achieve a desired product is necessary.

The batch crystallization process is high complexity and nonlinear system. One of the most effective techniques to solve this problem is an artificial neural network that is successfully applied to modeling the nonlinear system. An advantage of neural network is easy to design and use. The neural network estimates the relationship of input and output to explain the process. Kittisupakorn et al. (2005) applied neural network to predict the concentration of a hydrochloric acid in hydrochloric acid recovery process. Arpornwichanop and Shomchoam (2009) applied neural network to estimate the substrate concentration which use in optimal control in a fed-batch bioreactors. In addition, there are many researches applied neural network to modeling the nonlinear process (Nueaklong et al., 2011; Mujtaba et al., 2006; Georgieva and Azevedo, 2006; Wong et al., 2010; Charoenniyom et al., 2011).

One goal in production is the control of product properties such as crystal size, purity and crystal size distributions that influence downstream processing operations such as filtering, drying and storage. The batch crystallization process is high complexity and nonlinear system. Consequently, the control of this process is a significant challenge that neural network is successfully applied to control of nonlinear system. For instance, Damour et al (2010) presented NMPC of an industrial crystallization process using model-based observers. A neural network model based on the estimate mass of crystal was used as internal model to predict the process

output. Besides, their work compared between NMPC and industrial data from a PID-controlled process. Kittisupakorn et al. (2009) used neural network as a model in the model predictive control (MPC) algorithm for a steel pickling process. Furthermore, the neural network is applied with model based predictive control in many researches (Yu and Gomm, 2003; Ławryńczuk, 2008; Georgieva and Azevedo, 2007).

The aim of this study is to improve the product quality by optimization and using a neural network to modeling and controlling of a batch heating/cooling evaporative crystallization. In a neural network section, the Lenvenberg-Marquaedt algorithm is used to train the neural network and the optimized structure of neural network is based on minimum the mean squared error (MSE) of training and testing data. For optimization, modeling and controlling of batch crystallization, the citric acid-water system is chosen as a case study (Choong and Smith, 2004).

1.2 Research Objectives

The main objectives of this research are:

1. To investigate heating/cooling integrated with evaporation in batch crystallization in order to increase the crystal yield and crystal size.
2. To model and control the batch heating/cooling evaporative crystallization by using the neural network.

1.3 Scopes of Research

The main Scopes of this research are.

1. Integration and optimization of heating/cooling and evaporation in batch crystallization is studied in this work.
2. The batch heating/cooling evaporative crystallization is modeled by using neural network. The network is trained by using Lenvenberg-Marquaedt algorithm as well as determined base on MSE for the training data. The activation function is sigmoid function.
3. Neural network based controller is applied to control temperature in the batch heating/cooling evaporative crystallizer.

1.4 Contributions of Research

The main contributions of this research are:

1. The effect of integration of heating/cooling and evaporation in batch crystallization affect on crystal size and crystal yield.
2. Dynamic optimization of batch heating/cooling evaporative crystallizer to maximize average crystal size and crystal yield.
3. The neural network model and controller for control the batch heating/cooling evaporative crystallization to target value desired.

1.5 Activity plan

The activity of this research can be listed as follows:

1. Literature survey and review.
2. Study MATLAB program.
3. Simulate and validate a batch heating/cooling evaporative crystallization with citric acid/water system.
4. Modeling neural network of a batch heating/cooling evaporative crystallization.
5. Optimize the batch crystallization and design controller.
6. Analyze and summarize the simulation results.
7. Write thesis and prepare a manuscript for publication.

CHAPTER II

LITERATURE REVIEWS

This chapter presents the literature reviews of crystallization process, application of neural network which consists of neural network forward model, neural network inverse model and neural network based model predictive control

2.1 Crystallization process

A crystallization process plays an important role in many industries. In batch crystallization processes, the quality of final crystal product is focused. There are many researches focused on improving the quality product which is crystal yield and crystal size. Combining heating/cooling and evaporation in batch crystallization is also the main topic of interest due to it can produce high supersaturation, high crystal yield and large crystal size (Choong and Smith, 2004; Mesbah et al, 2011; Fagervik et al, 1988)

Mukhopadhyay and Epstein (1980) simulated the semi-batch evaporative crystallization of the citric acid-water system. The crystallizer was operated at constant temperature 50 ˚C and the feed flow rate was always equal to the evaporation rate. Resulting, the effect of changing flow rate was more pronounced than the effect of changing feed concentration.

Neelakantan and Mukesh (1979) studied the development of a model for a multistage continuous evaporative crystallizer for the production of crystal sugar based on assumption constant temperature 70 ˚C in each stage, constant crystal growth rate and negligible nucleation. The results showed that when the number of stages was increased the crystal flow rate decreased and the coefficient of crystal size variation decreased.

Fagervik et al. (1988) focused on an adaptive on-line simulation which is demonstrated by means of a developed adaptive on-line simulator connected with a

batch evaporative pilot plant crystallizer for sugar. As an adaptive model is able of predicting future behavior very accurately even of strongly non linear processes. The results presented on-line simulation an implementation of different control strategies has successfully been performed.

Sowul and Epstein (1981) developed a procedure for estimating crystallization kinetic parameters based on transient crystal size distributions for continuous evaporative crystallizers. The crystallizer temperature was maintained at 75 ˚C and pressure was operated under a vacuum of 0.38 atm.

Fevotte et al. (1990) presented a simple application of the non linear "L/A" controller to a batch evaporative crystallizer. A comparative study is proposed between the use of PID controller and the L/A algorithm. The simulation results presented the L/A algorithm which is easier to tune and more satisfactory behavior in closed loop than PID.

Choong and Smith (2004) simulated and optimized three modes of evaporative crystallization: (i) batch unseeded and seeded evaporative crystallization, (ii) semi-batch evaporative crystallization and (iii) heating/cooling evaporative crystallization. In first and second modes were simulated at isothermal 60 ˚C and third mode was simulated at non-isothermal which was decreased temperature from 60 ˚C to 40 ˚C. Furthermore, Choong and Smith simulated and optimized batch cooling crystallization. The results showed that the heating/cooling evaporative crystallization yielded an average crystal size, which was 35% larger than that produced by conventional and unseeded constant evaporation, 12% larger than that by batch seeded evaporative crystallization and 8% larger than that by semi-batch operation. In heating/cooling evaporative crystallization, the crystal yield can be increased to 47% higher than all modes of crystallization.

Mesbah et al. (2011) presented a model-based control approach for optimal operation of a seeded fed-batch evaporative crystallization. The evaporative crystallization is operated at isothermally 50 ˚C and pressure 100 mbar. In additions, the crystallization of ammonium sulphate-water system based on assumptions constant of crystallizer volume, size independent growth of crystals and constant evaporation rate.

Mesbah et al. (2008) focused on an optimal control of the seeded fed-batch evaporative crystallization process. An optimal control problem pertinent to maximization of the batch crystal yield is solved using the sequential optimization approach. Experimental results also demonstrate that the application of the proposed optimal control strategy leads to a substantial increase in the crystal volume fraction at the end of the batch.

2.2 Application of neural network

2.2.1 Neural network forward model

An artificial neural network is a mathematical structure that is used to model the linear or non-linear system. A neural network forward model is successfully to modeling the process which is complexity (Kittisupakorn et al., 2005; Charoenniyom et al., 2011; Nueaklong et al., 2011). The literatures on the designing of neural network are presented.

Yang and Wei (2006) studied the development of a neural network model to predict crystallization kinetics such as crystal nucleation, growth, and agglomeration rates. The model was trained and validated with data obtained from an anti-solvent crystallization system that composed of ciprofloxacin hydrochloride, H2O, and ethanol. The result indicated that the neural network model gave much more-accurate prediction of the crystallization kinetics. The mean relative error of the predicted growth rates between this model and the measured data is generally <10% and, in some cases, is as good as 5%.

Georgieva and Feyo de Azevedo (2006) focused on the benefits of applying hybrid strategy for dynamics behavior modeling of crystallization processes combining analytical and the artificial neural network (ANN) approaches and model predictive control (MPC) based on a feed forward neural network (FFNN) model in an industrial fed-batch evaporative sugar crystallization. The ANN was trained offline in batch mode with input-output data generated by the KBHM.

Keong et al. (2004) designed the neural network to simulate and predict the crystallizer temperature of the Ni–P based amorphous alloys, as functions of alloy

composition and heating rate. The input parameters of the neural network (NN) model are alloy composition, heating rate of the heating processes, and processing parameters of the alloys. The output parameters are crystallization temperatures (onset, peak and end temperatures) of major crystallization reaction of the amorphous alloys. In training and testing, the neural network for the simulation and prediction of crystallization peak temperature showed a comparatively good result.

Arpornwichanop and Shomchoam (2009) improved the control performance by hybrid neural network and on-line optimal control strategy and demonstrate for the control of a fed-batch bioreactor for ethanol fermentation. The information of unmeasured state variables obtained from the neural network as an on-line estimator was used to modify the optimal feed profile of the fed-batch reactor. The simulation results show that the neural network provides a good estimate of unmeasured variables and the on-line optimal control with the neural network estimator gives a better control performance.

Kittisupakorn et al. (2005) described neural network models for the prediction of the concentration profile of a hydrochloric acid recovery process consisting of double fixed-bed ion exchange columns. Backpropagation and Lenvenberg-Marquardt techniques were used to train various neural network architectures and the accuracy of the obtained models were examined by using test data set. The optimal neural network architectures of this process can be determined by MSE minimization technique. The simulation results showed that multilayer feedforward neural network models with two hidden layers provide sufficiently accurate prediction of the concentration profile of the process.

Wong et al. (2010) used of an artificial neural network (ANN) to model a lactose crystallization process and estimate lactose crystallization kinetic rates, nucleation, growth, and aggregation rates, under different processing conditions by using a mixed-suspension, mixed-product removal (MSMPR) crystallizer. The lactose crystallization kinetic rates were modeled both by the conventional power law relationship and by the feed-forward ANN architecture. The results revealed that the ANN approach gave a much better prediction of the kinetic rates. Compared to the power law model, the R^2 and root mean-square error (RMSE) of the ANN approach improved by at least 49% and 33%, respectively.

Kumar et al. (2008) investigated modeling of a batch sucrose crystallization using neural network to predict the corresponding crystal growth rate and compared with conventional nonlinear regression analysis. A Levenberg-Marquardt algorithm and back propagation was used to train neural network. The simulation results showed that the optimal neural network consists of 4 input nodes, 10 hidden nodes and 1 output nodes. The correlation coefficient between the experimentally determined crystal growth rate and the crystal growth rate by the neural network and nonlinear regression were 0.999 and 0.748, respectively.

Mujtaba et al. (2006) studied three different types of nonlinear control strategies which were developed and implemented in batch reactors using neural network techniques. These are generic model control (GMC), direct inverse model control (DIC) and internal model control (IMC) strategies. The neural network was used as dynamic estimator, dynamic model (forward model) and control (inverse model). The simulation results showed all types of controllers performed well in tracking the optimal temperature profile and achieving target conversion to the desired product. However, the neural network used in DIC and IMC controllers need training beyond the nominal operating condition to cope with uncertainties better.

Lin et al. (2010) applied a neural-network-based approximate dynamic programming (ADP) method in an industrial sucrose crystallization optimal control problem. The neural network model of the crystallization used the data from the actual sugar boiling process of sugar factory to train the network. The results of simulation showed the controller based on action dependent heuristic dynamic programming approach can optimize industrial sucrose crystallization.

Charoenniyom et al. (2011) presented neural network modeling for the prediction of concentration profile of methacrylic acid in batch reactor of esterification reaction for a Methyl methacrylate (MMA) production process. And this work studied optimization to find optimal reactor temperature profile to maximum desired product. The simulations showed that the optimal neural network architecture of this process, which was 6 nodes in input layer, 8 nodes in 1^{st} hidden layer and 13 nodes in 2^{nd} hidden layer, was a best structure network to predict concentration profile of methacrylic acid and reactor temperature. The MSE index of the data test is 1.1101e-006.

Nueaklong et al. (2011) presented neural network modeling for hard chrome electroplating process to predict the plating solution temperature in hard chrome electroplating bath and inverse model-based control strategy for controlling plating solution temperature to the desired temperature range. Simulation results showed that 6-6-1 appear to be the best for predict the plating solution temperature in hard chrome electroplating bath. In control strategy, the comparison of performance between NNDIC and PI controller under nominal and model mismatch cases, it was found the conventional PI controller better results in both cases.

2.2.2 Neural network model based predictive control

Model predictive control (MPC) is a control strategy which is computed based on an optimization that is formulated over a prediction horizon, using a model to predict the effect of future inputs on system states or outputs. Neural network model which is successfully to predict in optimization in MPC represent a mathematical model (Kittisupakorn et al., 2009; Ławryńczuk, 2008; Damour et al.; 2010).

Yu and Gomm (2003) studied the multivariable neural network modeling and MPC technique for application to a MIMO laboratory-scaled chemical reactor. Three variables which were temperature, pH and dissolved oxygen were controlled in the reactor. On-line control results were presented to illustrate the closed-loop performance of the neural network model-based predictive control scheme, and a comparison is made with conventional PID control.

Ławryńczuk (2008) studied artificial neural networks for modeling and temperature control using model predictive control of a yeast fermentation biochemical reactor. The neural model of the process was trained using available data sets generated from the fundamental model. As a result, the classical PID controller and the MPC algorithm based on a linear model were unable to control the process. But MPC algorithms based on neural models can control the process.

Kittisupakorn et al. (2009) developed a multi-layer feedforward neural network model based predictive control. In the acid baths three variables under controlled are the hydrochloric acid concentrations. In the modeling, multiple input, single output recurrent neural network subsystem models were developed using

input–output data sets obtaining from mathematical model simulation. In the control (MPC) algorithm, the feedforward neural network models were used to predict the state variables over a prediction horizon within the model predictive control algorithm for searching the optimal control actions via sequential quadratic programming. The proposed algorithm was tested for control of a steel pickling process in several cases in simulation such as for set point tracking, disturbance, model mismatch and presence of noise. The results for the neural network model predictive control (NNMPC) overall showed better performance in the control of the system over the conventional PI controller in all cases.

Georgieva and Feyo de Azevedo (2007) investigated modeling an industrial fed-batch evaporative sugar crystallization process combining analytical and artificial neural network (ANN) approaches and controlling by using Model Predictive Control (MPC) and Feedback Linearizing Control (FLC) based on an ANN nonlinear control. A comparison of MPC veasus FLC approaches were evaluated with respect to closed loop performance, constrains feasibility and computational efforts. MPC algorithm outperformed the FLC approach with respect to satisfactory reference tracking and smooth control action. However, the MPC is computationally much more involved since it requires an online numerical optimization, while for the FLC, an analytical control solution was determined.

Damour et al. (2010) presented a nonlinear model-based predictive control (NMPC) approach applied to an industrial process. A neural network model based on the estimate mass of crystal was used as internal model to predict the process output. Besides, their work compared between NMPC and industrial data from a PID-controlled process. The simulation results presented the NMPC strategy leaded to significant time and energy saving more than the PID-controlled process about 11%.

2.2.3 Neural network inverse model

One of the most applications of neural network is inverse modeling that is applied to control the process. Some researchers trained the neural network by feedback learning (Li and Häuβler, 1996; Yan et al., 2009) and some researchers

trained the neural network by feed forward learning (Hussain et al., 2001; Daosud et al. 2005).

Li and Häuβler (1996) developed direct neural control systems by the proportional plus derivative control. Direct neural control systems were developed and trained by genetic algorithm. The optimal structure of neural network was 3-3-4-1. The simulation results showed the response of the controlled system exhibits a satisfactory performance and a small overshoot.

Nakanishi and Schaal (2004) presented the technique of feedback error learning (FEL) from the viewpoint of adaptive control. Numerical results demonstrated the significant difference in the stability property between the adaptive feedback formulation and the adaptive feed forward formulation, particularly when learning of unstable plant dynamics was considered.

Gomi and Kawato (1990) proposed learning schemes using feedback error learning for a neural network model applied to adaptive nonlinear feedback control. After the neural network compensated perfectly or partially for the nonlinearity of the controlled object through learning, the response of the controlled object finally follows the desired response set in the conventional feedback controller. This learning scheme did not require the knowledge of the nonlinearity of a controlled object in advance.

Yan et al. (2009) proposed an adaptive inverse controller for nonlinear discrete-time system. A compound neural network was constructed to identify the nonlinear system, which included a linear part to approximate the nonlinear system and a recurrent neural network to minimize the difference between the linear model and the real nonlinear system. The inverse model can be used to control both nonlinear dynamic discrete-time SISO and MIMO systems in real time. Simulation studies showed that the scheme is simple and affects good control accuracy and robustness.

Hussain et al. (2001) presented the nonlinear inverse-model based control strategy to control an exothermic reactor. The utilization of two different inverse-model schemes namely the direct inverse control and the internal-model control methods were shown for both set point and disturbance rejection cases. The overall

results for set point tracking were good in both control strategies but the direct inverse control method had limitations when dealing with disturbances.

Guez et al. (1998) indicated how the neural network computation algorithms can be used for adaptive control and outlines resulting computational advantages. The neuromorphic control approach was compared to model reference adaptive control on a specific example. The utilization of neural networks for adaptive control offered definite speed advantages over traditional approaches for very large scale systems.

Daosud et al (2005) investigated the use of neural network direct inverse model-based control strategy (NNDIC) to control a steel pickling process. The concentration of acid solution in the pickling step needs to be maintained at the optimum value in order to obtain the maximum reaction rate. The optimal neural network architectures are determined by the mean squared error (MSE) minimization technique. The robustness of the proposed inverse model neural network control strategy was investigated with respect to changes in disturbances, model mismatch and noise effects. Simulation results showed the superiority of the NNDIC controller in the cases involving disturbance, model mismatch and noise while the conventional controller gives better results in the nominal case.

CHAPTER III

THEORY

In this chapter, the theoretical background of a crystallization process, neural network and model predictive control are described.

3.1 Crystallization

Crystallization is a separation and purification process, used in the production of a wide range of materials. It involves the formation of one or more solid phases from a liquid phase or amorphous solid phase. Crystallization is one of the older unit operations in the chemical industry and it differs from most unit operations because of the presence of a solid product. The main advantages of crystallization are a high purity in one process step, a low level of energy consumption and relatively mild process conditions. The crystallization process is taken place in two steps: nucleation of the crystal and crystal growth. The driving force of crystallization process is supersaturation. The terminologies involved in the crystallization process are shown below.

3.1.1 Supersaturation

Crystallization processes can take place only in supersaturated, or supercooled, phases. The rate of crystallization is often determined by the degree of supersaturation, the driving force for crystallization, which has been commonly expressed either as the difference in concentration between the supersaturated and saturated solutions. The most common definitions of supersaturation in industrial crystallization are the concentration driving force, ΔC, the supersaturation ratio, S, and the relative supersaturation, σ_s . The various definitions of the supersaturation are as follows:

$$\Delta C = C_S - C^* \tag{3.1}$$

$$S = \frac{C_S}{C^*} \tag{3.2}$$

$$\sigma_S = \frac{C_S}{C^*} = S - 1 \tag{3.3}$$

3.1.2 Phase Diagram for Crystallization

There are three region on a phase diagram for crystallization, as shown in Figure 3.1. The first region is below the equilibrium solubility curve between the liquid phase and the solid phase. This zone, called the unsaturated or undersaturated region, is the region in which any crystals present in the solution will dissolve until the saturation point is reached.

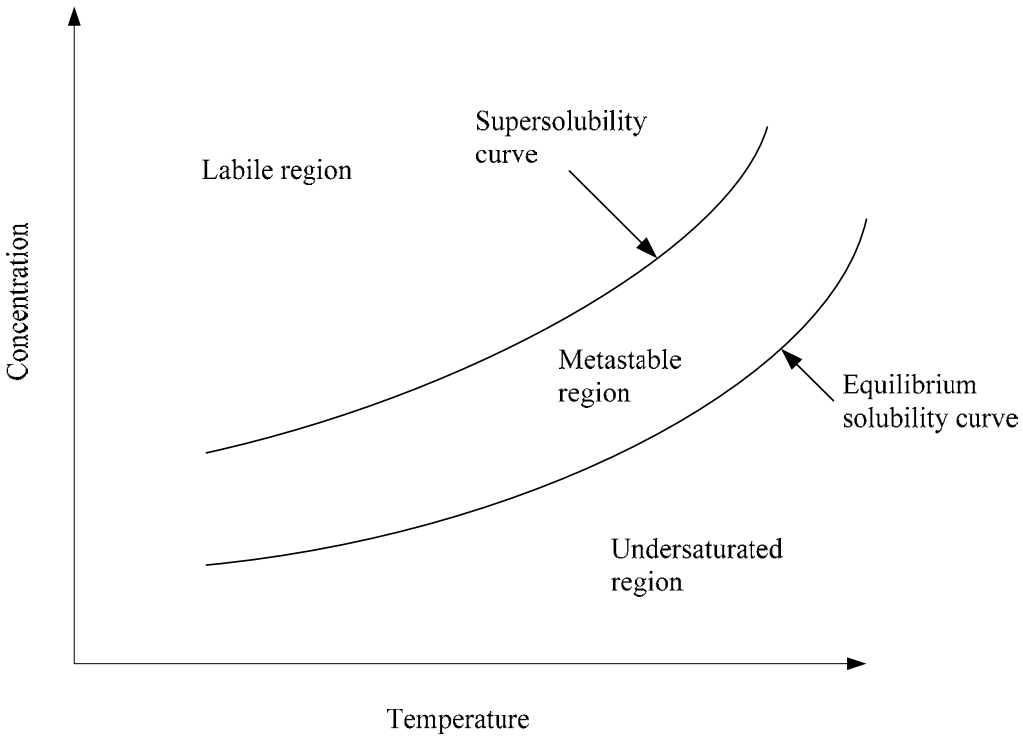


Figure 3.1 Phase diagram of crystallization

The second region is called the metastable region. This region bounded between the supersolubility curve and the solubility. Here the solution is supersaturated, but the driving force is not enough to form nuclei or new crystals. If there are some crystals already present in the solution, they will grow in size so that the concentration is reduced to the equilibrium point, however no nuclei can be formed in a finite length of time. If there are not any crystals present in the solution then the solution will remain Metastable region

The region above the supersolubility curve is called the labile region. If the solution is this region, spontaneous crystallization will occur and large amounts of crystal nuclei will be formed.

3.1.3 Solubility

For the process of crystallization, the solubility is a very important value because it determines the value of the supersaturation of the solution, which in turn partly determines the value of crystallization rate. The solubility often increases significantly with the temperature, which is demonstrated at higher temperature, commonly used in industrial crystallization, by the high viscosity of mother liquors and high dry solids content. However there are also other systems where the saturation concentration remains approximately constant or decreases with increasing temperature.

3.1.4 Nucleation

Nucleation of the crystals is the creation of the hypothetical solid particles in the solution. This includes the formation of small crystals nuclei in which there is or there is not a presence of the other crystals. The nucleation in case of no presence of other crystals is called primary nucleation. On the other hand, the nucleation in case of the presence of an influence of the existing macroscopic crystals in the solution is called secondary nucleation. Total nucleation is the summation of the effects of the primary and the secondary nucleation. The primary nucleation can occur in two conditions.

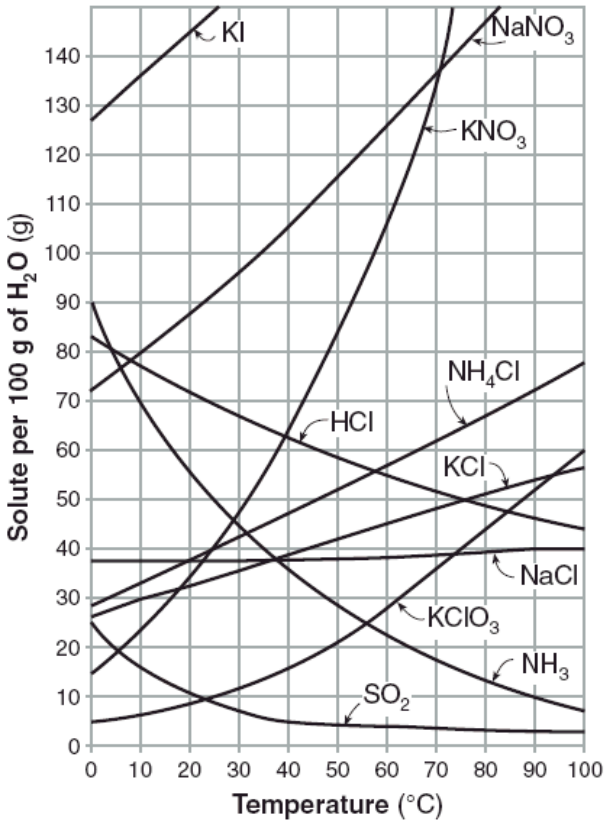


Figure 3.2 Solubility curve

First is homogeneous nucleation, which is nucleation that is no influence of any other solids such as the wall of the crystallizer or particles of any foreign substances. Second is heterogeneous nucleation. This occurs when solid particles of foreign substances do an influence on the nucleation process by catalyzing and increasing the nucleation rate. For the primary nucleation, the homogeneous nucleation is rarely occurs in practice due to the high energy necessary to begin nucleation without a solid surface to catalyze the nucleation, except perhaps in some precipitation reaction.

In secondary nucleation, the two kinds of secondary nucleation are known, a fluid-shear nucleation and a contact nucleation. The fluid-shear nucleation is take place when a growing crystal is swept away at the surface by the supersaturated solution. The swept-away-nuclei become grow to a new crystal. The contact nucleation is the result of collisions between existing crystals with one another or with the walls of the crystallizer and rotary agitators. This occurs at low supersaturation where the growth rate of the crystals is at the optimum for good quality. Contact

nucleation has been found to be the most effective and common method for nucleation because low energy is required and easy control without unstable operation.

$$B_p = k_p(T) * (C_s - C^*)^p \tag{3.4}$$

$$B_s = k_s(T) * (C_s - C^*)^s M^{b}_{crys} \tag{3.5}$$

There is another method of crystals nucleation that to be avoided. This method is called spurious nucleation. The crystals are attrition, more likely to crushing than to real nucleation. This happened when crystals are impacted with the moving part of the crystallizer. The sources of crystals generating are presented in Figure 3.3.

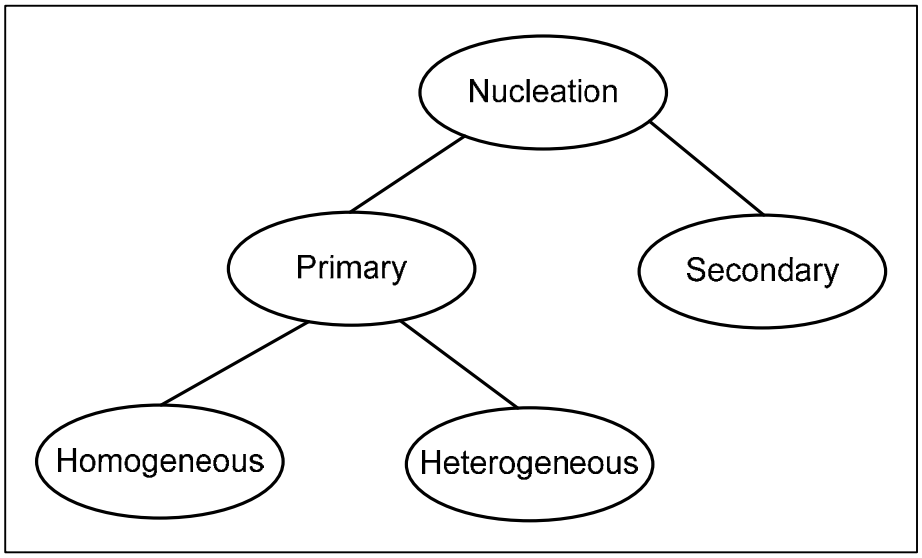


Figure 3.3 Various types of nucleation in crystallization

3.1.5 Crystal Growth

A crystal face is a planar surface that grows as existing steps or kinks on the surface area augmented by the incorporation of new solute molecules. Layers spread progressively across the face as new molecules attach themselves to the accessible and energetically favorable leading edges. At the microscopic level, solute molecules moving from the bulk solution adsorb on the crystal solid surface and are incorporated into the crystal lattice. A crystal face is a planar surface that grows as existing steps or

kinks on the surface are augmented by the incorporation of new solute molecules. Surface adsorption and diffusion determine whether a solute molecule is incorporated into the crystal or returns to the bulk phase. The observed growth rate is then caused by the flow of steps across the surface.

3.1.6 Crystal Size Distribution

The appearance and size range of a crystalline product is extremely important in crystallization. If further processing of the crystals is desired, large crystals with uniform size are important for washing, filtering, transportation, and storage. The importance lies in the fact that large crystals are easier to filter out of a solution than small crystals. Also, larger crystals have a smaller surface area to volume ratio, leading to a higher purity. This higher purity is due to less retention of mother liquor which contains impurities, and a smaller loss of yield when the crystals are washed to remove the mother liquor. The theoretical crystal size distribution can be estimated as a function of operating conditions with a fairly complicated mathematical process called population balance.

3.1.7 Crystallization Methods

Supersaturation can be generated by one or more of three methods. That is cooling, evaporation and antisolvent crystallization. (Tung et al., 2009)

(i) Cooling crystallization

Crystallization by cooling is commonly practiced for solutions in which solubility is a strong function of temperature. Because of it is energetically favorable and simple to operate. Cooling alone can achieve the desired degree of crystallization when solubility is sufficiently low at the termination of the cooling operation. In some cases, additional reduction in solubility is necessary to achieve the desired yield. However, a disadvantage of cooling method is the early fouling of the cooling surfaces and feed point with crystallizing solute. The effect factors to the

supersaturation profile either locally or globally for crystallization by cooling are following.

- rate of cooling and wall temperature
- width of the metastable region
- nucleation rate and inherent crystal growth rate
- presence or absence of seed and seed quantity
- mixing and mass transfer
- solvent system
- impurities (dissolved and undissolved)

(ii) Evaporative crystallization

Increasing the concentration by evaporation or distillation is a common method of increasing supersaturation and inducing crystallization. Since solvent is removed over a finite period of time, it is inherently a semibatch operation. Semicontinuous or continuous operation is also possible. The evaporation or distillation can be run at atmospheric pressure, or at reduced pressure when substrate stability is not compatible with the required atmospheric distillation temperature. One of the primary advantages of evaporative procedures is that they can often be combined with other process operations to reduce equipment requirements and/or time cycles. In addition, it is possible in some cases to complete the crystallization without the addition of a second solvent, thereby avoiding the costs of separation and recovery. Some of the process advantages that may be realized are as follows:

- combination with a change in solvent and simultaneous crystallization
- combination with reaction and removal of a volatile reaction by-product and simultaneous crystallization
- combination with cooling crystallization

These operational advantages must be evaluated against the disadvantages that are discussed in the sections to follow. These disadvantages may include

- difficulty in controlling mean particle size and particle size distribution (PSD)

- difficulty in determining the seed point
- unpredictability on scale-up
- inconsistent batch-to-batch performance

(iii) Antisolvent crystallization

Addition of an antisolvent is potentially the best method to achieve controlled and scalable particle size distribution (PSD). Control of both supersaturation and crystal growth area is readily achievable by control of the antisolvent addition rate. This control requires consideration of both the change in solubility as addition proceeds and the crystal growth area and is, therefore, potentially more complex than for the single-solvent processes of cooling and concentration.

The obvious disadvantage of the antisolvent process is the necessity to introduce an additional solvent or solvents, thereby reducing the volumetric productivity and creating a solvent mixture requiring some form of purification/separation for downstream processing and/or recovery.

Table 3.1 Advantage and disadvantage of crystallizer

Crystallizer type	Advantage	Disadvantage
Cooling	- simple to operate	- early fouling of the cooling surfaces and feed point with crystallizing solute
Evaporation	- combine with other process operations to reduce equipment requirements and/or time cycles	- difficulty to control
Antisolvent	- easy to control	- create a solvent mixture requiring separation for recovery.

3.2 Neural Network

Neural network is an information-processing system that has certain performance characteristics in common with biological neural networks. Neural networks have been developed as generalizations of mathematical models of human cognition or neural biology, based on the assumptions that information processing occurs at many simple elements called neurons, signals are passed between neurons over connection links, each connection link has an associated weight and each neuron applies an activation function (usually nonlinear) to its net input to determine its output signal. Neural Network is a structure composed of a number of interconnected units (neuron by neuron, element by element). A Neuron consists of axons, dendrites, synapses and soma (cell body) as shown in Figure 3.4. The Axon acts as transmission line of the neuron. At the ends of the axon are complex, highly specialized structure called synapases. The dendrites receive the inputs from other cells and the soma integrates the inputs to transmit along the axon to the synapses.

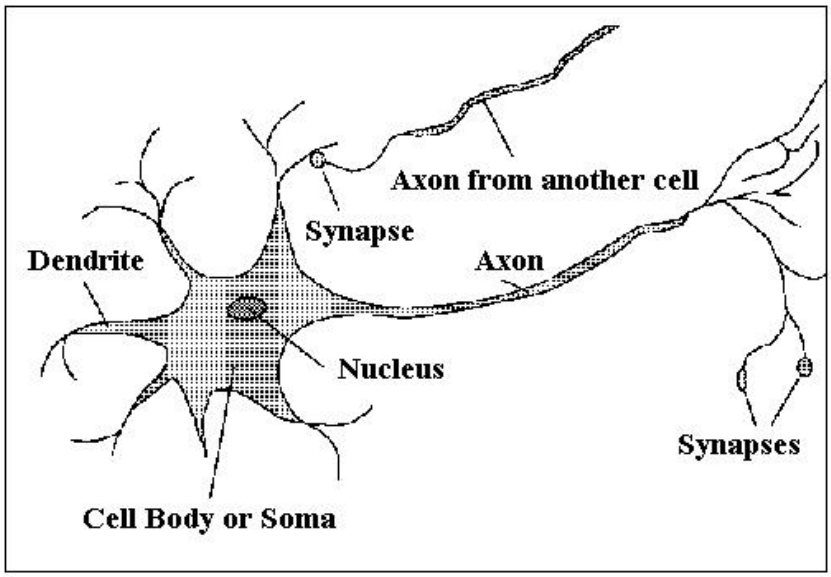


Figure 3.4 Axons, dendrites and synapse in a biological neuron

3.2.1 Components of Neural Network

The neural network consists of many interconnected neurons or nodes. In each node, there are many components for creating the neural network. These components are described as the follow.

3.2.1.1 Weighting Factors

Weight factors are the coefficients that determine the intensity of the network inputs. Each network input an associated weight that controls the connection strength. Weighting factors can be modified corresponding to various training sets and according to a network's specific topology.

3.2.1.2 Basis function

Basis function is mathematical mapping with function $u(w,x)$ which w, x refer to metric weight and input vector. Basis function is applied to combined and summed up the inputs and their corresponding weighting factors and passing to next step. There are two common forms of basis function.

1) Linear basis function (LBF)

Linear basis function is the most common basis function which is summation of multiplying each input and corresponding weight While (x_1, x_2, …, x_n) represent the input vectors and (w_1, w_2, …, w_n) represent the corresponding weight vectors. The linear basis function can be written by the following equation:

$$u_i(w,x) = \sum_{j=1}^{n} w_{ij} x_i \tag{3.6}$$

2) Radial basis function (RBF)

Radial basis function is more complex than linear basis function. The radial basis function can be written by the following equation:

$$u_i(w,x) = \sqrt{\sum_{j=1}^{n}(x_i - w_{ij})^2} \quad (3.7)$$

3.2.1.3 Activation function

The outputs of basis function are transmitted to activation function which transforms the results from summation to the network output. The activation function is linear functions or nonlinear functions that nonlinear functions are popular but linear functions are not useful because linear functions are limit. The most commonly used activation functions are summarized in Table 3.2

Table 3.2 Activation functions of a neural network

Activation functions	Formula	Characteristic
Step	$\begin{cases}1 & \text{if } x > 0 \\ 0 & \text{otherwise}\end{cases}$	Linear
Ramp	$\begin{cases}1 & \text{if } x \geq 1 \\ x & \text{if } x < 1 \\ -1 & \text{if } x < -1\end{cases}$	Linear
Linear	$f(x) = x$	Linear
Sigmoid	$f(x) = \frac{1}{1+e^{-x}}$	Nonlinear
Tan-Sigmoid	$f(x) = \frac{e^{+x} - e^{-x}}{e^{+x} + e^{-x}}$	Nonlinear

3.2.1.4 Error function

In network training, the difference between the desired output value and predicted value by the network is used as the error signal to train the network. The

target of network training is minimum error. This error is transformed by the error function that has an effect on the network structure. These error functions are described as the follow

1) Sum square error

$$SSE = \sum_{i=1}^{N}(y_i - p_i)^2 \quad (3.8)$$

2) Mean square error

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - p_i)^2 \quad (3.9)$$

3) Mean absolute error

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - p_i| \quad (3.10)$$

which y_i is the network output

p_i is the network target

3.2.1.5 Learning Function

The objective of the learning function is to modify the weighting factors that connect the inputs of each process element. The neural network learns new knowledge by adjusting these weighting factors and the learning ability of a neural network is determined by its architecture and by the algorithmic method chosen for training. There are two types of learning algorithm as shown in following.

1) Supervised learning

In supervised learning, an explicit signal is provided by the teacher throughout to guide the learning process that consists of the sets of input-output examples. By the build in knowledge, the supervised is able to provide the neural networks with the desired respond for the training. During the

training, the output of neural network is compared to the desired output and weighting factors are adjusted by the network

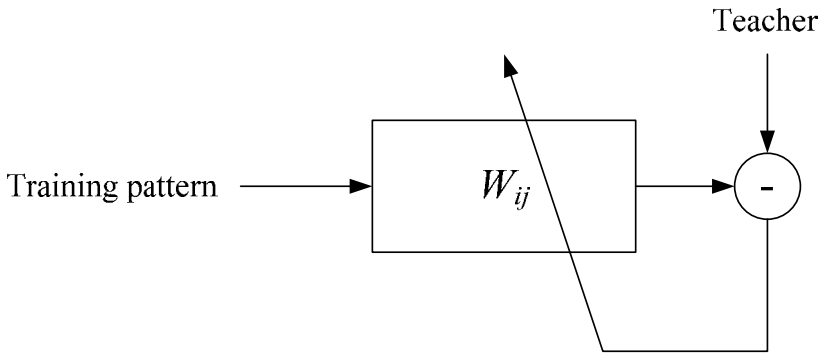


Figure 3.5 Supervised learning

2) Unsupervised learning

The unsupervised learning which is sometimes called self-supervised learning does not require an external teacher to guide the learning process. The network use only inputs data to learn. Unlike the supervised learning require an external teacher to provide training signals that guide the learning process.

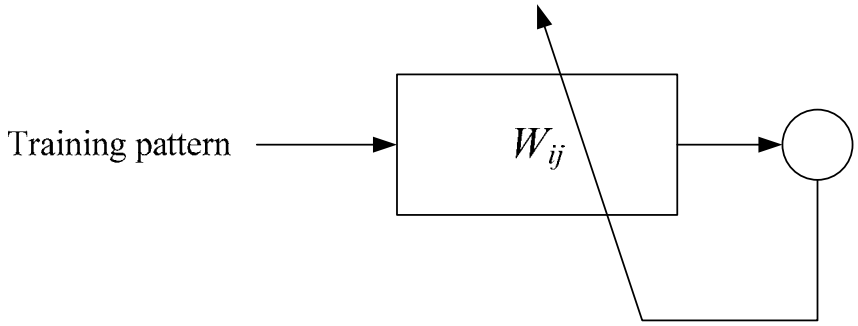


Figure 3.6 Unsupervised learning

3.2.2 Neural Network Architecture

3.2.2.1 Network Structure

Neural network structure can be divided into common types such as feedforward networks and feedback networks

1) A feedforward neural network is a basic process neural network model, and is an information forward propagation network model that consists of some process neurons including the traditional time-invariant neuron as a special case. The signals travel from input to output one way only. There is no feedback in the network such as the output of any layer.

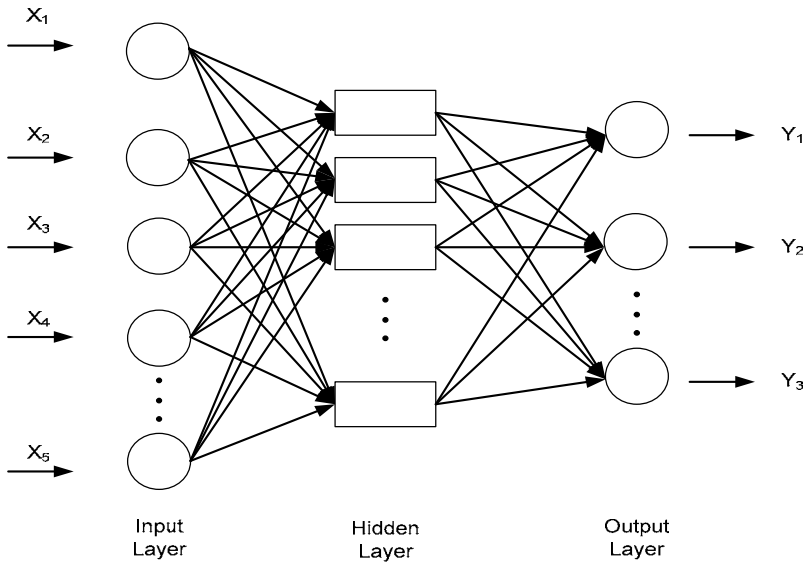


Figure 3.7 Feedforward neural networks

2) A feedback neural network has connections from output to input neurons. Such a neuron keeps a memory of previous state so that the next state depends not only on input signals but also on the previous states of the network.

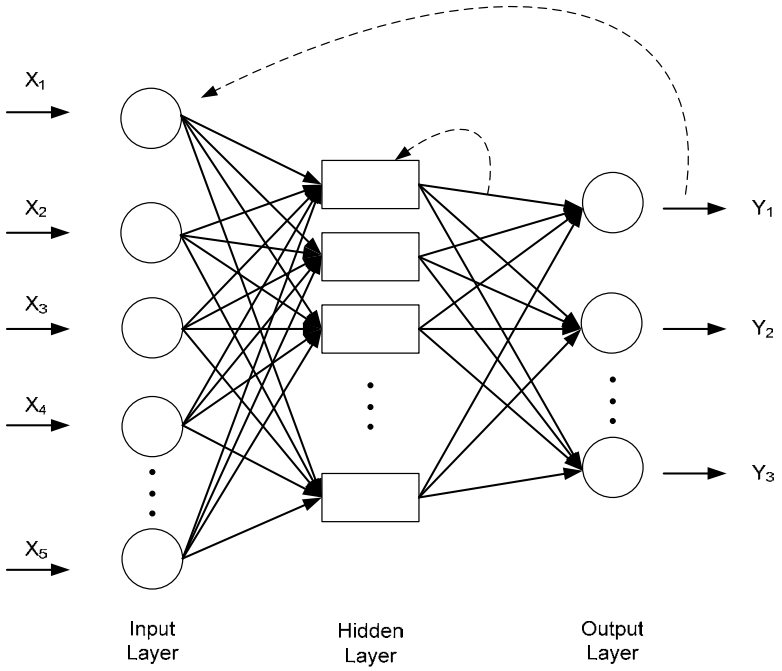


Figure 3.8 Feedback neural networks

3.2.2.2 Connection Structure

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 feedforward, feedback, lateral and time delayed connections.

1) Feedforward Connections

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

2) Feedback Connections

For the neural network models, the feedback connections bring the data from neurons of an upper layer back to neurons of lower layer.

3) Lateral Connections

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

4) Time Delayed Connections

Delay element may be incorporated into the connections to yield temporal dynamics model. They are more suitable for temporal pattern recognition.

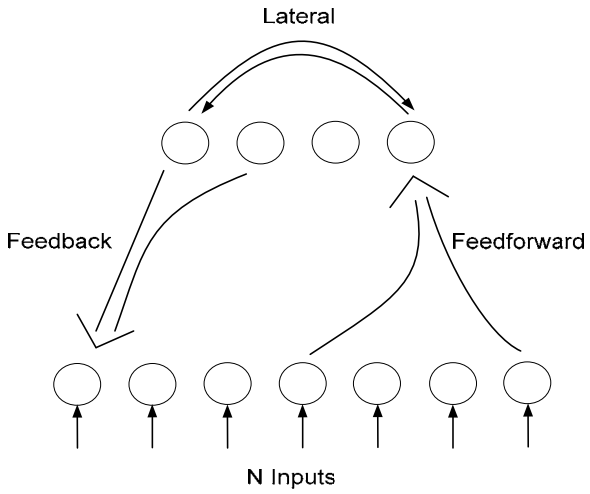


Figure 3.9 Connection structures of neural networks

3.2.2.3 Network Layers

The layers of a 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. Figure 3.10 show a single-layer neural network and a multiple-layer neural network.

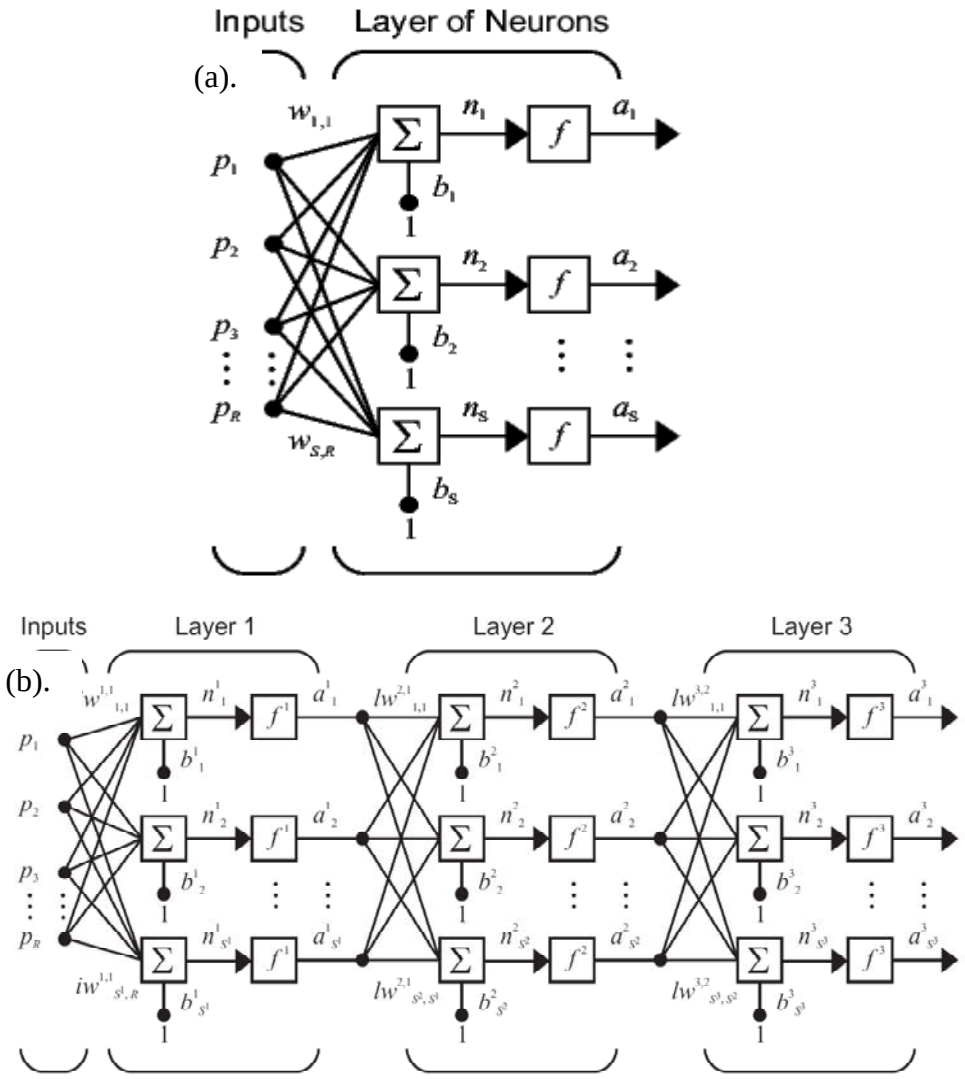


Figure 3.10 (a). A single-layer neural network (b). A multiple-layer neural network (MATLAB 2009)

3.2.3 Training Algorithm

The aim of network training is to minimize the error between targets and network outputs. Training is procedure that determines the optimal values of the connection weights and biases. Training begins by initially assigning arbitrary small random values to the weights. Training proceeds iteratively until a satisfactory model is obtained.

3.2.3.1 Back Propagation Algorithm (Zilouchian and Jamshidi, 2001)

Back propagation algorithm is one of the most popular algorithms for training a network due to its success from both simplicity and applicability viewpoints. The algorithm consists of two phases: Training phase and recall phase. In the training phase, first, the weights of the network are randomly initialized. Then, the output of the network is calculated and compared to the desired value. In sequel, the error of the network is calculated and used to adjust the weights of the output layer. In a similar fashion, the network error is also propagated backward and used to update the weights of the previous layers. Figure 3.11 shows how the error values are generated and propagated for weights adjustments of the network. In the recall phase, only the feedforward computations using assigned weights from the training phase and input patterns take place. Figure 3.12 shows both the feedforward and back propagation paths. The feedforward process is used in both recall and training phases. On the other hand, as shown in Figure 3.12 (b), back propagation of error is only utilized in the training phase. In the training phase, the weight matrix is first randomly initialized. After that, the output of each layer is calculated starting from the input layer and moving forward toward the output layer. Thereafter, the error at the output layer is calculated by comparison of actual output and the desired value to update the weights of the output and hidden layers.

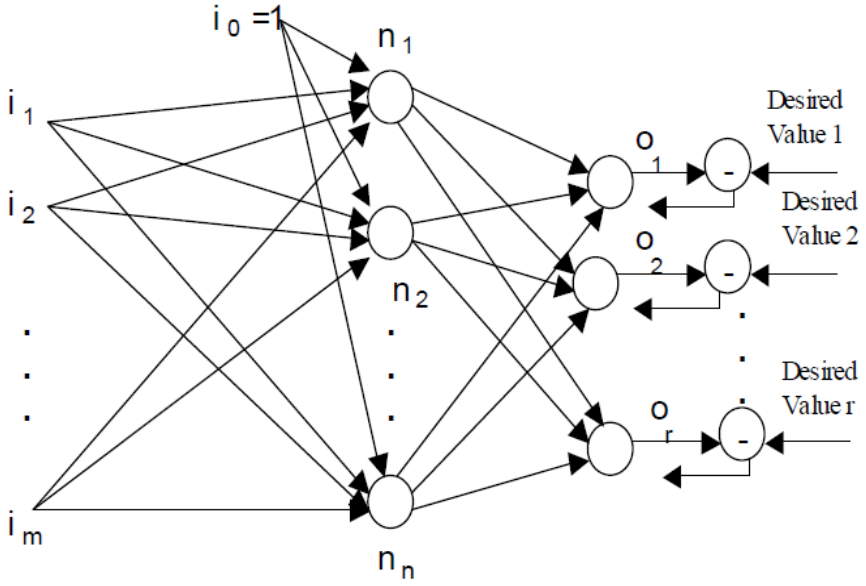
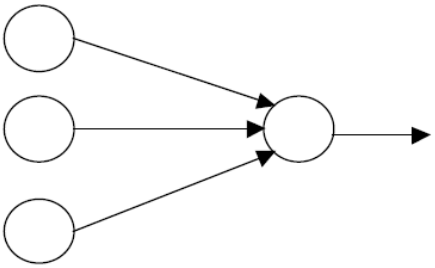
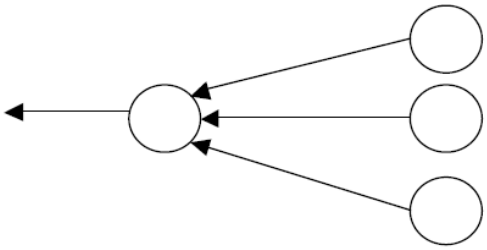


Figure 3.11 Back propagation of the error in a two-layer network



a) Forward propagation (Training and Recall Phase)



b) Backward propagation (Training Phase)

Figure 3.12 Forward propagation in recall and training phase and backward propagation in training phase

There are two different methods of updating the weights. In the first method, weights are updated for each of the input patterns using an iteration method. In the second method, an overall error for all the input output patterns of training sets is calculated. In other words, either each of the input patterns or all of the patterns together can be used for updating the weights. The training phase will be terminated when the error value is less than the minimum set value provided by the designer. One of the disadvantages of back propagation algorithm is that the training phase is very time consuming. During the recall phase, the network with the final weights resulting from the training process is employed. Therefore, for every input pattern in this phase, the output will be calculated using both linear calculation and nonlinear activation functions. The process provides a very fast performance of the network in the recall phase, which is one of its important advantages. The methodology of the conventional back propagation method is mentioned below

Inputs are summed and propagated to the hidden layer for a node *j* as:

$$net_j = \sum_{i=1}^{Ni} W_{ij} p_i^1 + b_j \tag{3.11}$$

Output from node *j* is given by

$$a_j^2 = f(net_j) \tag{3.12}$$

where *f* is the transfer function or activation function used in the hidden nodes

Hidden layer output is propagated to node *k* at the output layer given as:

$$net_k = \sum_{j=1}^{N_j} W_{kj} a_j^2 + b_k \tag{3.13}$$

Output from the node *k* is:

$$a_k^3 = f(net_k) \tag{3.14}$$

Error is calculated at the output layer as:

$$e = \frac{1}{2} \sum_{k=1}^{N_k} (p_k - a_k^3)^2 \tag{3.15}$$

Weights are adjusted along the negative gradient descent of the error, *e* as:

$$\Delta w_{kj} = -\eta \frac{\partial e}{\partial w_{kj}} \tag{3.16}$$

Weights in the output and the hidden layers are then corrected using equations below: The constant η (called the learning rate, and nominally equal to one) is put in to speed up or slow down the learning if required.

The gradient descent is simply the technique where parameters, such as weights and biases, are moved in the opposite direction to the error gradient. Each step down, the gradient results in smaller errors until an error minimum is reached. The network can get a better performance by using an approximation of Newton's method called Levenberg-Marquardt. This technique is more powerful than the gradient descent, but also requires more memory.

3.2.3.2 Levenberg-Marquardt Method

Like the quasi-Newton methods, the Levenberg-Marquardt algorithm was designed to approach second-order training speed without having to compute the Hessian matrix. When the performance function has the form of a sum of squares (as is typical in training feedforward networks), then the Hessian matrix can be approximated as (Othman, and Naseri, 2011)

$$H = J^T J \tag{3.17}$$

and the gradient can be computed as

$$g = J^T e \tag{3.18}$$

where J is the Jacobian matrix that contains 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 back-propagation technique that is much less complex than computing the Hessian matrix.

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

$$W_{k+1} = W_k - [\, J^T J + \mu I \,]^{-1} J^T e \tag{3.19}$$

When the scalar μ is zero, this is just 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 toward 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 is always reduced at the each iteration of algorithm.

3.3 Model Predictive Control

Model predictive control (MPC) differs from other control methods mainly in its implementation of the control actions. Usually, MPC solves a finite-horizon optimal control problem at each sampling instant, so that the control moves for the current time and a period of future time are obtained. MPC is a set of algorithms based on the models. MPC pays more attention to the function, than to the formulation, of the model. The function of a prediction model is based on the past information and the future inputs to predict the future output. Any collection of information, as long as it has the function of prediction, irrespective of the concrete form, can be the prediction model. (Seborg et al., 2004)

The basic ideas of model predictive control are follows:

1. Explicit use of a model to predict the process output at future time instants.

2. Calculation of a control sequence minimizing an objective function.

3. Receding strategy, so that at each instant the horizon is displaced towards the future that involves the application of the first control signal of the sequence calculated at each step.

The methodology of all the controllers belonging to the MPC family is characterized by the following strategy, represented in Figure 3.13:

The objective of the MPC calculations is to determine a sequence of control moves (that is, manipulated input changes) so that the predicted response moves to the set point in an optimal manner. The actual output y, predicted output $\hat{y}$ and manipulated input u are shown in Figure 3.13. At the current sampling instant, denoted by k, the MPC strategy calculates a set of M of values of the input $[u(k+i-1), i=1,2,\ldots,M]$. The set consist of the current input $u(k)$ and M-1 future inputs. The input is held constant after the M control moves. The inputs are calculated so that a set of P predicted output $[\hat{y}(k+i), i= 1, 2,\ldots, P]$ reaches the set point in an optimal manner. The number of predictions P is referred to as the prediction horizon while the number of control moves M is called the control horizon.

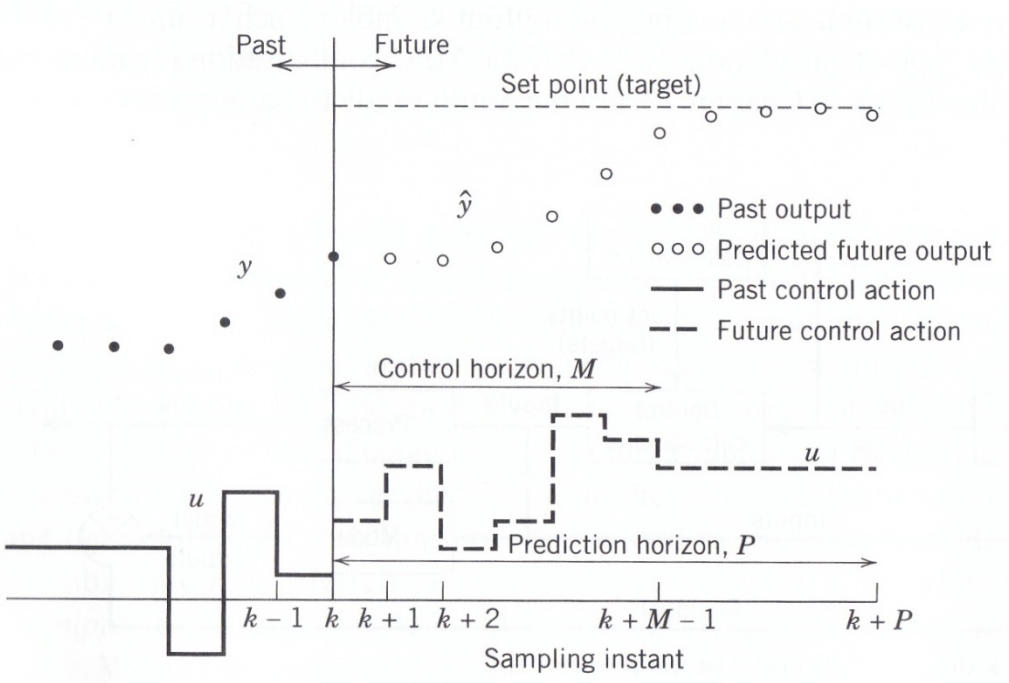


Figure 3.13 MPC Strategy

The idea of model predictive control is to utilize a model of the process in order to predict and optimize the future system behavior. The model form can be described by the following.

$$\dot{X} = f\big(X(t), U(t)\big) \tag{3.20}$$

The control law of the model predictive control is determined from the minimization of the controlled variable and manipulated variable. The optimization problem is as follows. (Kittisupakorn, 2008)

Objective function :$\min \int_0^{t_f} \{W_1(X - X_{sp})^2 + W_2(\Delta U)^2\} dt$ (3.21)

State space model :$\dot{X} = f\big(X(t), U(t)\big)$ (3.22)

Manipulated variable constraint :$U_{min} \leq U \leq U_{max}$ (3.23)

State variable constraint :$X_{min} \leq X \leq X_{max}$ (3.24)

Terminal state constrain :$X\big(t + t_f\big) = X_{sp}$ (3.25)

where W_1 and W2 are the weighting factors on the controlled and manipulated variables, respectively, t_f is the terminal time, U_{min} and U_{max} are the minimum and maximum bounds of manipulated variables and X_{min} and X_{max} are the minimum and maximum bounds of state variables.

CHAPTER IV

MATHEMATICAL MODEL OF A BATCH HEATING/COOLING AND EVAPORATIVE CRYSTALLIZATION PROCESS

4.1 Mathematical Model

A mathematical model of a batch heating/cooling and evaporative crystallization consists of two mass balances of solute and solvent, energy balance and crystallization model (Choong and Smith, 2004). The assumptions are applied in this model include (1) agglomeration and breakage of crystals are negligible, (2) total nucleation rate is the sum of primary and secondary nucleation rates, (3) crystal size is independent growth, (4) the solution is well mixed and (5) crystal nuclei produced from primary and secondary nucleation have negligible size (Choong and Smith, 2004). By performing mass balances, the following equations are obtained:

The mass balance of solute:

$$\frac{dC_s}{dt} = \frac{-k_v \rho_c}{M} \frac{d(m_3 M)}{dt} - C_s \frac{d(\ln M)}{dt} \tag{4.1}$$

The mass balance of solvent:

$$\frac{dM}{dt} = -Q_{evap} \tag{4.2}$$

where C_s is the concentration of solute, k_v is the volumetric shape factor, ρ_c is the density of crystals, m_3 is the third moment of crystal size, M is the mass of solvent and Q_{evap} is the evaporation rate.

The evaporation rate is a function of temperature and can be calculated by the following equation: (Apelblat et al., 1995; Wang et al., 1996)

$$Q_{evap} = \frac{\alpha(P_v\text{-}P_a)A_s}{\sqrt{2\pi mRT}} \tag{4.3}$$

$$\ln P_v = -58.9840 - \frac{1571.31}{T} + 11.4377\ln(T) \tag{4.4}$$

where P_v is the saturation vapor pressure, P_a is the partial pressure in the crystallizer, A_s is the evaporation surface area, α is the evaporation coefficient, m is the molecular weight and R is the gas constant.

The energy balance around the crystallizer is shown as follows:

$$\frac{dT}{dt} = \frac{UA_j(T_j\text{-}T) - H_{evap}Q_{evap} + H_{crys}k_v\rho_c\frac{dm_3}{dt}M}{C_pM_{tot}} \tag{4.5}$$

$$\frac{dT_j}{dt} = \frac{F_j\rho_jC_{pj}(T_{jsp}\text{-}T_j) - UA_j(T_j\text{-}T)}{\rho_jC_{pj}V_j} \tag{4.6}$$

and

$$H_{evap} = Ar \times abs(1 - T/T_c)^{Nr} \tag{4.7}$$

where T is the crystallizer temperature, U is the overall heat transfer coefficient, A_j is the total heat transfer surface area, T_j is the jacket temperature, H_{evap} is the heat of vaporization, H_{crys} is the heat of crystallization, C_p is the heat capacity of the solution, M_{tot} is the total mass of solution, F_j is the water flow rate in jacket, ρ_j is the density of water in jacket, C_{pj} is the heat capacity of the water in jacket, T_{jsp} is the jacket temperature set point and V_j is the jacket volume.

For the crystallization model can be describe by following equations.

The crystal growth rate

$$G = k_g(T)(C_s - C^*)^g \tag{4.8}$$

where G is the crystal growth rate, k_g is the crystal growth rate constant, C* is the solubility of the solute and g is the crystal growth rate exponent

The Population Balance Equation with the assumption of crystal growth rate, variable volume and well mixed batch crystallizer:

$$\frac{\partial n}{\partial t} + \frac{\partial(Gn)}{\partial L} + n\frac{d(\ln M)}{dt} = 0 \tag{4.9}$$

where *n* is the number density of the crystals, L is the characteristic crystal length and M is the total mass of solvent at any instant of time.

Mostly, the moment transformation method is applied to convert Eq. (4.9). The moment of the crystal size distribution are given in the following expression:

$$\frac{dm_0}{dt} = (B_p + B_s) - m_0 \frac{d(lnM)}{dt} \tag{4.10}$$

and

$$\frac{dm_i}{dt} = iGm_{i-1} + BL_0^i - m_i \frac{d(lnM)}{dt} \tag{4.11}$$

where B_p and B_s are the primary and secondary nucleation rate, respectively, L_0 is the characteristic crystal length of a newly formed crystal, and m_0 and m_i are the zero and i moment of crystal size (i = 0, 1, 2, 3, 4,…), respectively.

For nucleation, both primary and secondary nucleation is used to describe in the empirical power law form:

Primary nucleation:

$$B_p = k_p(T)(C_s - C^*)^p \tag{4.12}$$

Secondary nucleation:

$$B_s = k_s(T)(C_s - C^*)^s M_{crys}^b \tag{4.13}$$

where k_p and k_s are the primary and secondary nucleation rate constant, respectively, p and s are primary and secondary nucleation rate exponent, respectively, M_{crys} is the magma density of crystal and b is the magma density exponent.

The solubility of the solute can be expressed by a polynomial equation.

$$C^* = C_0 + C_1T + C_2T^2 + C_3T^3 \tag{4.14}$$

In this work, the objective functions of optimization are to maximize the average crystal size (weight mean crystal size), L_{aw} and the crystal yield. The expression for average crystal size and crystal yield are shown as follows:

$$L_{aw} = \frac{m_4}{m_3} \tag{4.15}$$

$$crystal\ \ yield = C_s(t_0)M(t_0) - C_s(t_f)M(t_f) \tag{4.16}$$

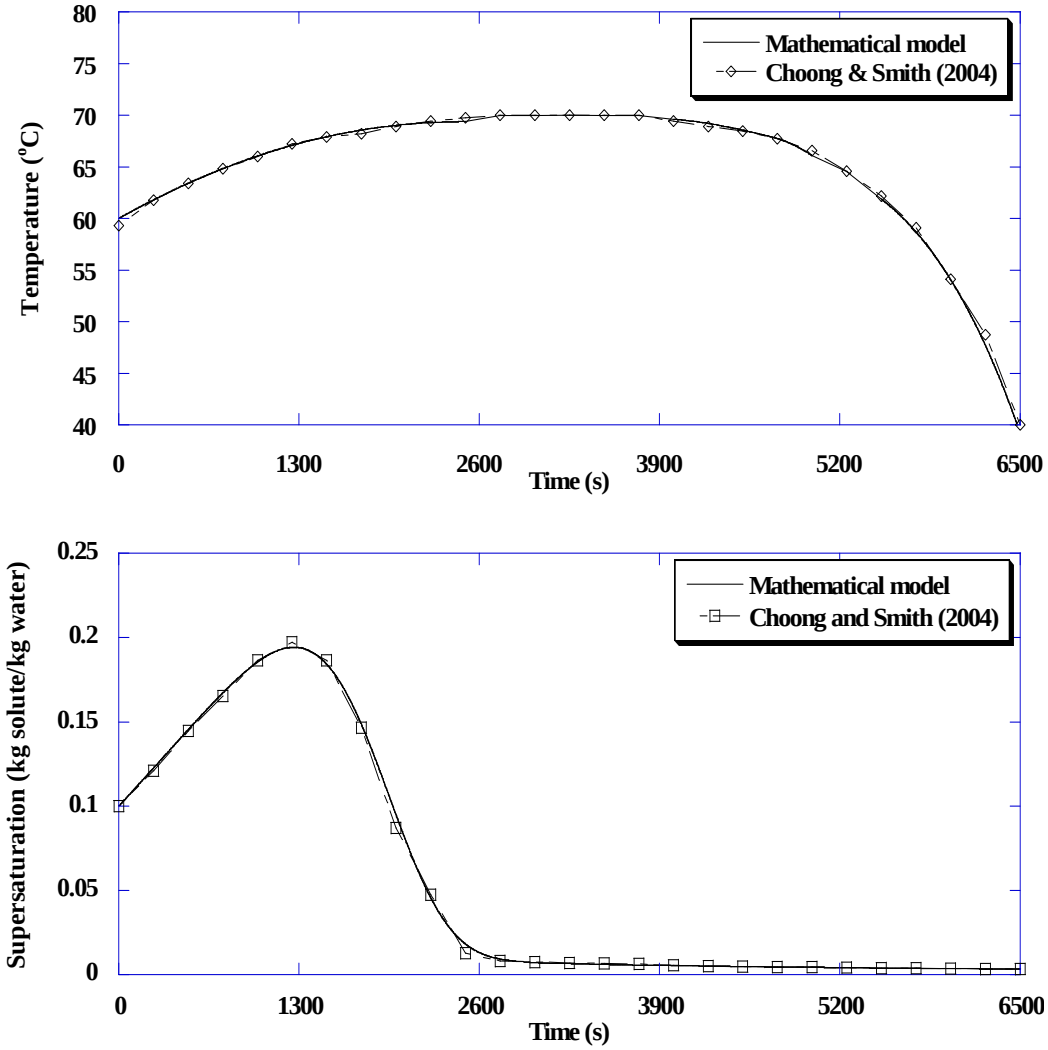


Figure 4.1 Comparison between mathematical model and Choong and Smith (2004)

The mathematical model of the batch heating/cooling and evaporative crystallization consists of two mass balances of solute and solvent, energy balance and crystallization model is solved by using Matlab. To ensure the simulation results, the proposed model is validated with simulation data of Choong and Smith (2004) who simulated heating/cooling evaporative crystallization in citric acid–water system. In his work, the process was operated at constant evaporation rate and non-isothermal that was increased temperature from 60°C to 70°C and decrease to 40°C. The crystallizer temperature and supersaturation are shown in Figure 4.1. It can be seen

that the supersaturation and the crystallizer temperature profiles which generate from mathematical model and reference are similar.

4.2 Crystallization and Physical Properties

The crystallization kinetics and physical properties of citric acid-water system are shown in Table 4.1 and the initial values as shown in Table 4.2 which were excerpted from Bohlin and Rasmuson (1992),Choong and Smith (2004) as well as Alexander Apelblat (1995).

Table 4.1 Crystallization and physical properties of citric acid–water system

Symbol	Unit	Value
A_j	m^2	0.07
Ar	kJ/kg	2891.83
A_s	m^2	0.019
b		0.84
C_0	kg/kgH_2O	0.91176
C_1	$kg/kgH_2O\ ^\circ C$	0.034857
C_2	$kg/kgH_2O\ ^\circ C^2$	-2.8785×10^{-4}
C_3	$kg/kgH_2O\ ^\circ C^3$	3.7228×10^{-6}
C_p	$kJ/kg\ ^\circ C$	2.1745
C_{pj}	$kJ/kg\ ^\circ C$	4.184
F_j		0.00003
g	kg/s	0.65
H_{crys}	kJ/kg	117
k_g	$m/(s(kg/kg)^g)$	0:02652 exp(−3584/T)
k_p	$no./((kg/kg)^p kgH_2O\ s)$	1.0×10^{-7}
k_s	$no./((kg/kg)^s (kg/kg)^b kgH_2O\ s)$	0:88774 exp(4781/T)
k_v		0.52
L_0	m	10×10^{-6}
m	kg/kmol	18
M_{crys}	kg/kgH_2O	0.024
Nr		0.321
p		3.54
P_a	kPa	10.1325
R	m^3 Pa/K mol	8.314
s		0.543
Tc	K	647.13
U	$W/m^2\ ^\circ C$	0.5
V_j	m^3	0.0014

Symbol	Unit	Value
α		0.5
ρ_c	kg/m^3	1540
ρ_j	kg/m^3	1000

Table 4.2 Initial value in this crystallization process

Symbol	Value
T	60 ˚C
m_i, i = 0, 1, 2, 3, 4	0
M	1.0 kg
C_s	2.871 kg/kgH_2O

4.3 Dynamic optimization

The dynamic optimization computes the optimal operating temperature policy to control the crystallizer. In batch crystallization process, the quality of crystal effect to downstream process such as filtration, drying and storage (Paengjuntuek et al., 2008). In this study, the crystal quality which is crystal yield and crystal size is focused (Choong and Smith, 2004). The aims of the dynamic optimization are to maximize the crystal yield and the crystal size. In this work, two optimization problems are considered.

Problem 1 (OPT1)

$$\max\ m_4(t_f)/m_3(t_f) \tag{4.16}$$

T(t)

subject to

the crystallizer model equations

$$T_{min} \leq T \leq T_{max} \tag{4.17}$$

$$L_{aw,min} \leq L_{aw} \leq L_{aw,max} \tag{4.18}$$

$$M_{min} \leq M \tag{4.19}$$

Problem 2 (OPT2)

$$\min C_s(t_f)*M(t_f) \quad (4.20)$$

T(t)

subject to

the crystallizer model equations

$$T_{min} \leq T \leq T_{max} \quad (4.21)$$

$$L_{aw,min} \leq L_{aw} \leq L_{aw,max} \quad (4.22)$$

$$M_{min} \leq M \quad (4.23)$$

where T_{min} and T_{max} is chosen as 40 and 70 °C, respectively, $L_{aw,min}$ and $L_{aw,max}$ is chosen as 315 and 550 μm, respectively and the minimum total mass of solvent, M_{min}, is 0.5 kg. The final batch time, t_f, is 6500 s.

The dynamic optimization calculates the optimal temperature profile by optimizing the objective function. Figure 4.2 shows crystallizer temperature profiles that provide maximum average crystal size (OPT1) and Figure 4.3 shows crystallizer temperature profiles that provide maximum crystal yield (OPT2).

In the optimization order, it classifies into 4 sets which consist of 1, 5, 8, and 10 intervals. In the optimization problems 1 and 2 get the maximum average crystal size and crystal yield at 10 intervals. The dynamic optimization is applied to be a set point of crystallizer temperature in PID control, neural network direct inverse control (NNDIC) and neural network model based predictive control (NNMPC). Moreover, it is applied in normal case and mismatch case for controlling.

Table 4.3 The simulation results obtained from two dynamic optimization problems

Problem	L_{aw}(μm)	Crystal yield (kg)
OPT1	451	1.585
OPT2	420	1.641

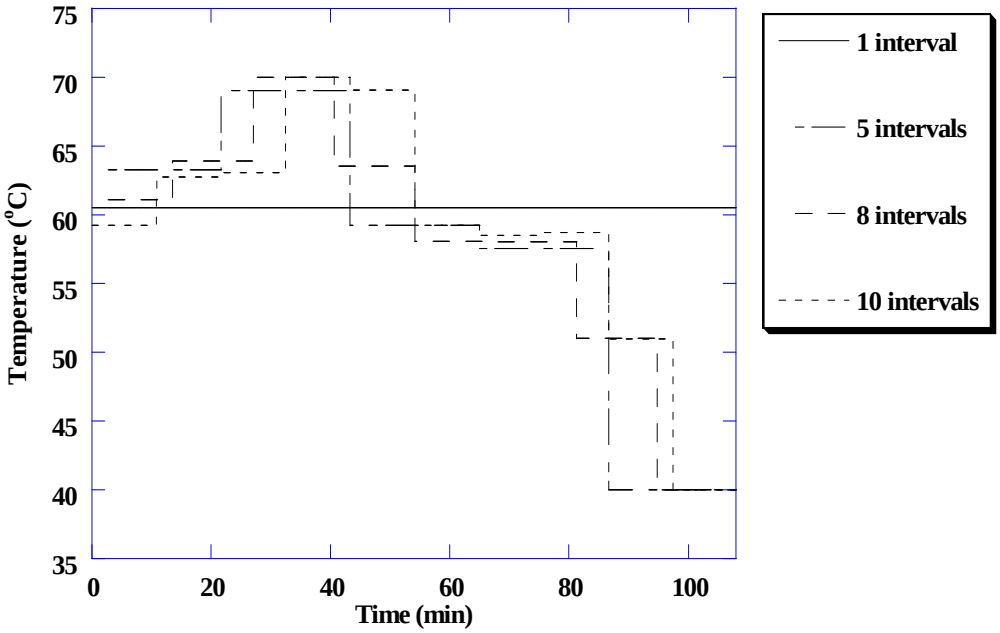


Figure 4.2 Temperature profile for the optimization problem 1 (OPT1)

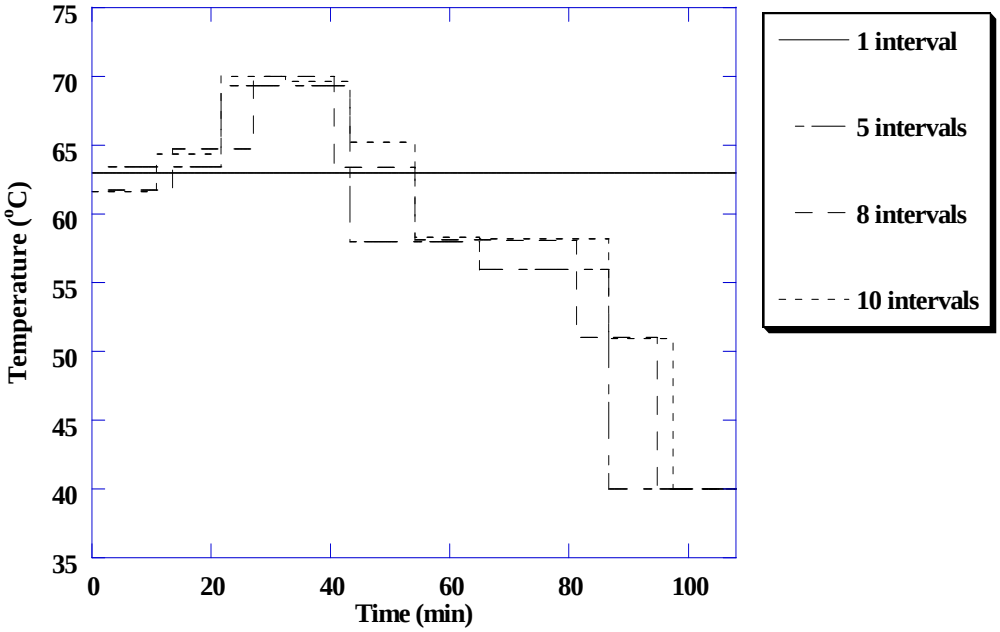


Figure 4.3 Temperature profile for the optimization problem 2 (OPT2)

The final product qualities in terms of average crystal size (L_{aw}) and crystal yield are shown in Table 4.3 It can be seen that OPT1 provides larger average crystal size than OPT2 about 6.46% but OPT2 gives more crystal yield than OPT1 about 3.39%. Moreover the average crystal size and crystal yield which are obtained from

the heating/cooling evaporative crystallization method are more than cooling and evaporation method. From Table 4.4, it can be seen that the heating/cooling evaporative crystallization method produces the average crystal size of 19% and 30% larger than cooling and evaporation method, respectively as well as the obtained crystal yield from heating/cooling evaporative crystallization method is higher than cooling and evaporation method about 50%.

Table 4.4 Comparison of the product quality among different crystallization method

Method	L_{aw}(μm)	Crystal yield(kg)
Unseeded cooling[a]	296	0.787
Evaporation[a]	315	0.787
Heating/cooling evaporative(OPT1)	451	1.585
Heating/cooling evaporative(OPT2)	420	1.641

[a]Obtained from Choong and Smith (2004)

CHAPTER V

NEURAL NETWORK FORWARD MODEL AND NEURAL NETWORK INVERSE MODEL FOR THE PROCESS

5.1 Neural network forward model

The neural networks consist of an input layer for receiving data from external source, one or some hidden layer(s) for processing input data and an output layer for displaying the output values. There are some neurons in each layer that are connected with some connections to previous and next layers. A neuron includes input and output values, weight factors and bias as well as transfer functions. The neural networks are trained using a training algorithm and a training data set in order to adjust the connection weights and biases.

For this neural network, the weight factors and biases are the coefficients that determine the relationship of the network inputs by randomly. The neural network uses tan-sigmoid as activation function of the nodes in hidden layer and linear function is used as the activation function in its output layer. The Levenberg-Marquardt algorithm is applied for training the network. The objective of neural network training is to minimize the error function which is mean square error (MSE) between the predicted neural network values and actual targeted value. The Eq.5.1 is shown the MSE calculation.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(T_{aci} - T_{pi})^2 \tag{5.1}$$

where n is the number of data, T_{ac} is the actual targeted temperature value and T_p is the predicted neural network values.

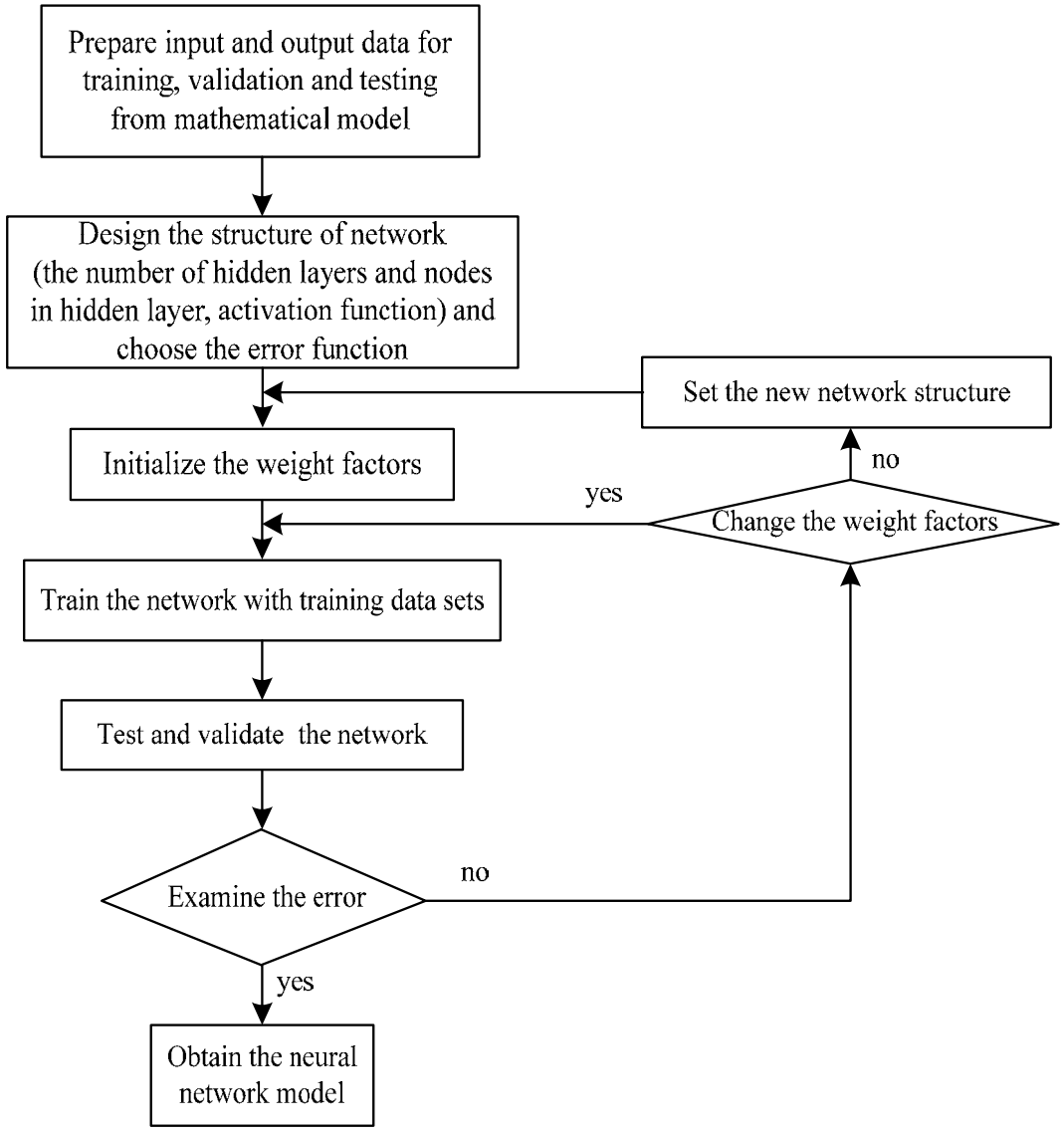


Figure 5.1 Steps of neural network structure designing

The steps of neural network structure designing are summarized in Figure 5.1. First step is preparing input and output data for training, validation and testing which are obtained by solving Eqs. (4.1) – (4.13) and varies the value of manipulated variable (jacket temperature set point, T_{jsp}) in step changes. Second step is to design the network structure and choose the error function. Next are to initialize the weight factors, train the network, test and validate the network and then examine the error. If

the error is unacceptable, the weight factors or structure will be reset to new values but if the error is acceptable, the neural network model will be obtained.

In this part, the neural network is applied for modeling of the batch crystallizer to predict the future value of concentration profile, crystallizer temperature profile and jacket temperature profile. The input and output data pattern of neural network forward model is given in Figure 5.2.

Neural network forward model

$C_s(k-1)$	$C_s(k)$	$C_s(k+1)$
$T(k-1)$	$T(k)$	$T(k+1)$
$T_j(k-1)$	$T_j(k)$	$T_j(k+1)$
$T_{jsp}(k-1)$	$T_{jsp}(k)$	

Output
Input

Figure 5.2 Input and output data pattern for forward model

The input and output data sets which consist of training, testing and validating data sets are obtained from mathematical model to train the neural network by varied the jacket temperature set point as shown in Figures 5.3-5.6. In the generated data for network training consist of plant certainty case called normal case and plant uncertainty cases that compose of -30% of U, +30% of H_{crys},-30% of H_{evap}, +30% of k_g and +30% of k_p. The input and output data sets which are trained in the neural network forward model are 18 sets.

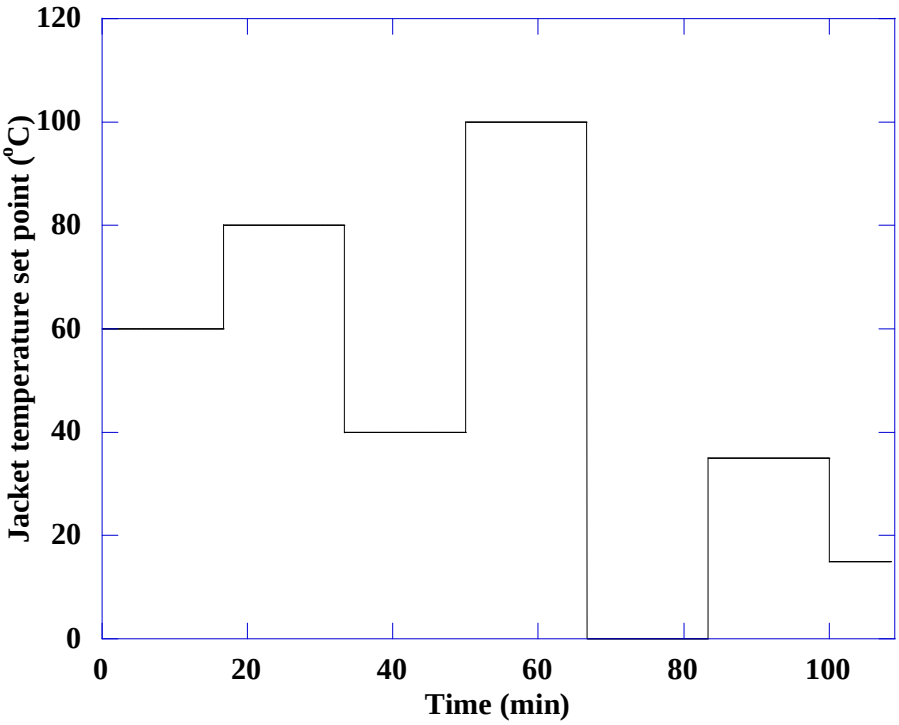
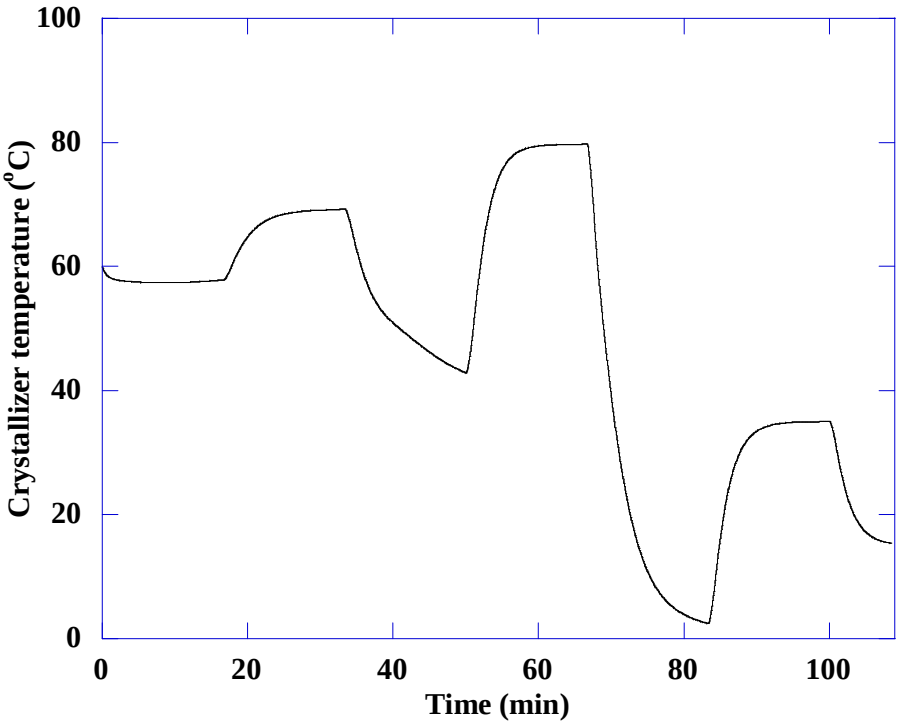


Figure 5.3 The jacket temperature set point profile of data set 1



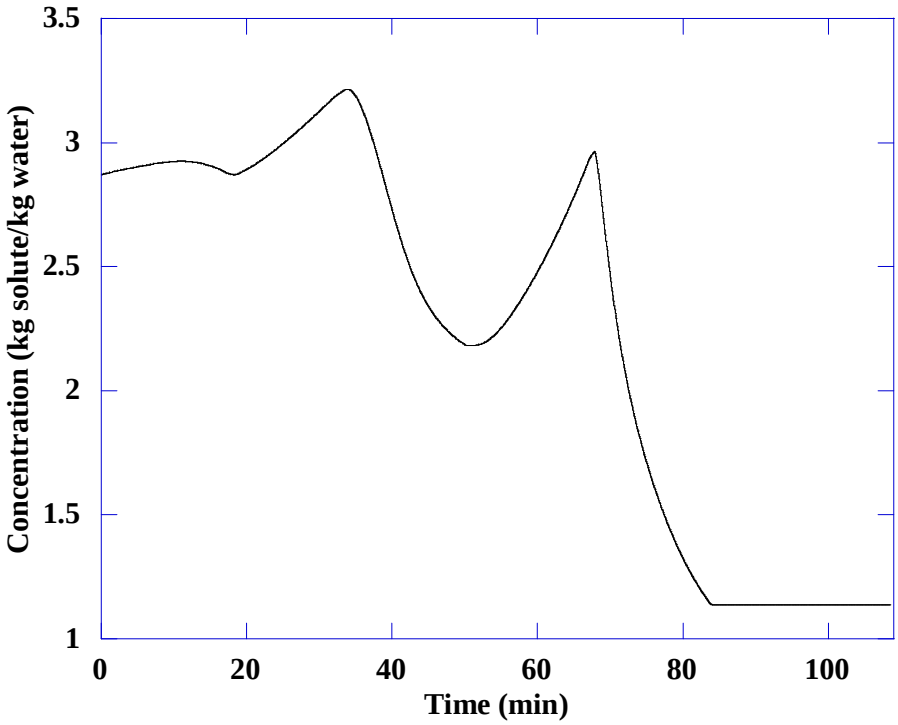


Figure 5.4 The crystallizer temperature and the concentration profile of data set 1

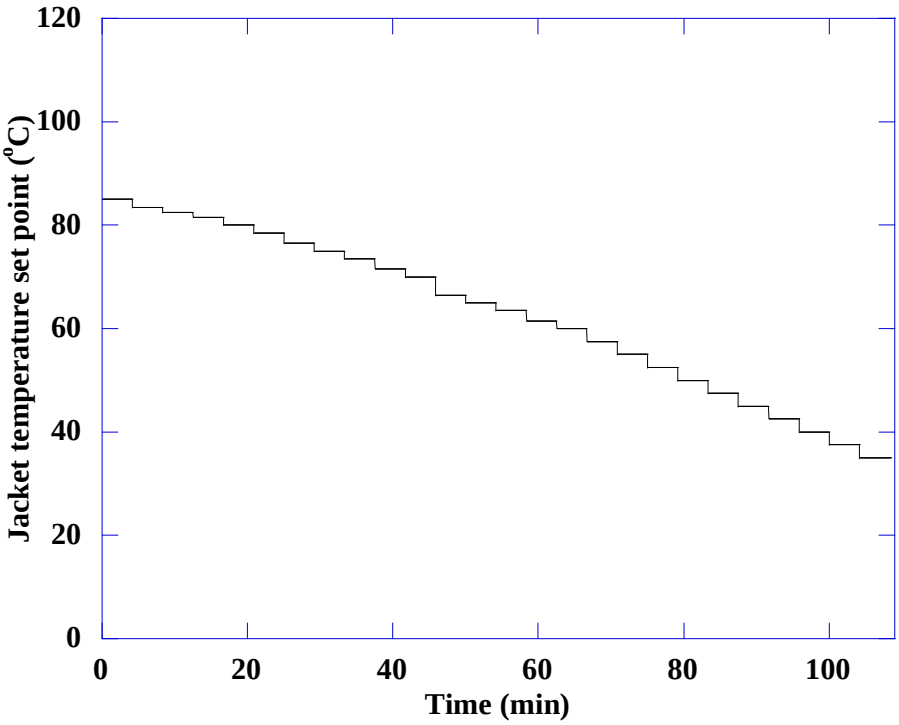


Figure 5.5 The jacket temperature set point profile of data set 2

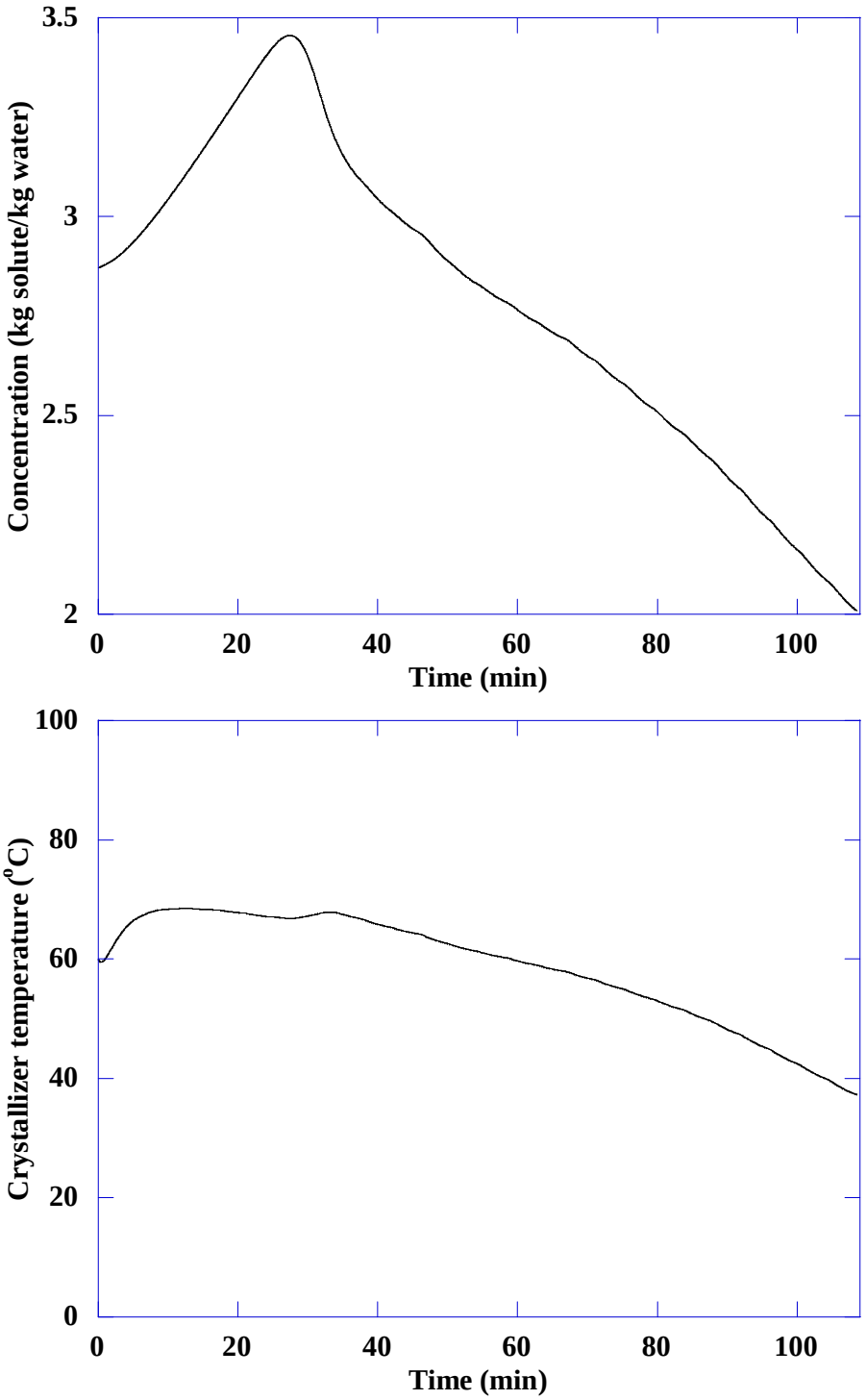


Figure 5.6 The crystallizer temperature and the concentration profile of data set 2

The data sets which are generated from mathematical model need to be normalized in order to overcome the significant minimum and maximum values used

in the training process. The raw process data generated are scaled down to between 0.05-0.95 using the following equations:

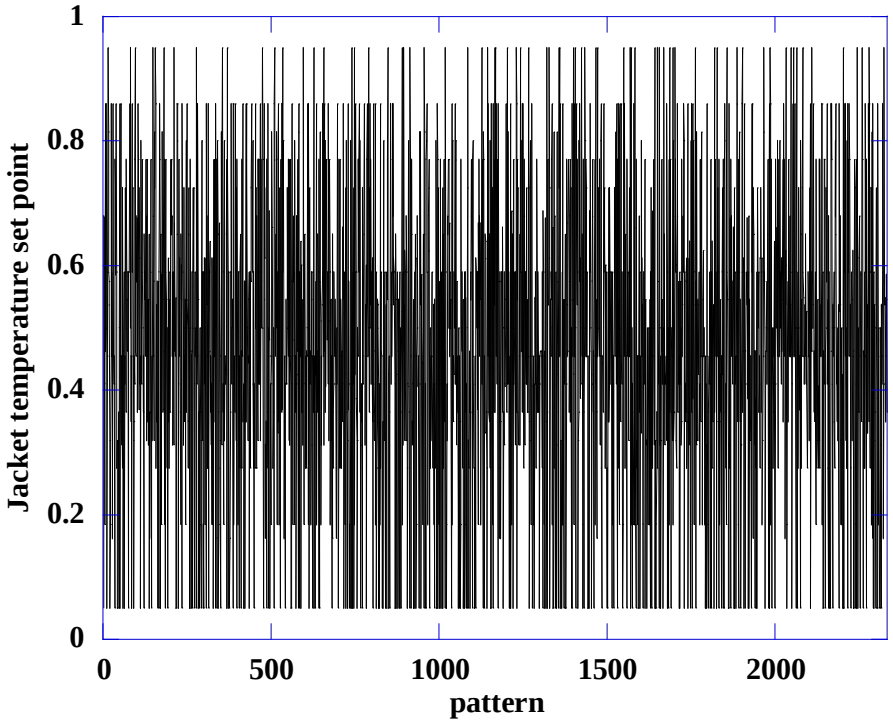
$$value_{sd} = \frac{(value_{ac} - value_{min})(0.95 - 0.05)}{(value_{max} - value_{min})} + 0.05 \tag{5.2}$$

and the actual value (scaled up) is given by

$$value_{ac} = \frac{(value_{sd} - 0.05)(value_{max} - value_{min})}{(0.95 - 0.05)} + value_{min} \tag{5.3}$$

where $value_{sd}$, $value_{ac}$, $value_{min}$, $value_{max}$ are the scaled down, actual, minimum and maximum values of the data, respectively.

The input and output data sets which are normalized are integrated and randomized to train test and validate the neural network. The training, testing and validating data set are 60%, 30% and 10% of all data, respectively. Figures 5.7-5.8 show the training data, testing data and validating data.



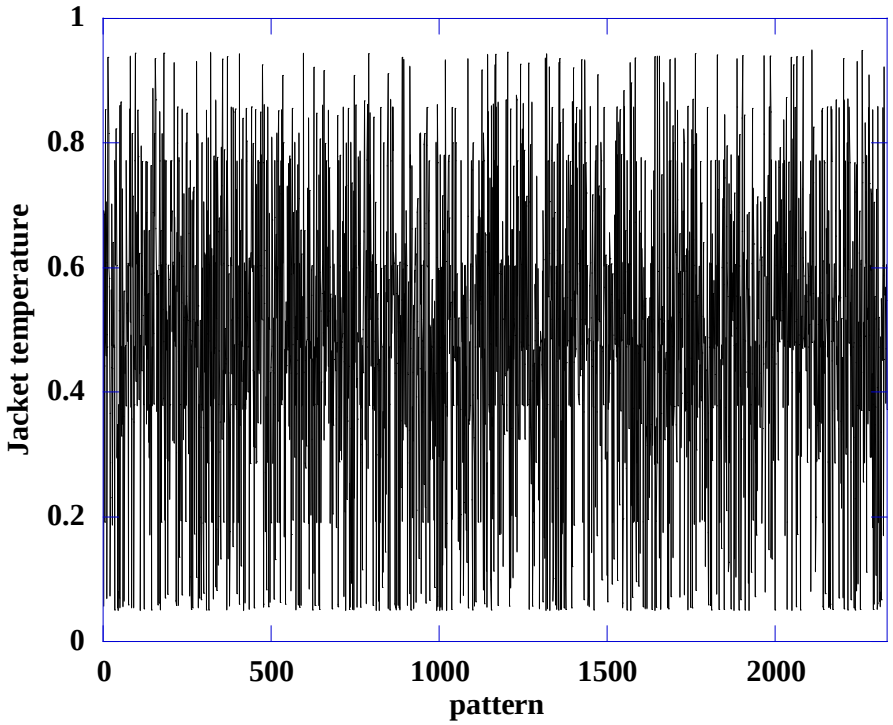
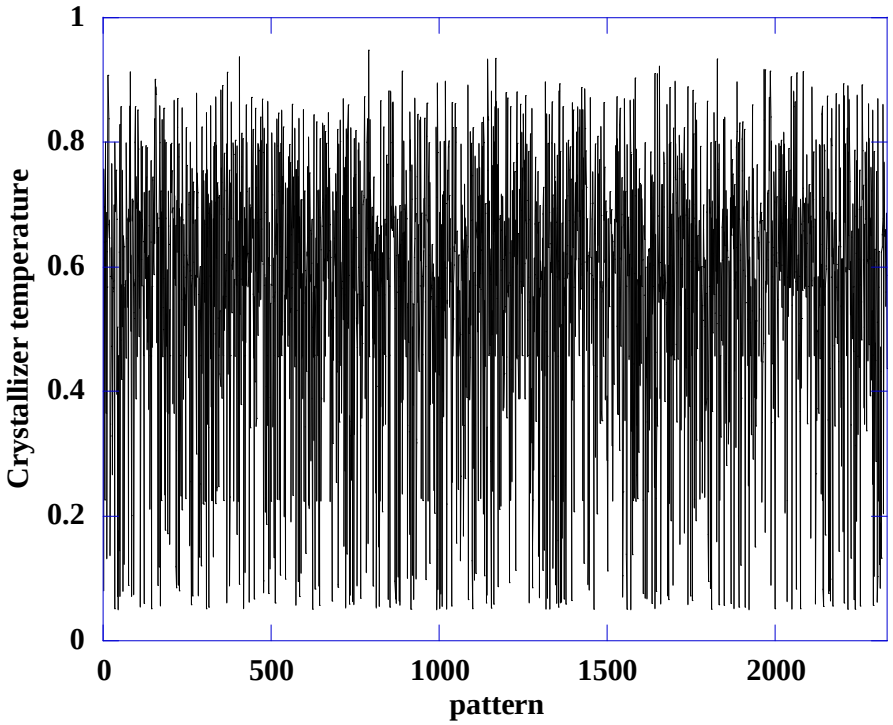


Figure 5.7 The jacket temperature set point and the jacket temperature profile of sum training data



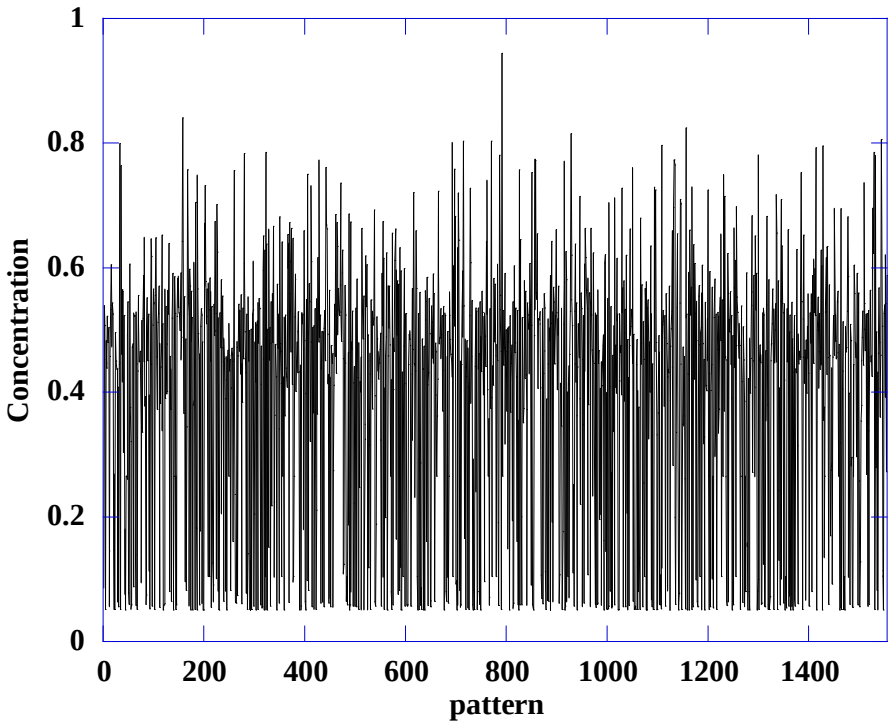


Figure 5.8 The crystallizer temperature and the concentration profile of sum training data

Figure 5.9 shows the final neural network structure is chosen to represent the forward model which is 8 input nodes that consists of $C_S(k-1)$, $C_S(k)$, $T(k-1)$, $T(k)$, $T_j(k-1)$, $T_j(k)$, $T_{jsp}(k-1)$ and $T_{jsp}(k)$ as well as 3 output nodes that is compose of $C_s(k+1)$, $T(k+1)$ and $T_j(k+1)$.

In the simulation demonstrates that the well-trained neural network can be used for modeling in order to predict the concentration, the crystallizer temperature and the jacket temperature profiles in this process. The optimal hidden layers for neural network forward model, input and output data pattern which is [8-14-10-3] (8 nodes in input layer, 14 nodes in first hidden layer, 10 nodes in second hidden layer and 3 nodes in output layer) as shown in the Figure 5.9.

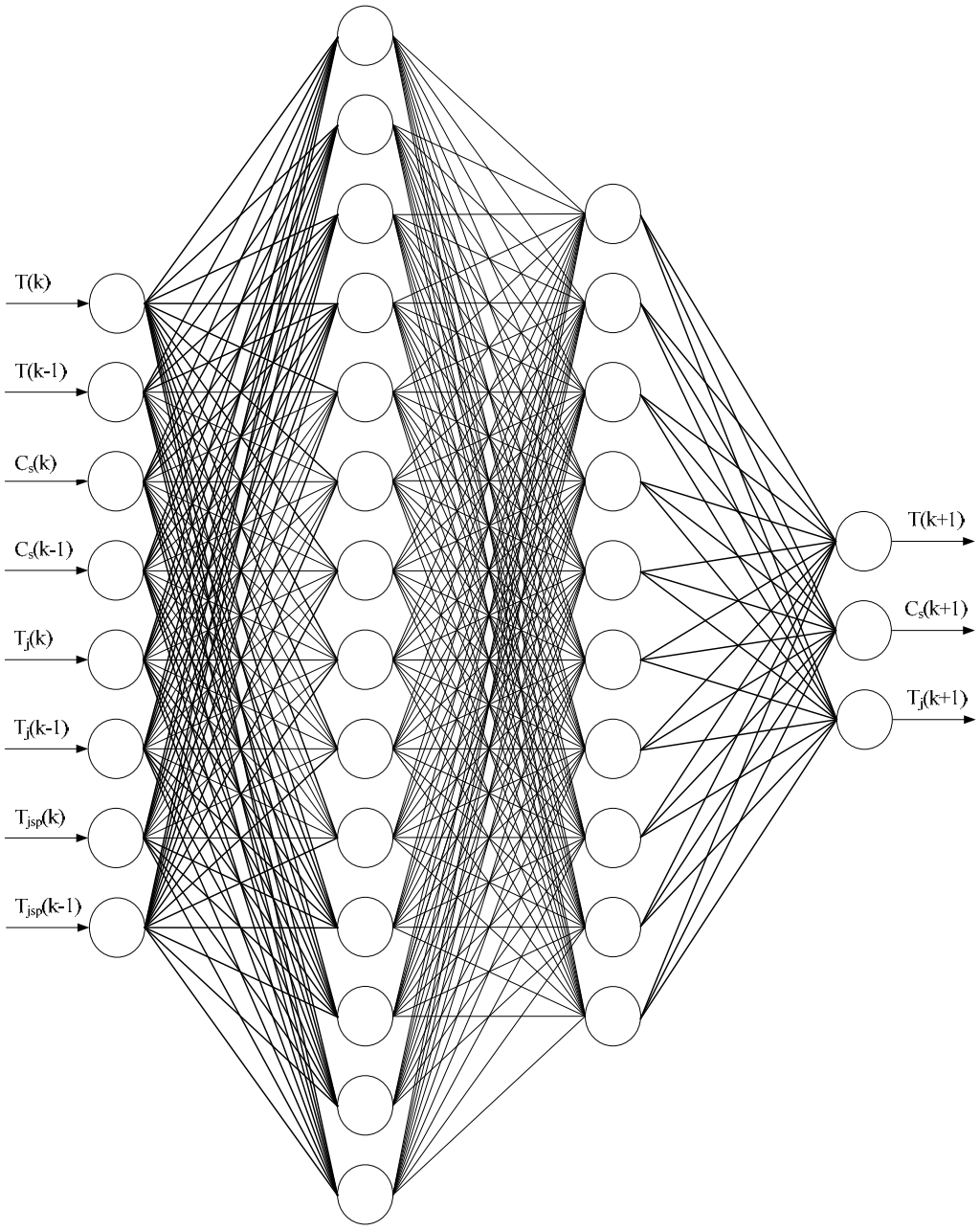
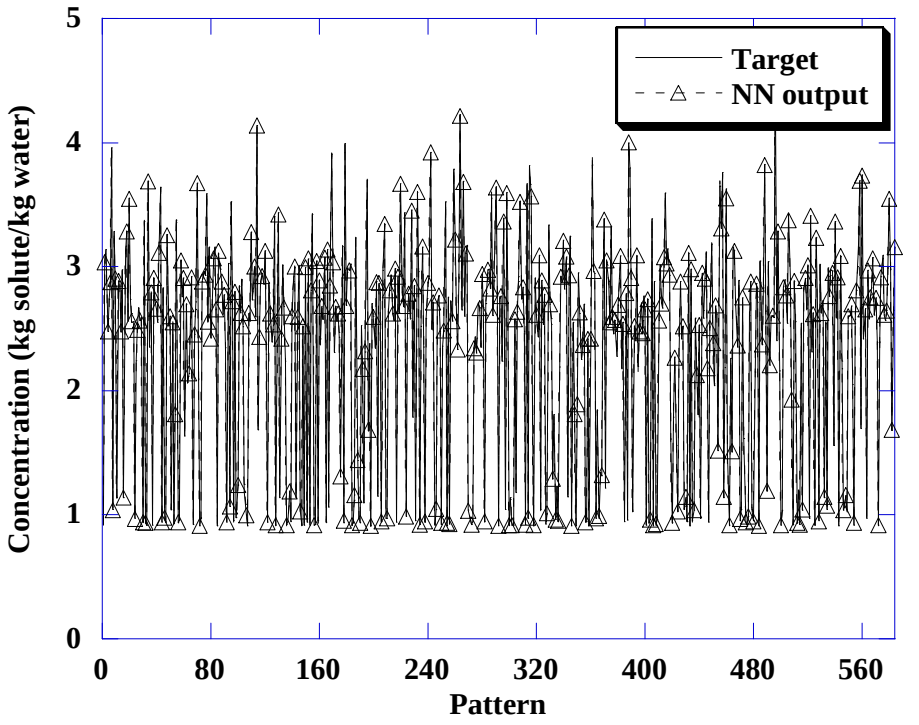
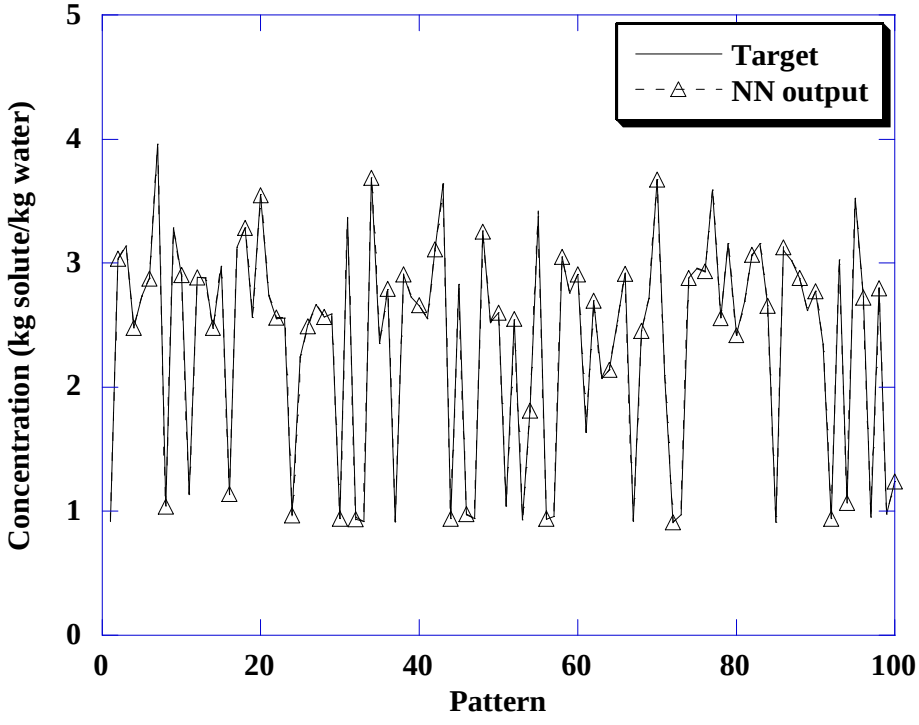


Figure 5.9 Neural network for forward model

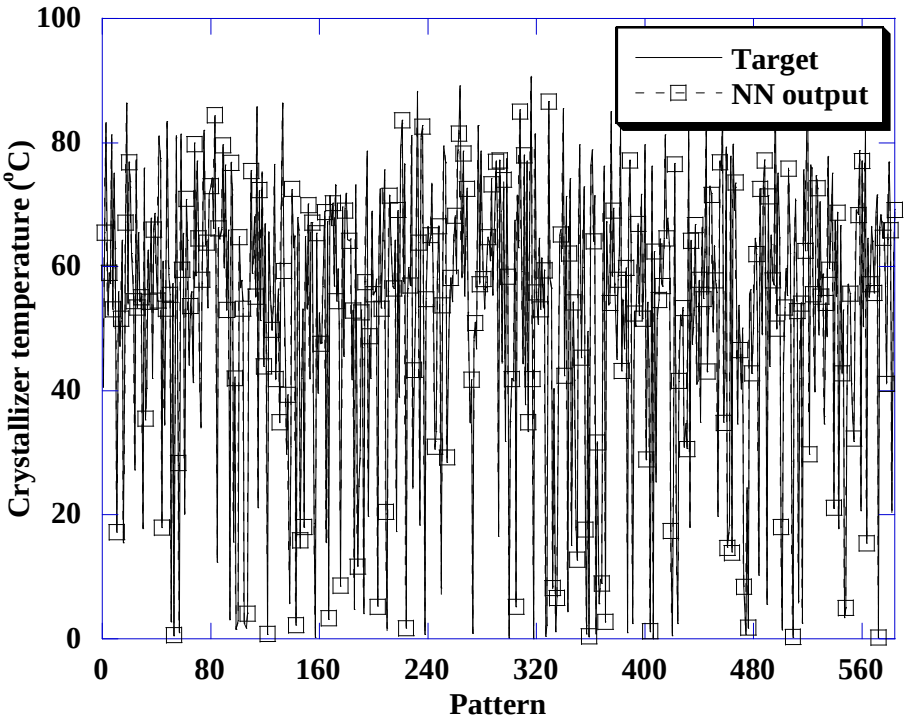


Whole

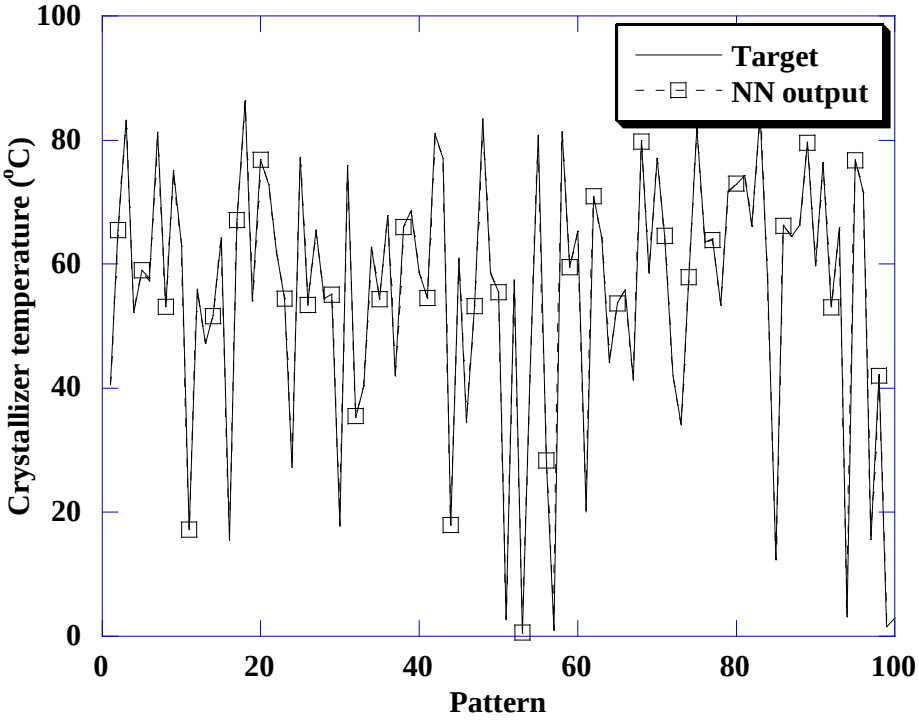


Zoom

Figure 5.10 Testing 1 results for the network prediction of concentration profile

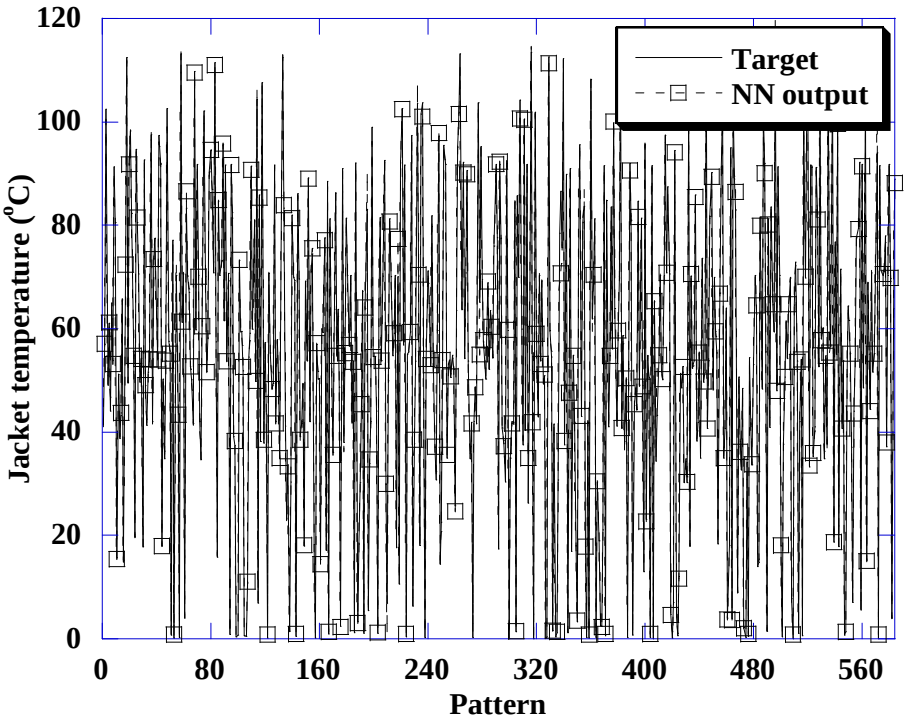


Whole

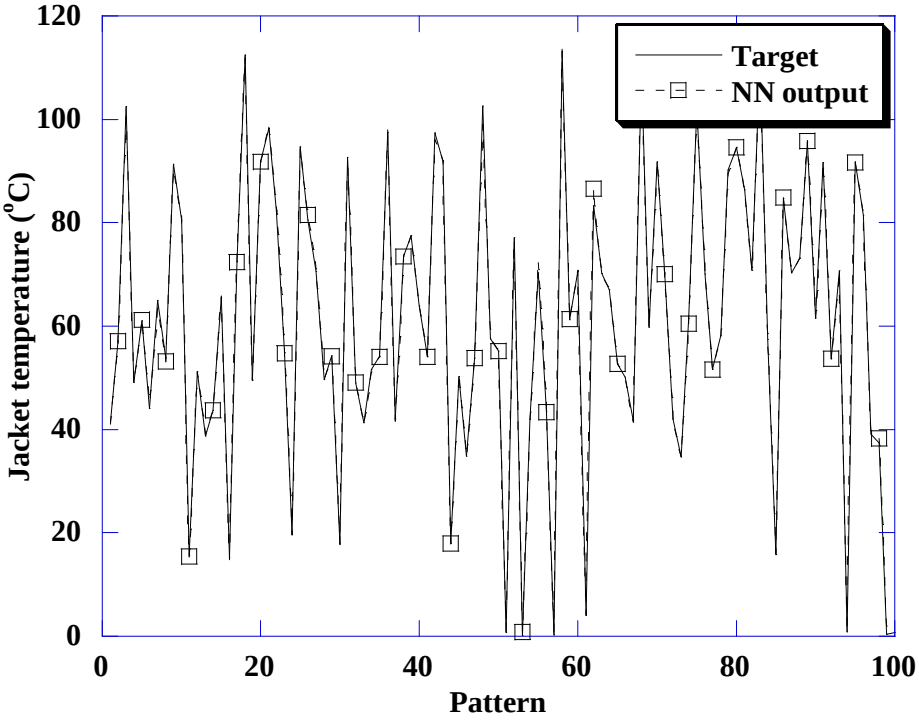


Zoom

Figure 5.11 Testing 1 results for the network prediction crystallizer temperature profile



Whole



Zoom

Figure 5.12 Testing 1 results for the network prediction jacket temperature profile

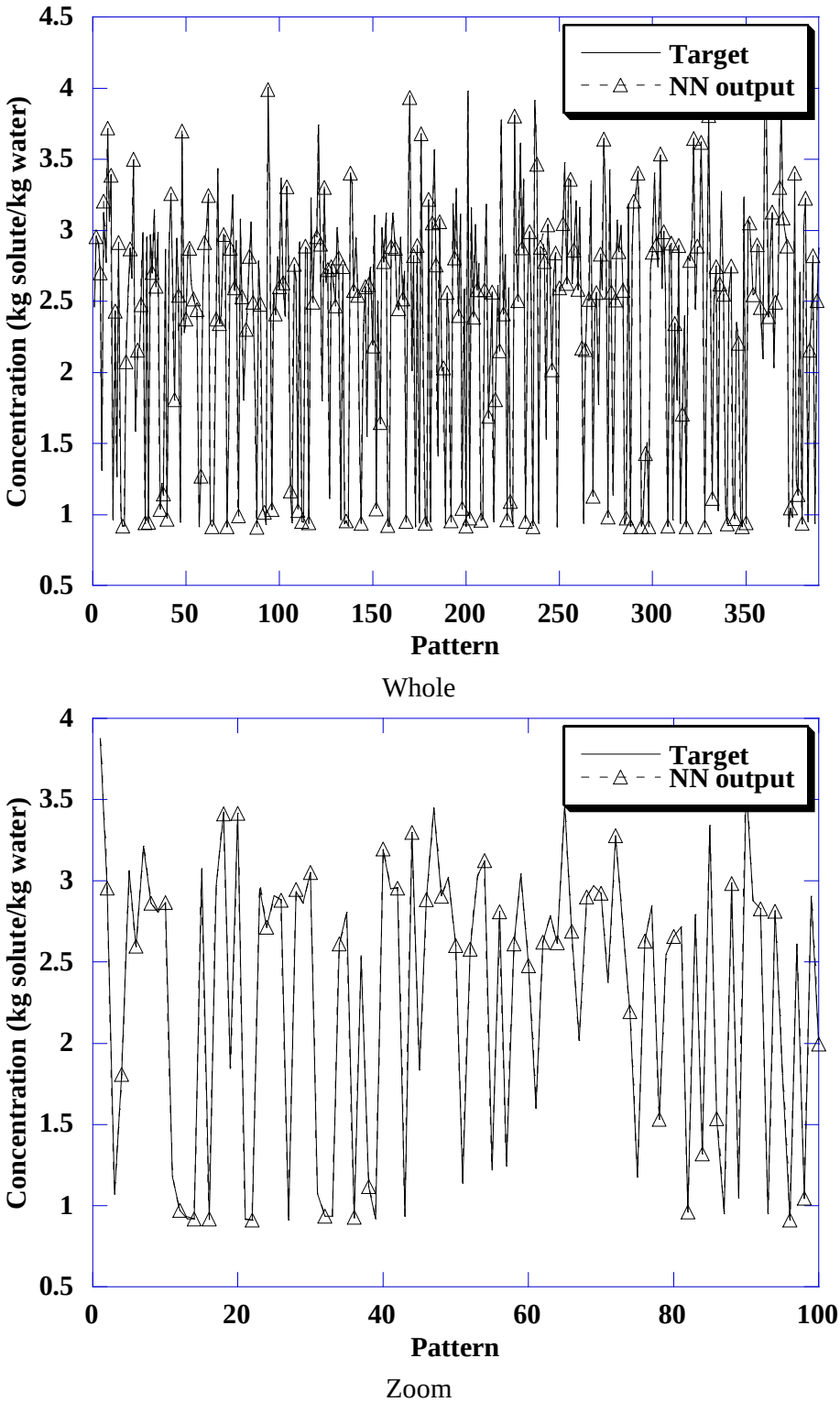


Figure 5.13 Validation results for the network prediction of concentration profile

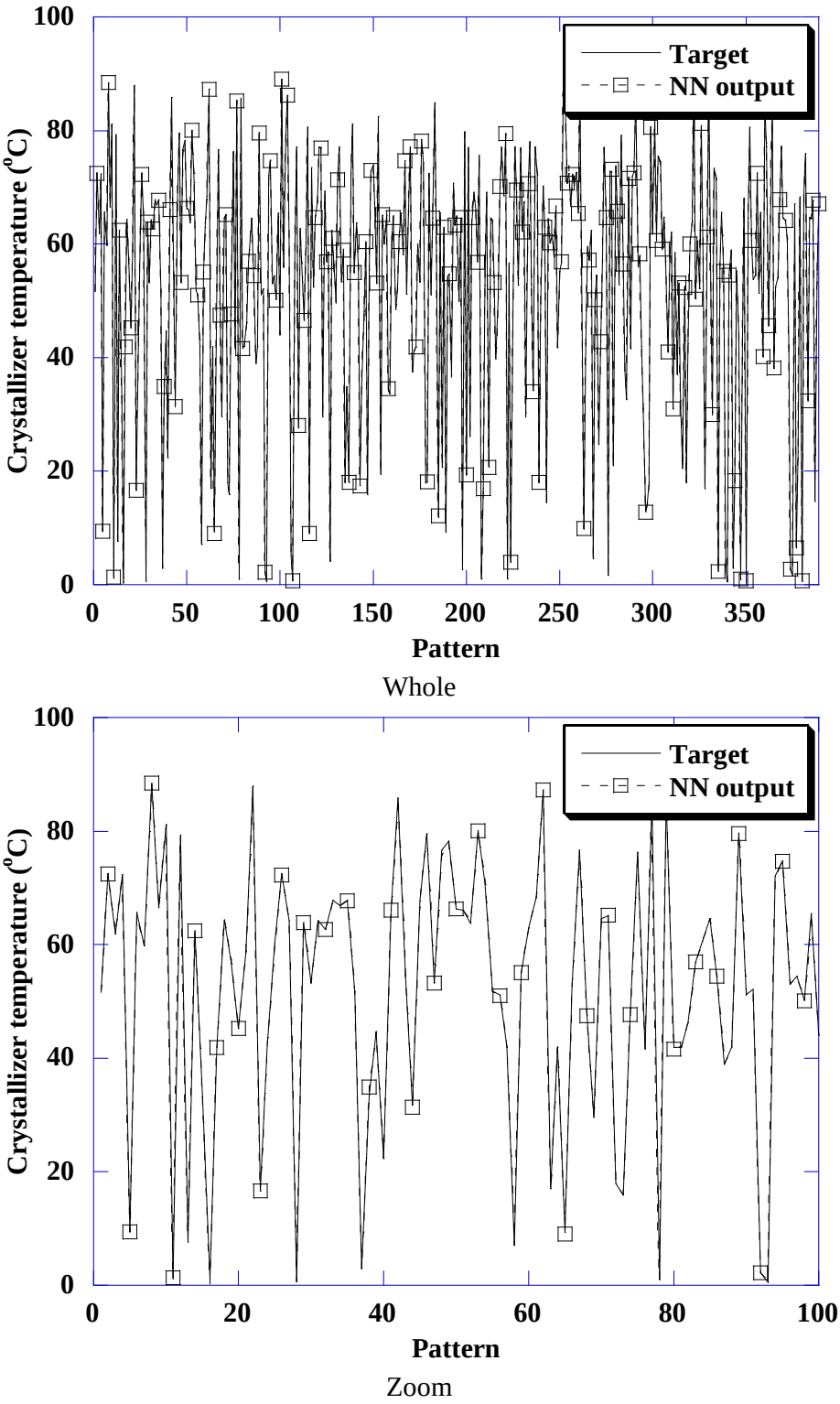
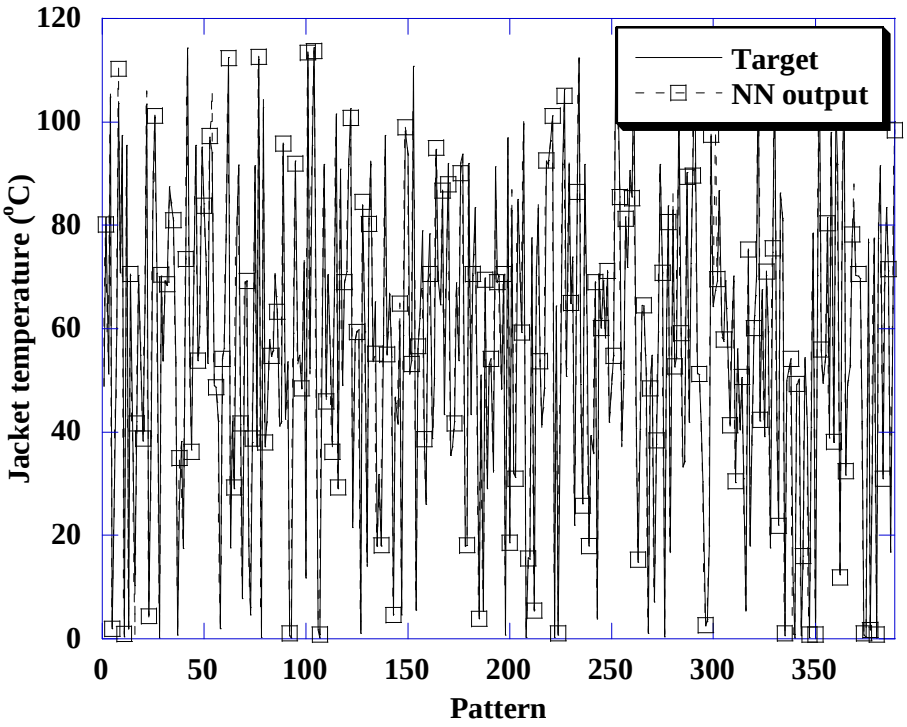
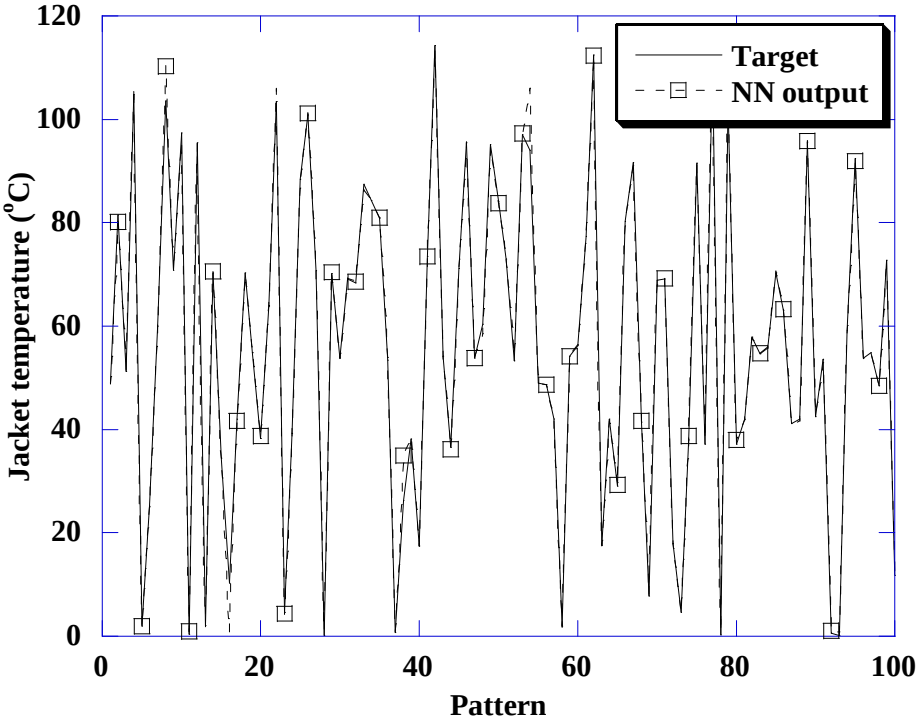


Figure 5.14 Validation results for the network prediction of crystallizer temperature



Whole



Zoom

Figure 5.15 Validation results for the network prediction of jacket temperature

The neural network forward model shows good accuracy for the prediction of the system. The data of concentration, the crystallizer temperature and the jacket temperature from mathematical model and predicted by the neural network are presented in Figures 5.10-5.15. The mean squared error index for training, testing and validation of the optimal neural network forward model are shown in Table 5.1.

Table 5.1 Mean squared error value of training, testing and validation in the neural network forward model

	No. of samples	Mean squared error
Training data	2334	7.5124×10^{-5}
Testing data 1	583	5.3528×10^{-5}
Testing data 2	583	8.0095×10^{-5}
Validation data	388	4.2852×10^{-5}

5.2 Neural network inverse model

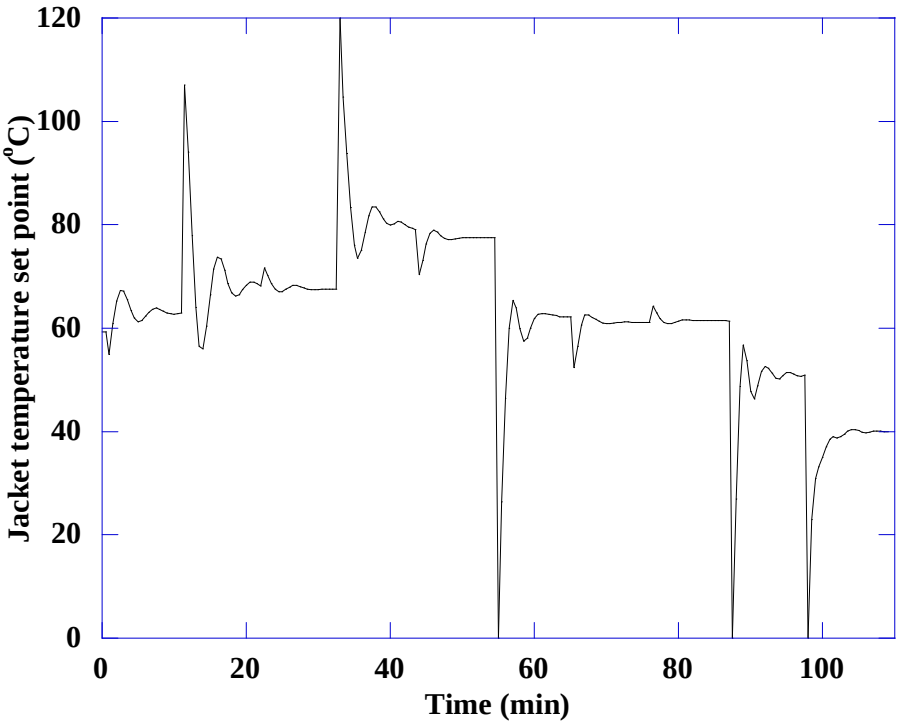
In this part, the neural network inverse model is used to control the crystallizer temperature. The steps of neural network structure designing are same as the neural network forward model (Figure 5.1). The neural network inverse model is applied to predict the present value of the manipulated variable (the jacket temperature set point, T_{jsp}). The input and output data pattern of the neural network inverse model is given in Figure 5.16

Neural network inverse model

error(k-1)	error(k)	error(k+1)	Output
T(k-1)	T(k)	T(k+1)	Input
C_s(k-1)	C_s(k)		
T_j(k-1)	T_j (k)		
T_{jsp}(k-1)	T_{jsp} (k)		

Figure 5.16 Input and output data pattern for inverse model

The input and output data sets are generated from closed-loop control that is feedback controller (PID) to train the neural network (Karniel et al., 2001; Nakanishi and Schaal, 2004; Gomi and Kawato, 2001) as shown in Figure 5.17-5.20. In this section, the generated data for network training are same the forward model which consist of plant certainty case and plant uncertainty cases that is compose of -30% U, +30% H_{crys},-30% H_{evap}, +30% k_g and +30% k_p. In addition, the data sets which are trained the neural network inverse model are 18 sets.



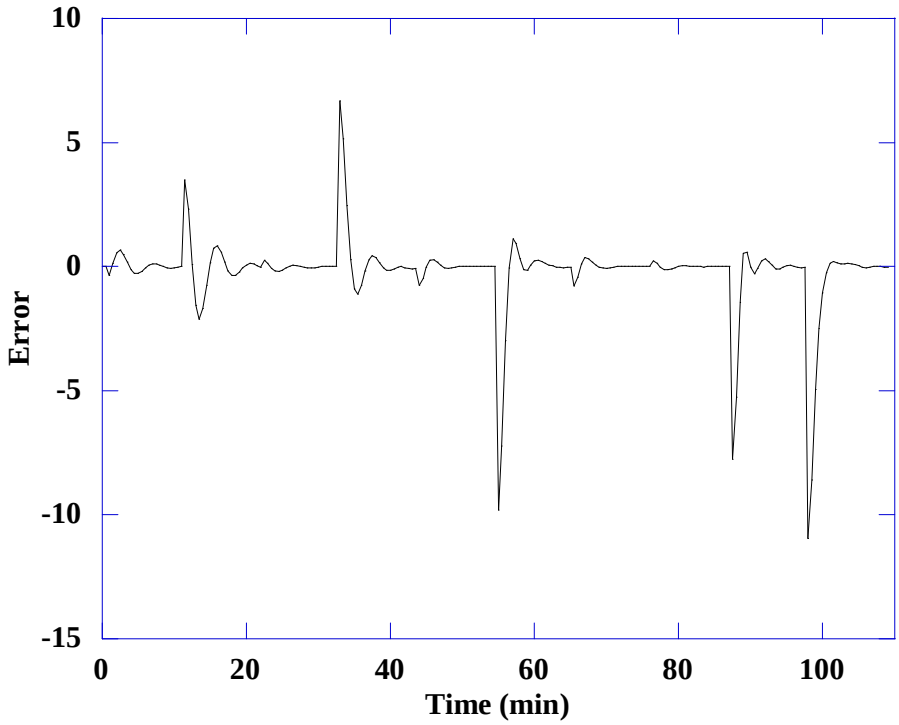
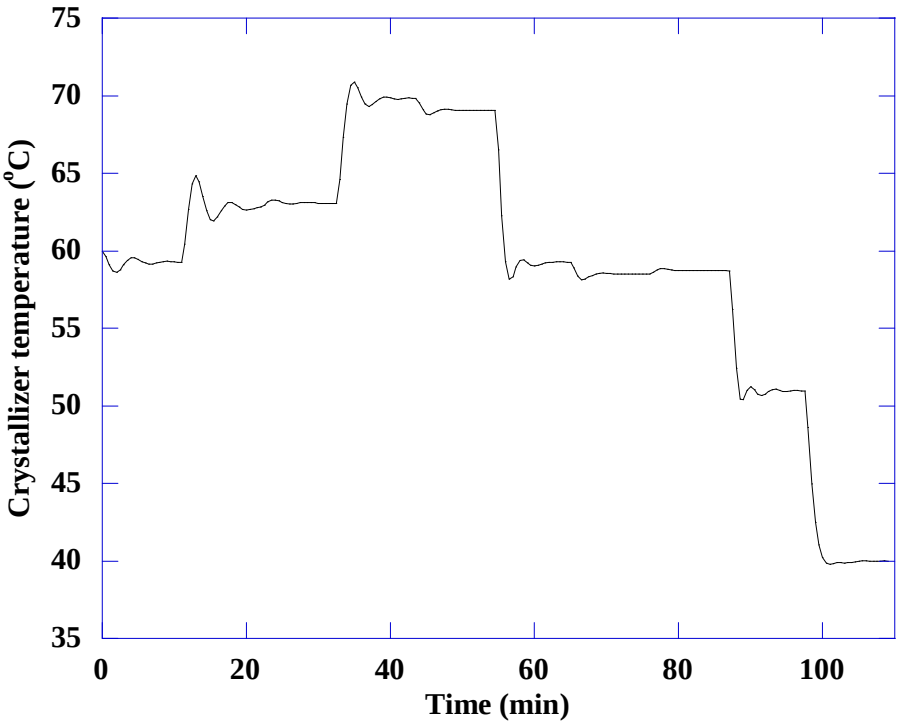


Figure 5.17 The jacket temperature set point and the error profile from closed-loop control of data set 1



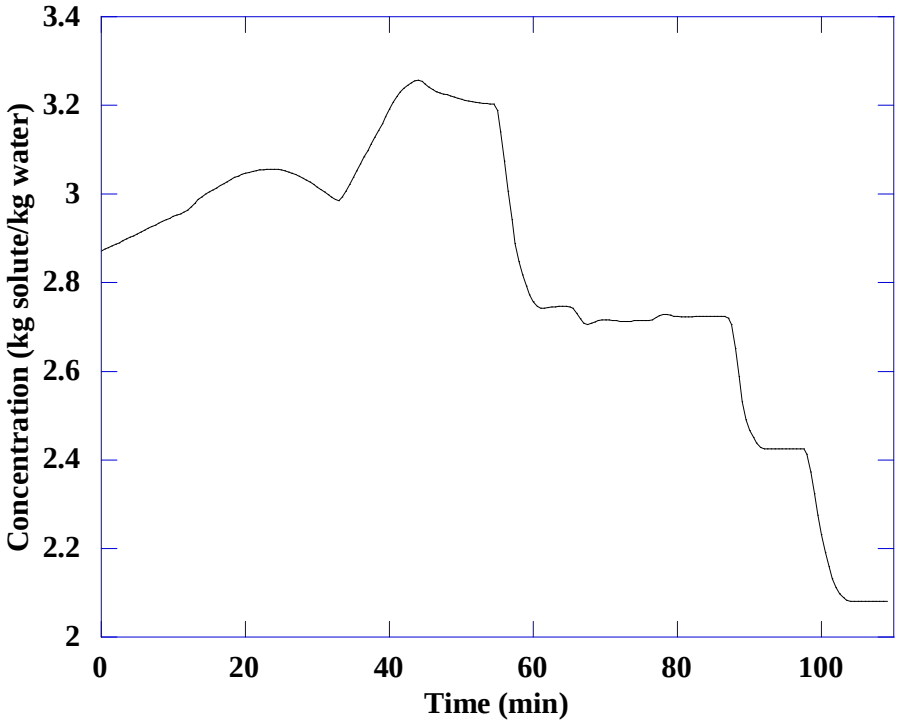
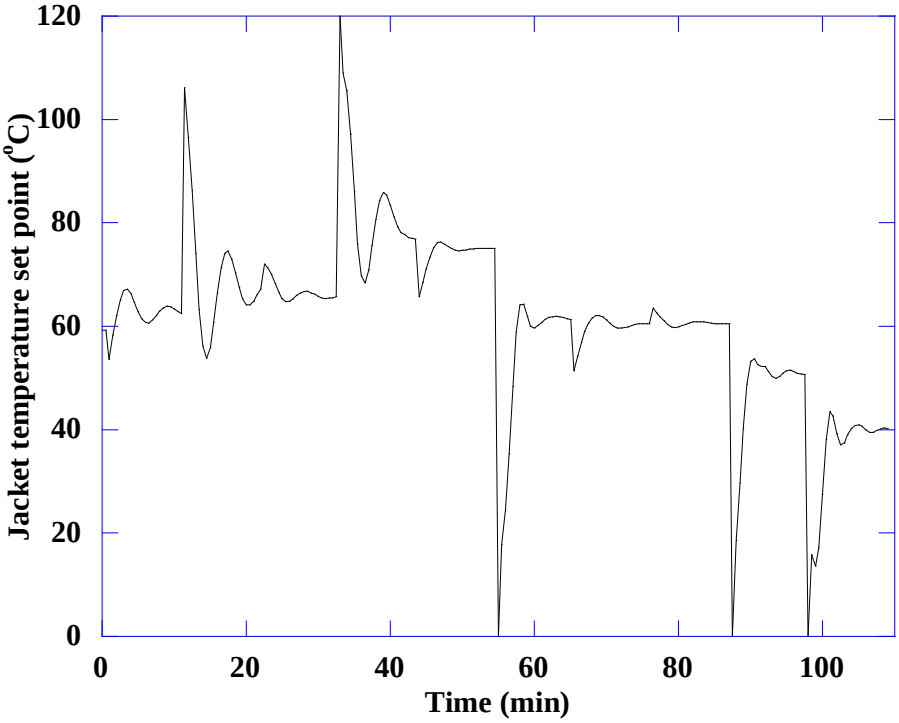


Figure 5.18 The crystallizer temperature and the concentration profile from closed-loop control of data set 1



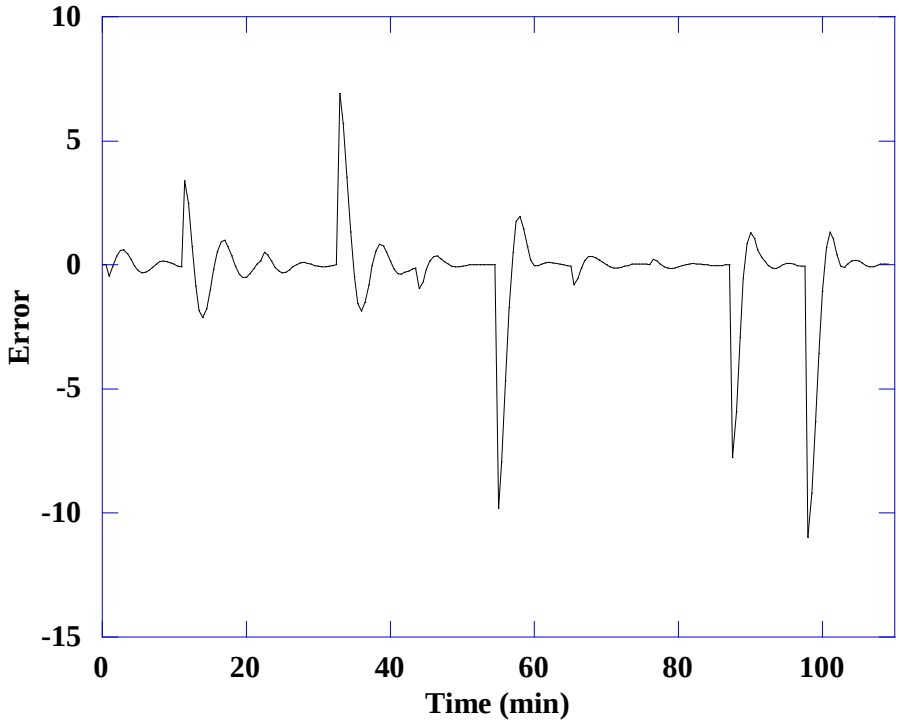
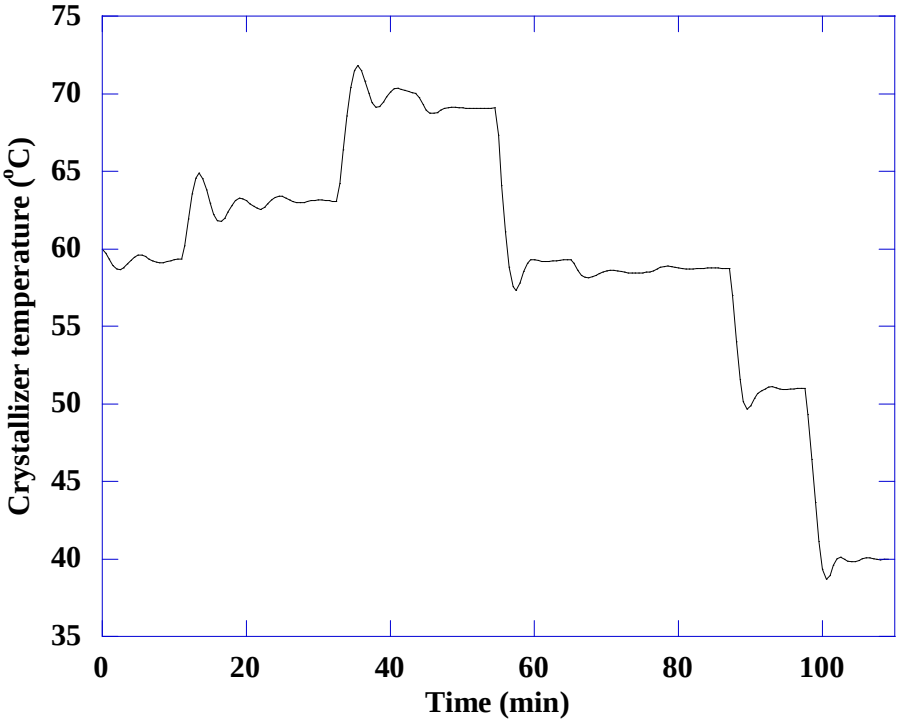


Figure 5.19 The jacket temperature set point and the error profile from closed-loop control of data set 2



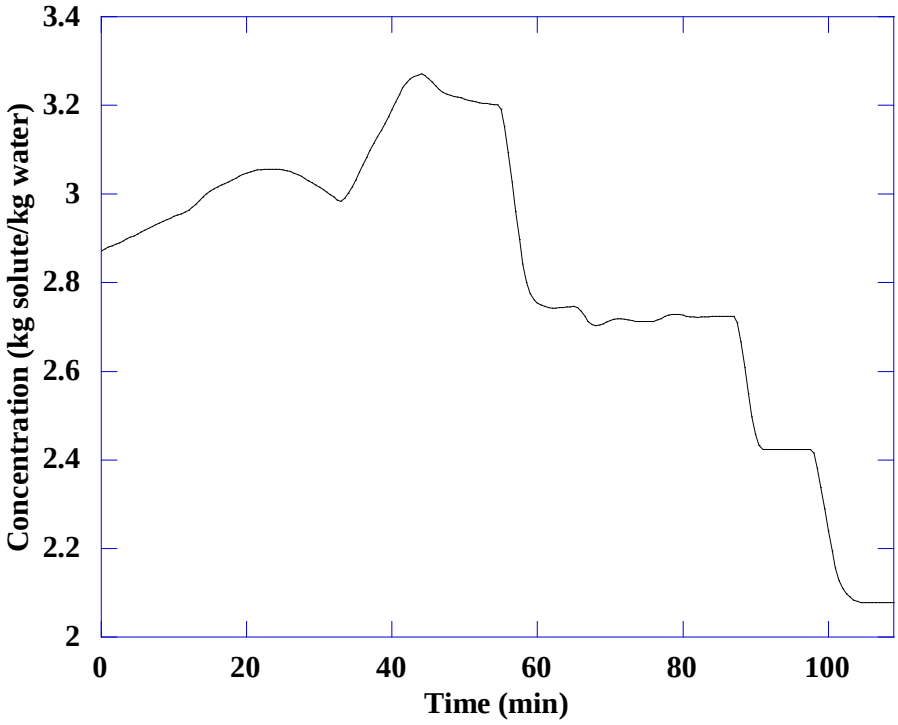
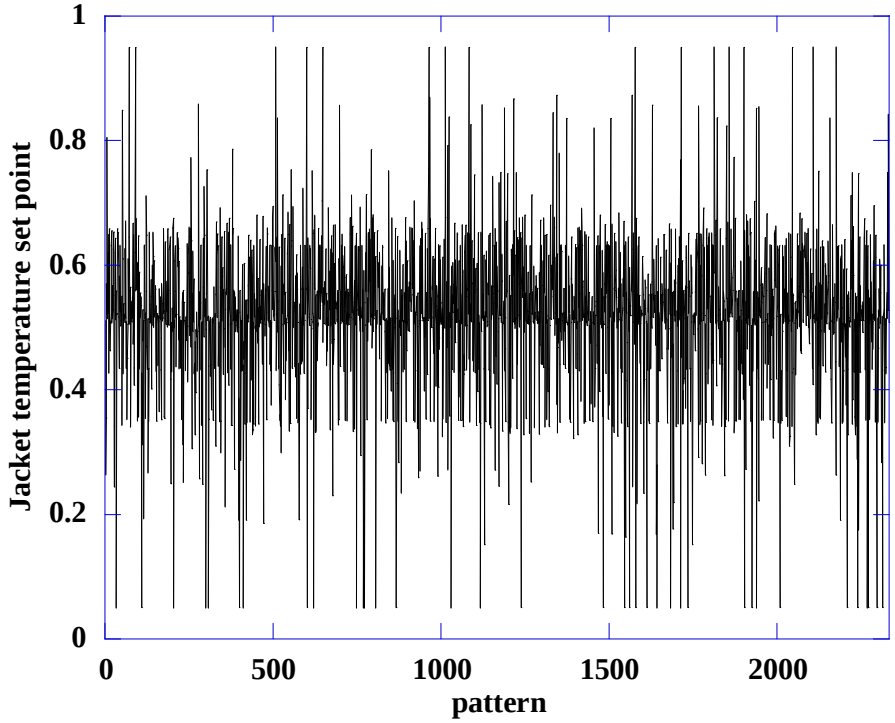


Figure 5.20 The crystallizer temperature and the concentration profile from closed-loop control of data set 2



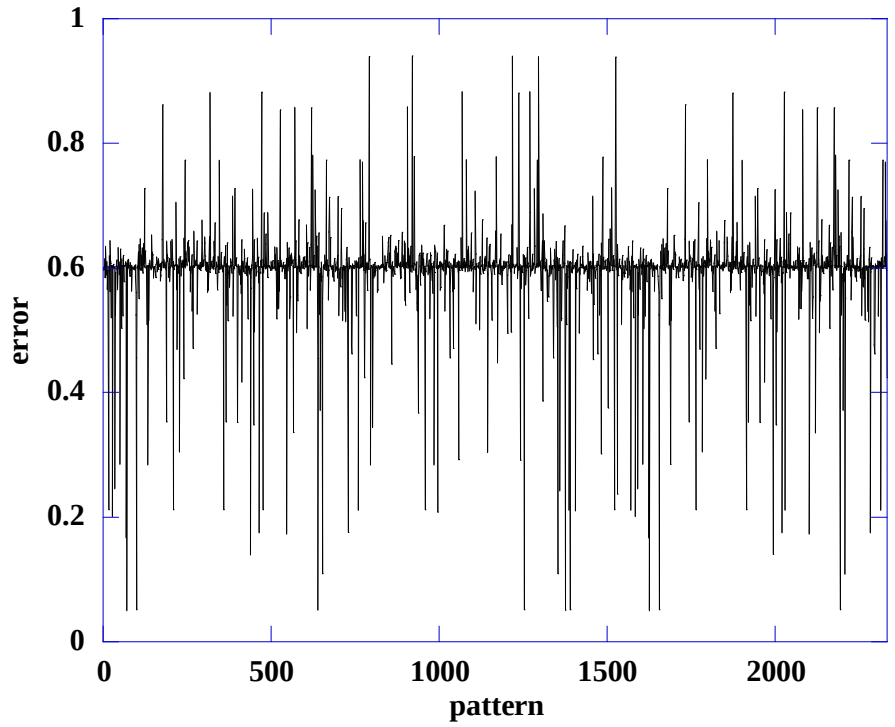
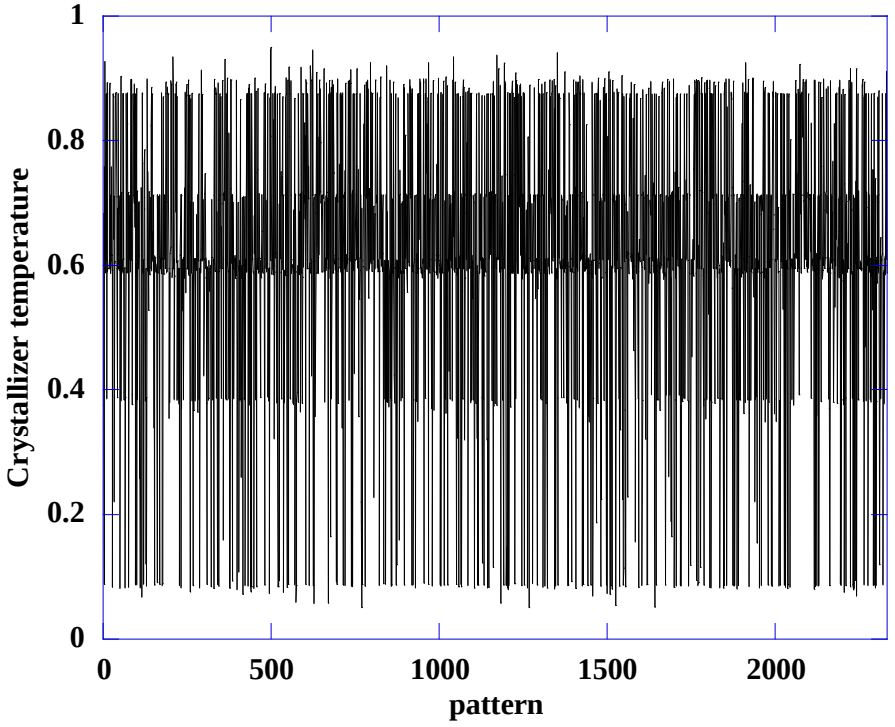


Figure 5.21 The jacket temperature set point and the error profile from closed-loop control of sum training data



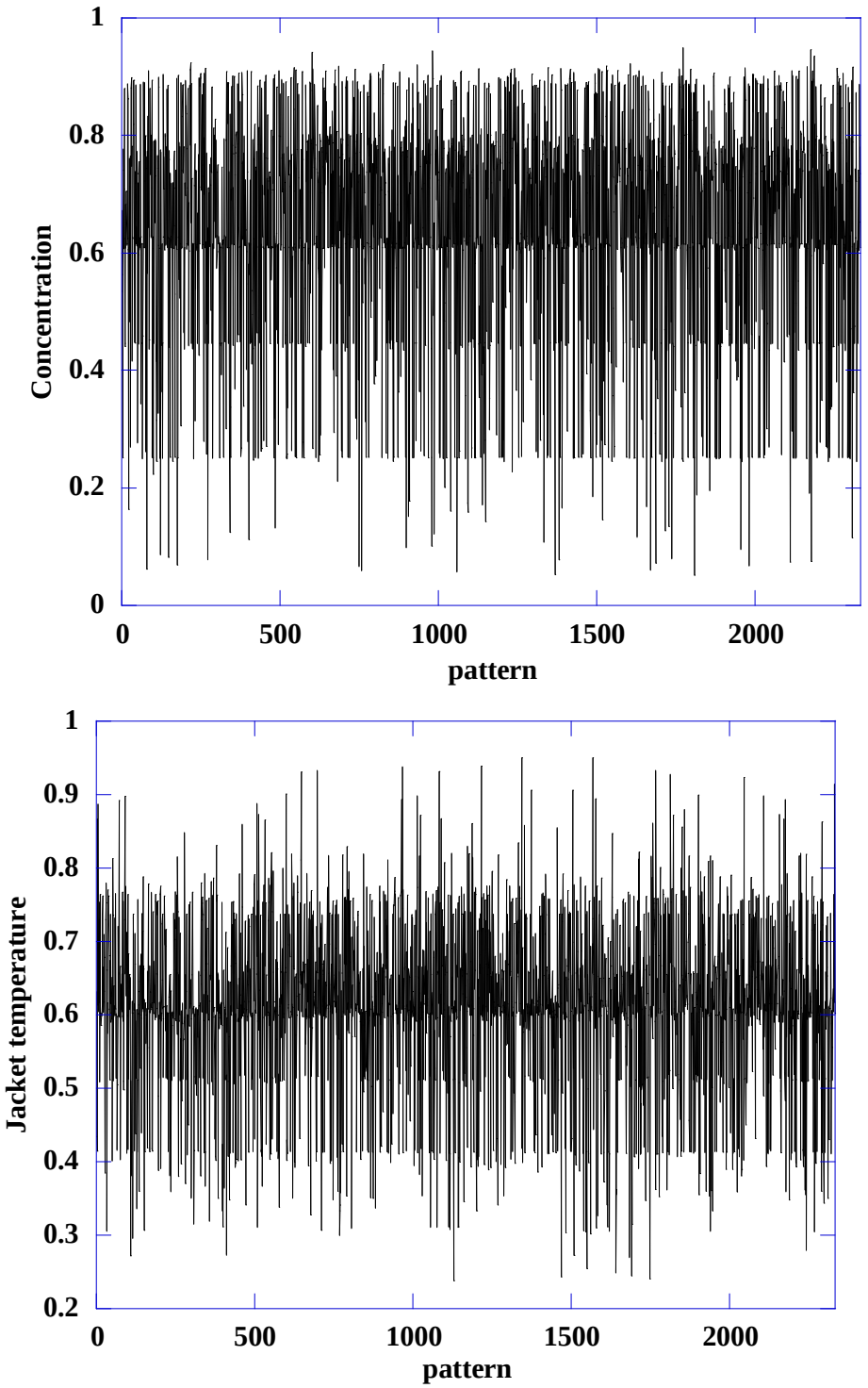


Figure 5.22 The crystallizer temperature, the concentration and the jacket temperature profile from closed-loop control of sum training data set

The data sets which are generated from mathematical model are normalized by Eq. 5.2. Then the data sets are combined and randomized. The data sets are divided into 3 groups which consist of 60% training data set, 30% testing data set and 10% validating data set. The data sets for training the network are shown in Figures 5.21-5.22.

The optimal neural network structure of the inverse model is shown in Figure 5.23 which is 11 input nodes that consists of error(k-1), error(k), error(k+1), C_S(k-1), C_S(k), T(k-1), T(k), T(k+1), T_j(k-1), T_j(k) and T_{jsp}(k-1) as well as 1 output nodes that is T_{jsp}(k+1).

In this part, the optimal structure of the neural network inverse model consists of 11 nodes in the input layer, 12 nodes in the first hidden layer, 8 nodes in the second hidden layer and 1 node in the output layer which is shown in Figure 5.23. The neural network inverse model show good accuracy for the prediction of the jacket temperature set point, T_{jsp}. Figures 5.24-5.26 show the profile of the jacket temperature set point which calculate from the mathematical model and the neural network inverse model. The mean squared error index of training, testing and validation of the optimal neural network inverse model are shown in Table 5.2.

Table 5.2 Mean squared error value of training, testing and validation in the neural network inverse model

	No. of samples	Mean squared error (MSE)
Training data	2334	3.0576×10^{-5}
Testing data 1	583	9.4699×10^{-6}
Testing data 2	583	8.3586×10^{-6}
Validation data	388	1.1091×10^{-5}

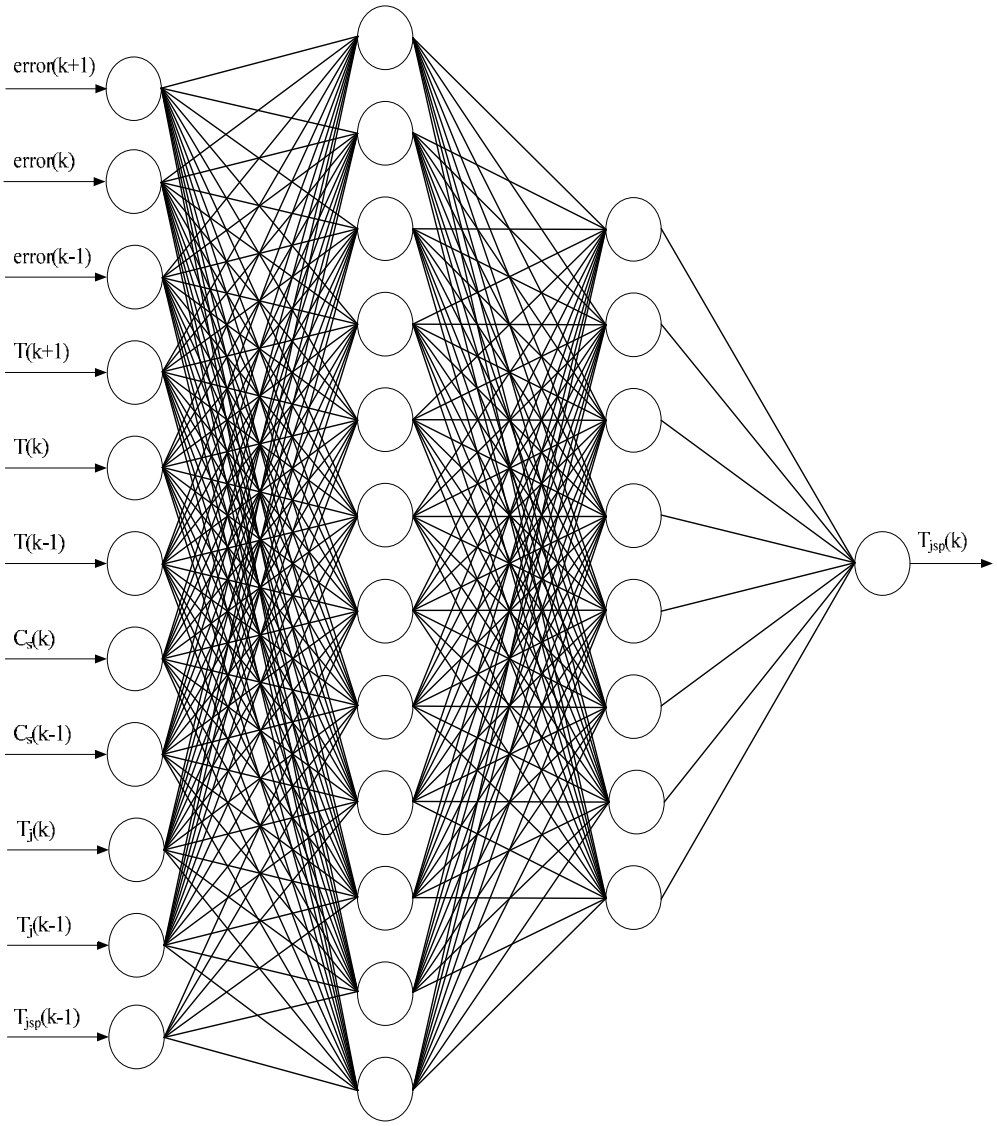
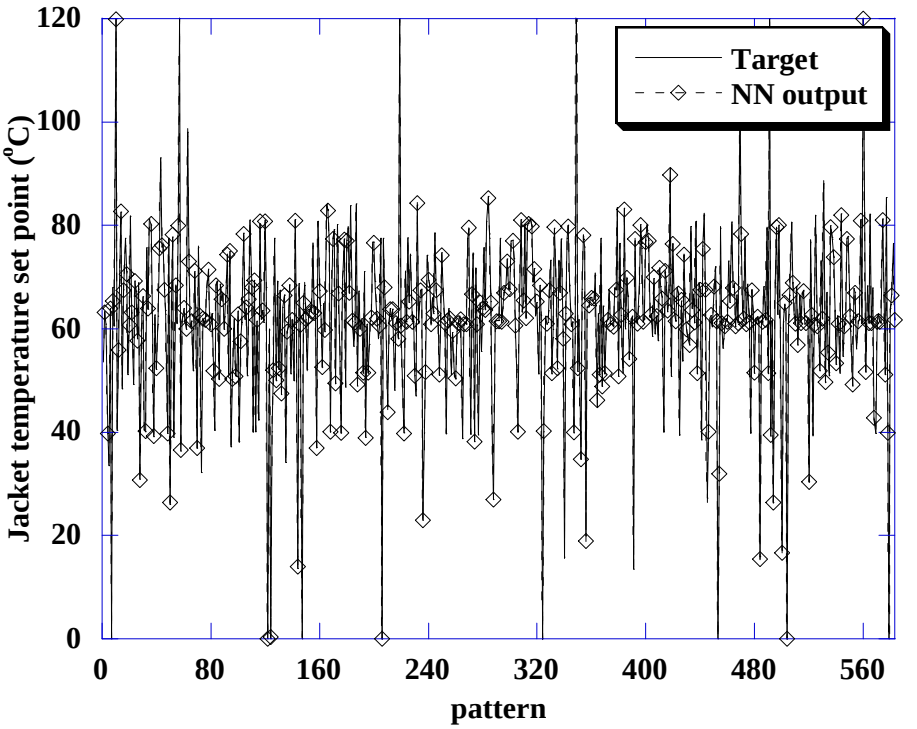
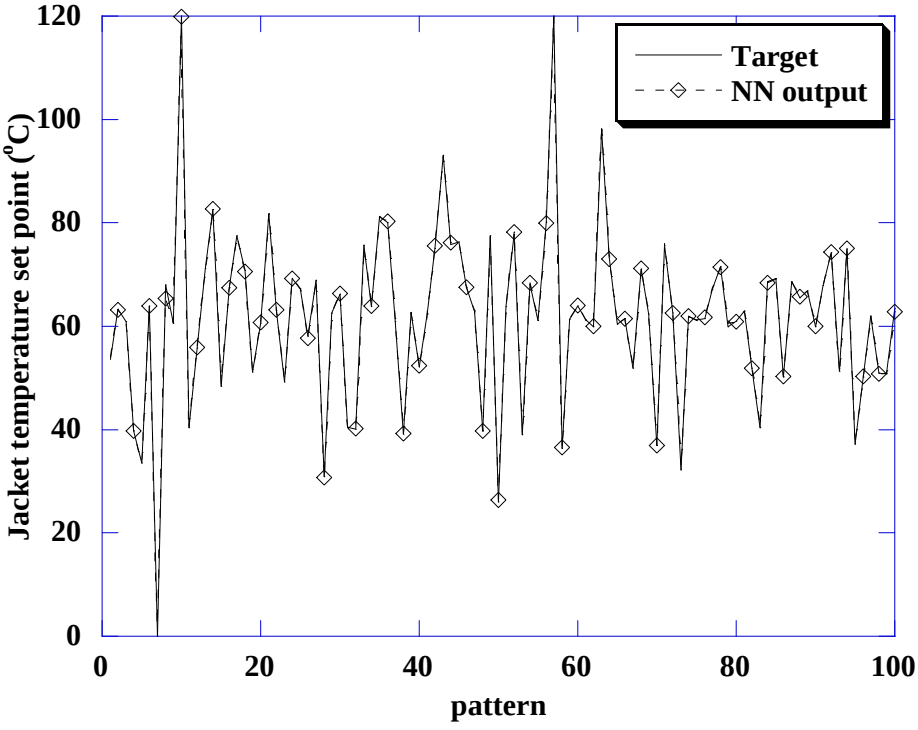


Figure 5.23 Optimal structure of neural network for inverse model

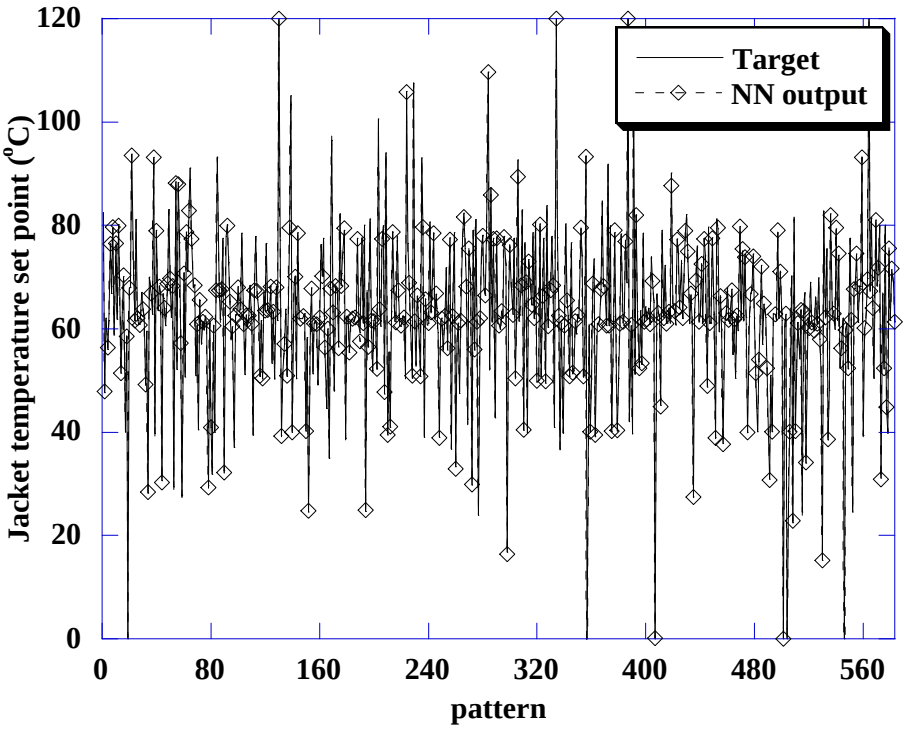


Whole

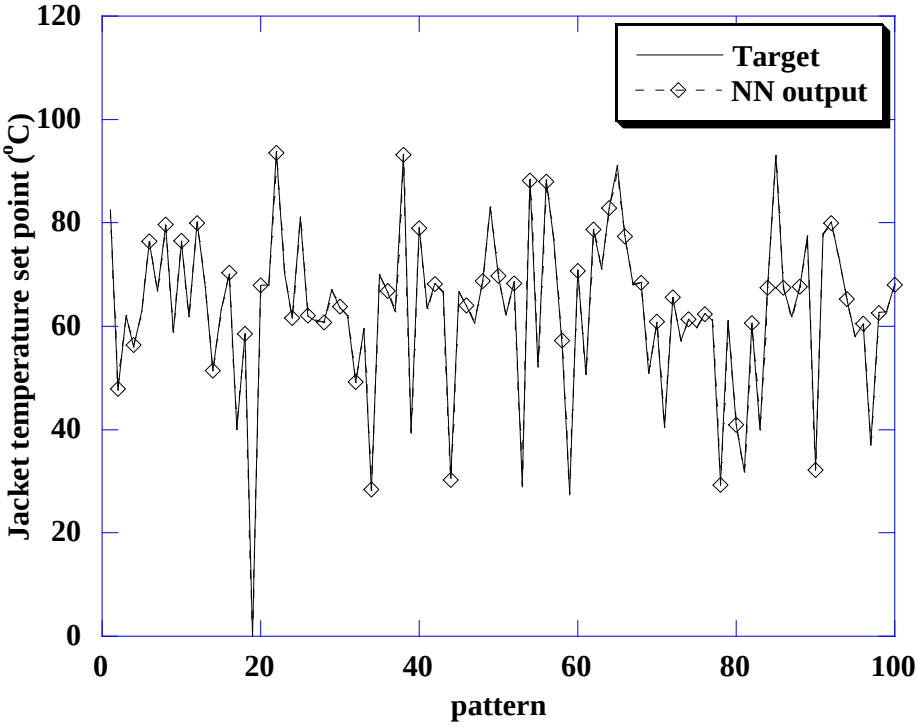


Zoom

Figure 5.24 Testing 1 results for the network prediction of the jacket temperature set point profile



Whole



Zoom

Figure 5.25 Testing 2 results for the network prediction of the jacket temperature set point profile

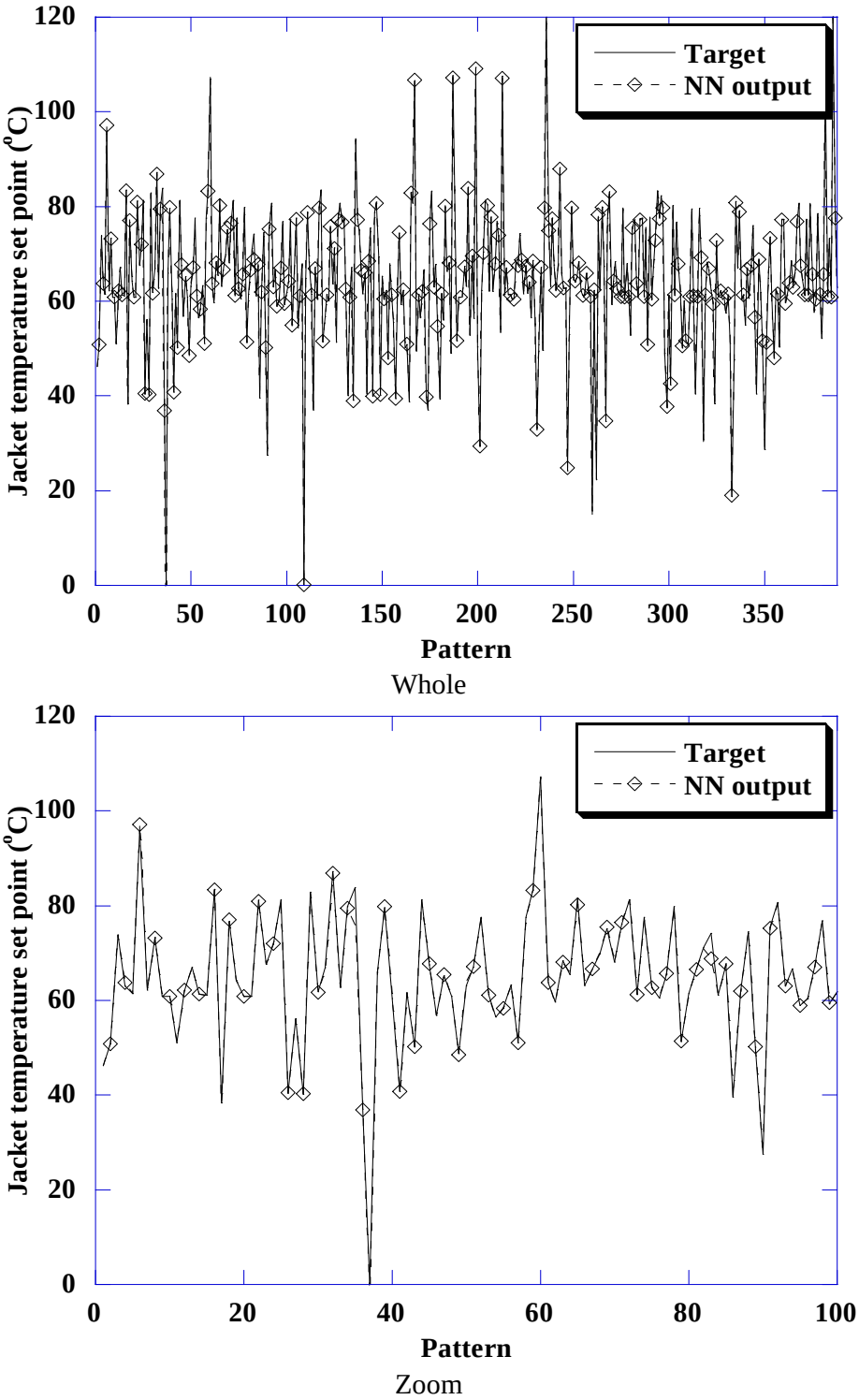


Figure 5.26 Validating results for the network prediction of the jacket temperature set point profile

CHAPTER VI

THE CONTROLLER DESIGN BASED NEURAL NETWORK FOR THE TEMPERATURE CONTROL OF THE PROCESS

In this section, the optimal temperature profile in chapter 4 is applied in PID, NNDIC and NNMPC controllers which control the crystallizer temperature to follow the desired profile by operating the set point of jacket temperature. The optimal crystallizer temperature profiles are shown in Figures 4.2-4.3. In this case, the optimization 1 (OPT1) is chosen to study.

6.1. Neural Network Model Based Predictive Controller

Model predictive control using a neural network has been focused by some researchers (Kittisupakorn et al., 2009; Yu and Gomm, 2003; Ławryńczuk, 2008; Georgieva and Azevedo, 2007). The neural network model based predictive control strategy in this work is shown in Figure 6.1. The neural network forward model is used to predict the future process outputs over the prediction horizon (*p*). The predictions are passed to the optimization which minimizes a specified objective function. Sequential quadratic programming (SQP) algorithm is used to solve the optimization problem of minimizing in Eq.6.1. The objective function of the model predictive control strategy is described as follows:

$$\min_{T_{jsp}} \sum_{i=1}^{p} \left[W_1 \{ T(k+i) - T_{sp}(k+i) \}^2 + W_2 \{ \Delta T_{jsp} \}^2 \right] \tag{6.1}$$

Subject to

$$25\ ^{\circ}\mathrm{C} \le T_{jsp}(k+i) \le 100\ ^{\circ}\mathrm{C}, \qquad i = 1,2,3,\ldots,p \tag{6.2}$$

$$T(k+p) = T_{sp}(k+p) \tag{6.3}$$

where p is a parameter specifying the prediction horizon, W_1 is weighting parameter used to give different weights to different squared tracking error, W_2 is weighting parameter of different squared of manipulated variable and T_{sp} is the set point of crystallizer temperature as obtained from optimization.

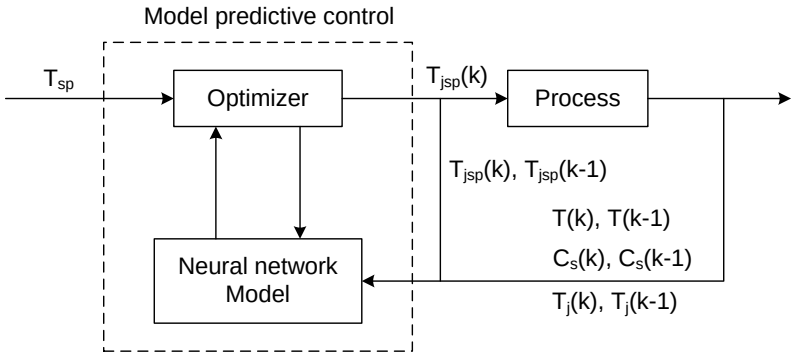


Figure 6.1 The neural network based model predictive control strategy

The simulations have been done using the neural network model to find a set of suitable control parameter W_1, W_2, p and m. The prediction horizon p corresponds to the future time interval used to compute prediction with the neural network model. The control horizon is equal or less than the prediction horizon ($m \leq p$). In the same IAE performance case, the choice of W_1 and W_2 are chosen 1 and 1, respectively. The prediction horizon and control horizon are set as 6 and 6, respectively. In the best performance case, the choice of W_1 and W_2 are chosen 3 and 3, respectively. The prediction horizon and control horizon are set as 10 and 1, respectively.

6.2. Neural Network Direct Inverse Controller

In this part, the neural network inverse model in chapter 5 is applied to control the crystallizer temperature. The optimal neural network inverse model (11-12-8-1 structure) is utilized to predict the manipulated variable (jacket temperature set point). The prediction of manipulated variable for control requires the past and present values of the process outputs and the past values of manipulated variable as well as it requires the future value of crystallizer temperature set point. Figure 6.2 shows the structure of NNDIC strategy for controlling of the crystallizer temperature.

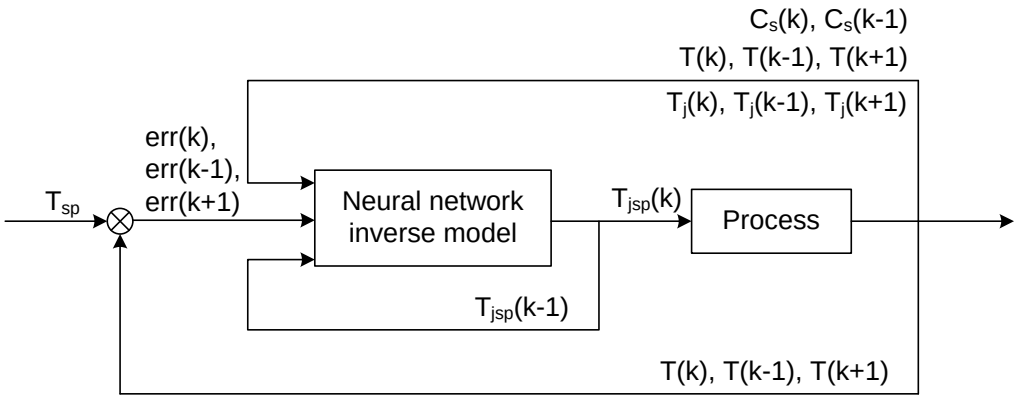


Figure 6.2 Neural network direct inverse control strategy

6.3. Comparisons of Controller in Nominal and Mismatch Cases

In this section, the comparisons of controller are based on 2 instances which consist of the same IAE performance and the best performance of controller. In each instance, the simulations are divided into 2 cases, the first case is a nominal case and the second case is a parameter mismatch case.

6.3.1. Comparisons of the Same Controller Performance

Nominal case

First, the simulations are investigated in nominal case which all parameters in Table 4.1 are exact. The PID, NNDIC and NNMPC controllers are tuned as the same performance which is integral absolute error (IAE). Figures 6.3-6.5 show the response profiles of the crystallizer temperature and the jacket temperature set point. It can be seen that the control response of the PID and NNDIC controllers are corresponding. The PID and NNDIC controllers give some overshoot of the control variable (T) but the response of the control variable of the NNMPC controller is a little overshoot. The required time for the control variable to reach and remain to the set point of the NNMPC controller is less than the NNDIC and PID controllers. In term of the offset, the NNDIC controller give some offset which shows in Figure 6.4 but the NNMPC and PID controllers give a little offset. The response of the manipulating variable (T_{jsp}) of the NNMPC controller is smoother than the NNDIC and PID controllers.

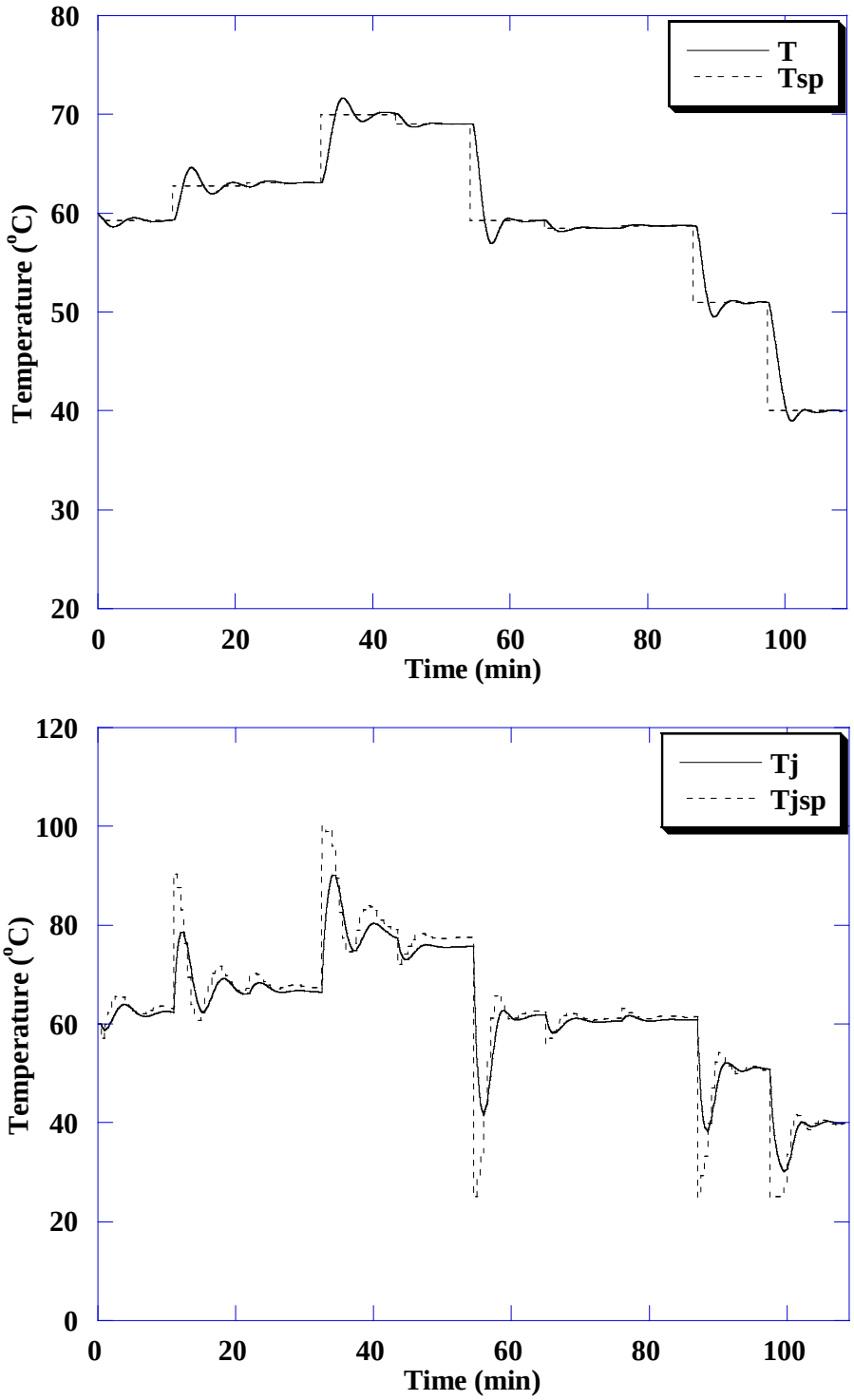


Figure 6.3 The crystallizer temperature control and the manipulated variable (T_{jsp}) in nominal case using PID controller

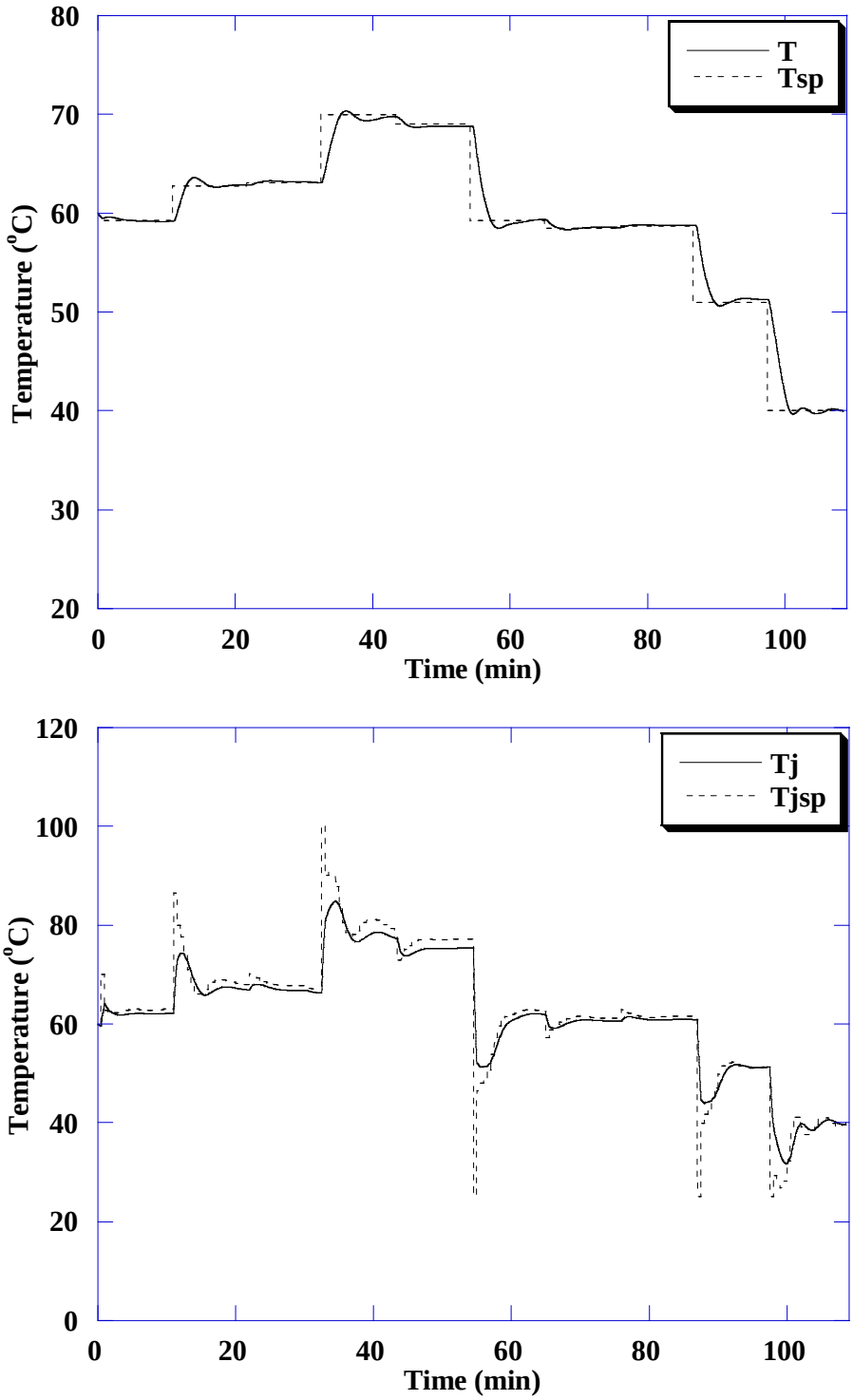


Figure 6.4 The crystallizer temperature control and the manipulated variable (T_{jsp}) in nominal case using NNDIC controller

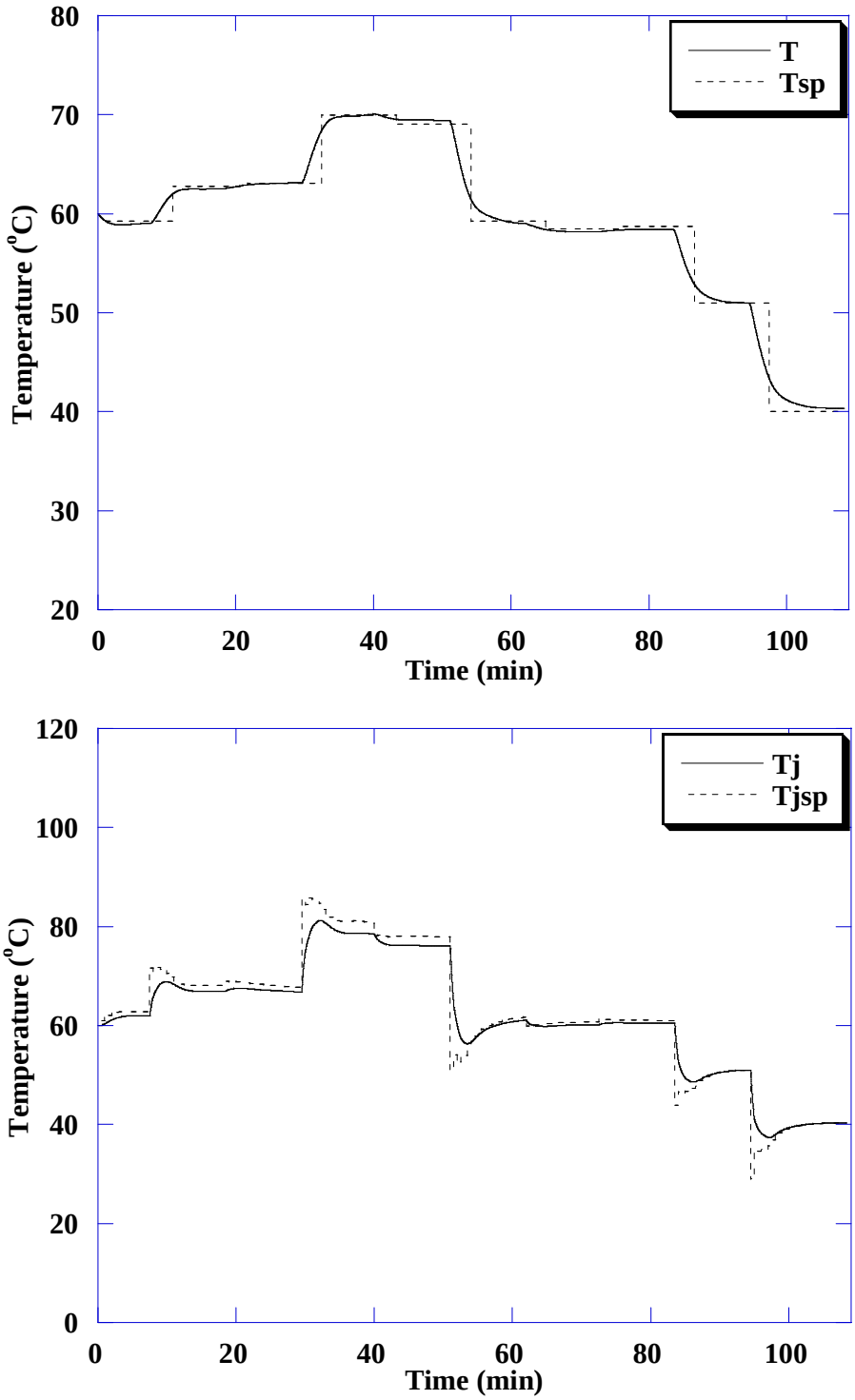


Figure 6.5 The crystallizer temperature control and the manipulated variable (T_{jsp}) in nominal case using NNMPC controller

Parameter mismatch case

In this case, the performance of controller is investigated with parameters mismatch to test the robustness of controller (Kittisupakorn et al., 2009; Daosud et al., 2005). The overall heat transfer coefficient, heat of crystallization, heat of vaporization, nucleation rate and crystal growth rate are considered as the parameter mismatch. The percent choosing for mismatch parameters is considered the percent changing from its nominal values that affect to the process response. The parameters mismatch are divided into 5 parameters consisting of decreasing 30% of U, increasing 30% of H_{crys}, decreasing 30% of H_{evap}, increasing 30% of k_p and increasing 30% of k_g from its nominal value. For robustness testing of the controller, the parameters mismatch cases are divided 6 cases which are shown in Table 6.1. Figures 6.6-6.11 show the response profiles of the crystallizer temperature and the jacket temperature set point.

Table 6.1 Performance of the same performance controller for nominal and parameters mismatch cases

Cases	IAE values		
	NNMPC	NNDIC	PID
Nominal	79.55	79.85	79.32
-30%U	82.06	93.76	98.93
+30%k_p	80.22	80.29	79.38
+30%k_g	79.97	81.12	79.68
+30%H_{crys}	79.98	80.87	80.43
-30%H_{evap},	80.05	80.01	81.67
-30%U, +30%H_{crys}, -30%H_{evap}, +30%k_g, +30%k_p	83.86	98.13	106.76

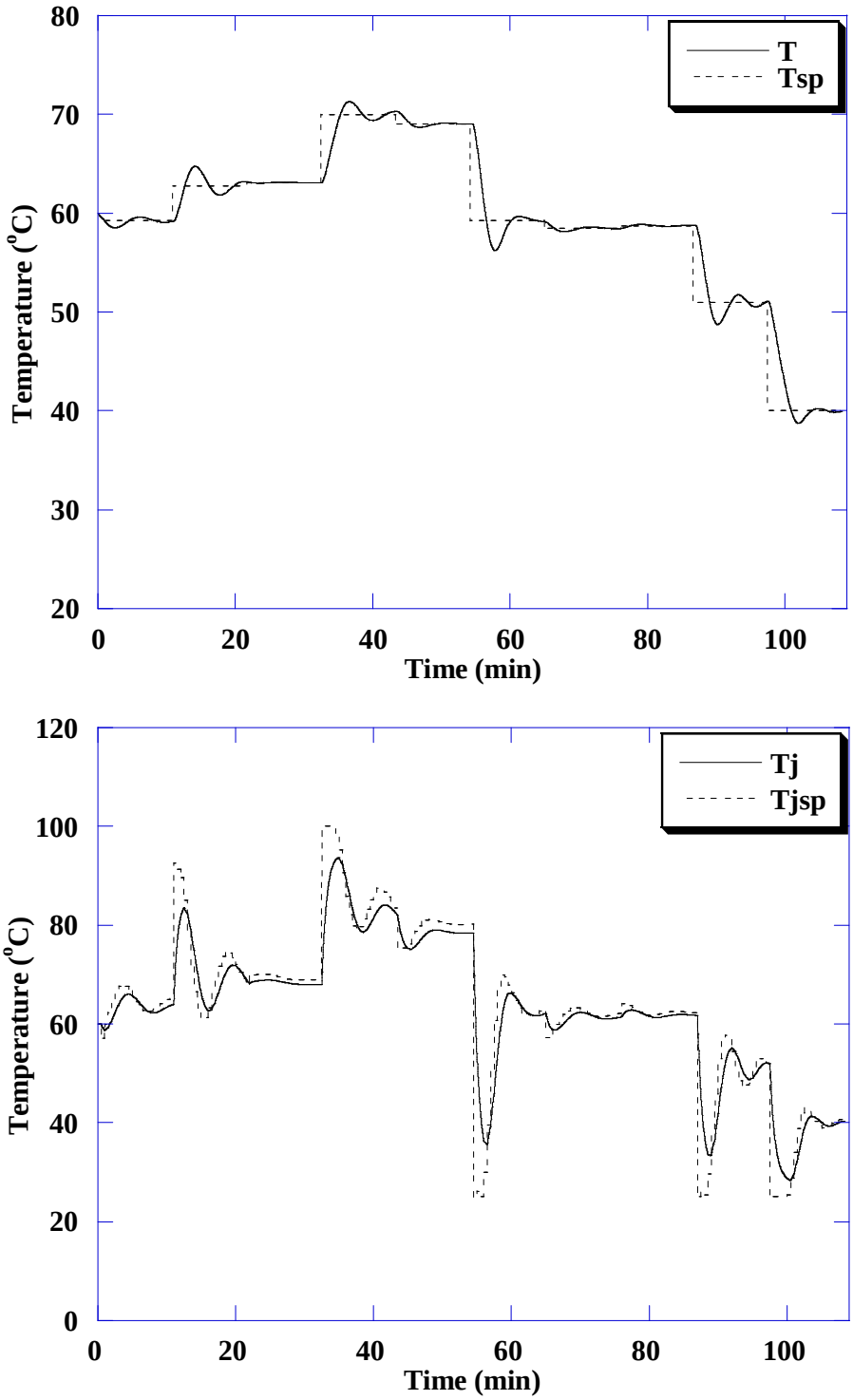


Figure 6.6 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (-30%U) using PID controller

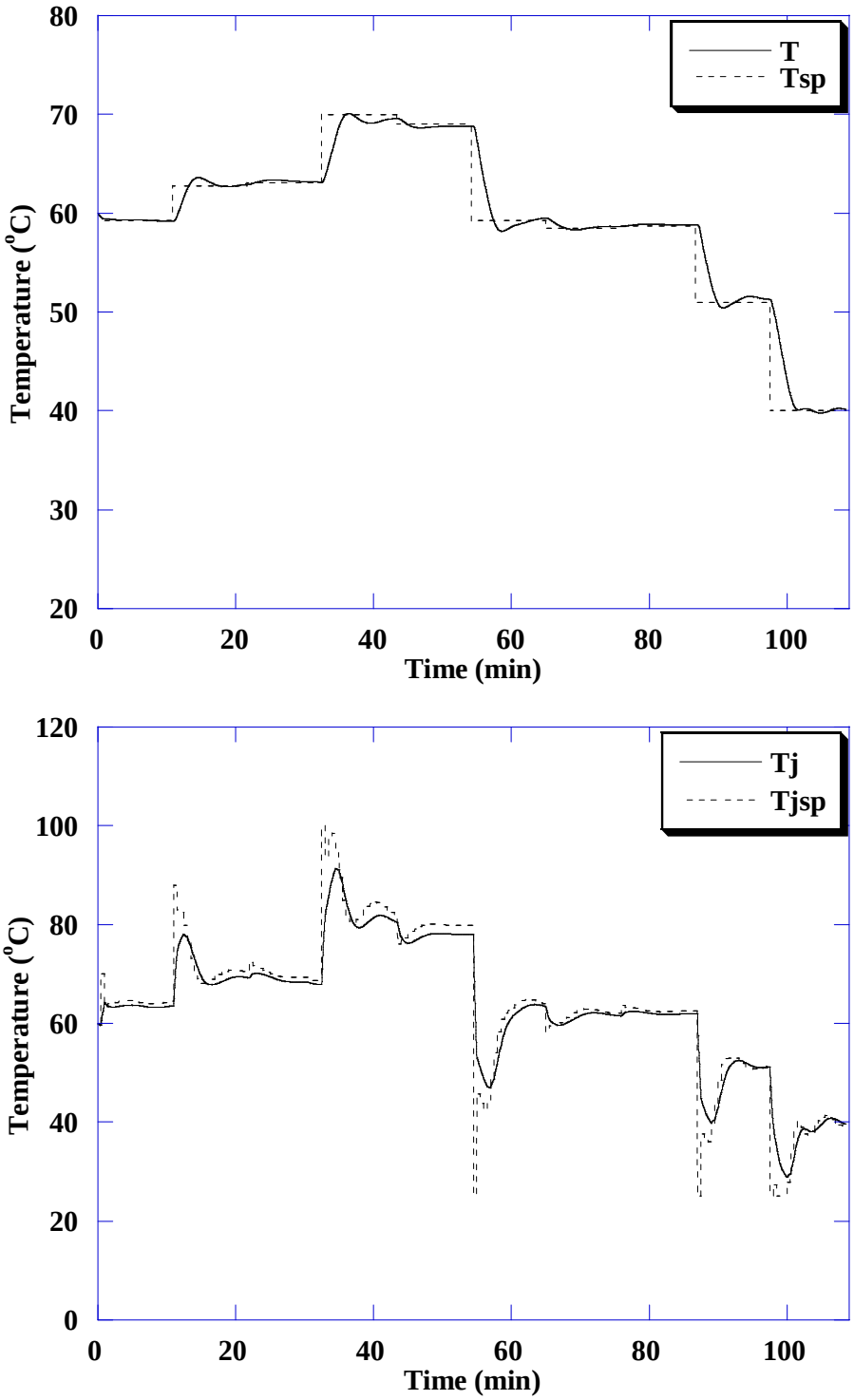


Figure 6.7 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (-30%U) using NNDIC controller

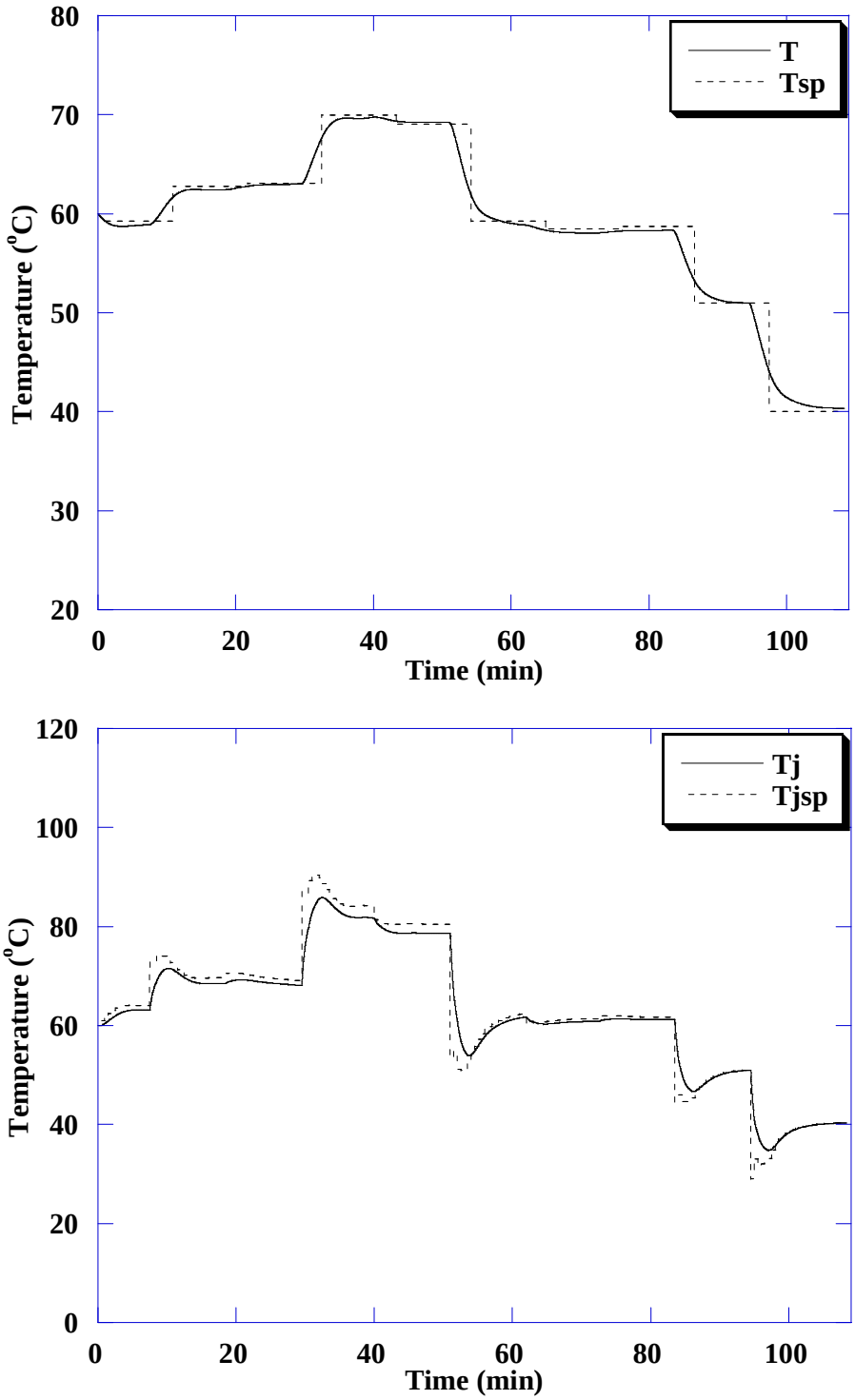


Figure 6.8 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (-30%U) using NNMPC controller

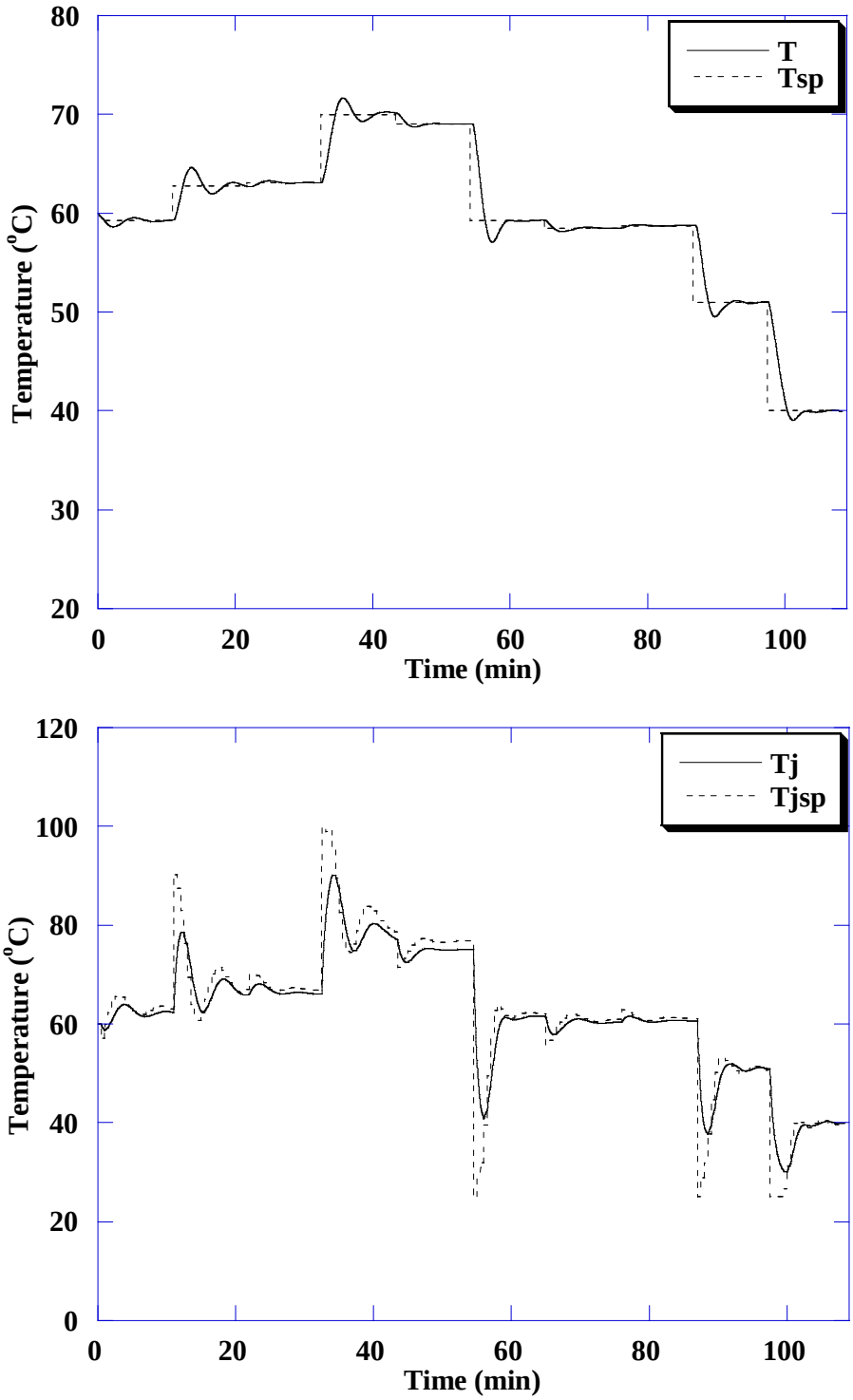


Figure 6.9 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (+30%H_{crys}) using PID controller

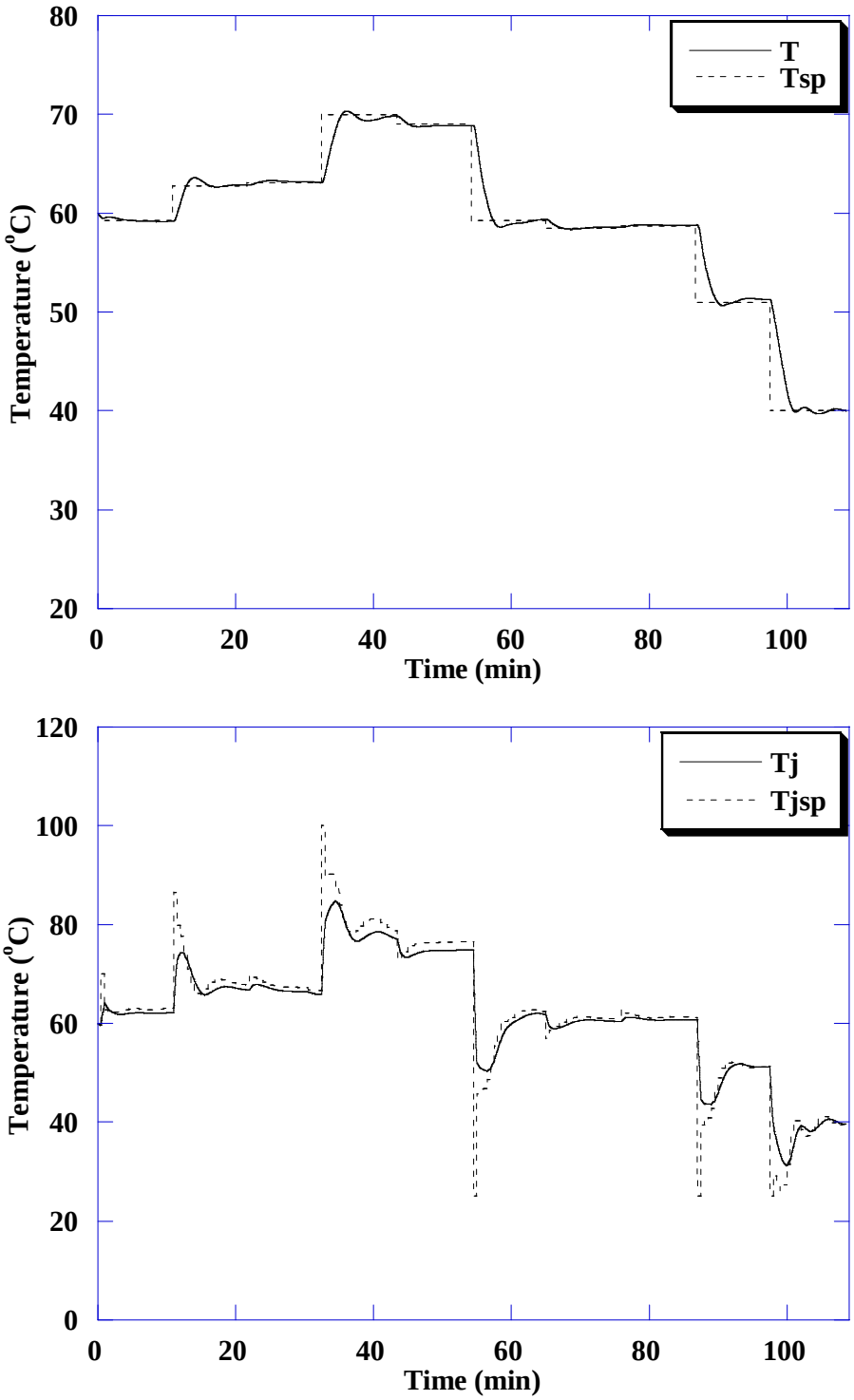


Figure 6.10 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (+30%H_{crys}) using NNDIC controller

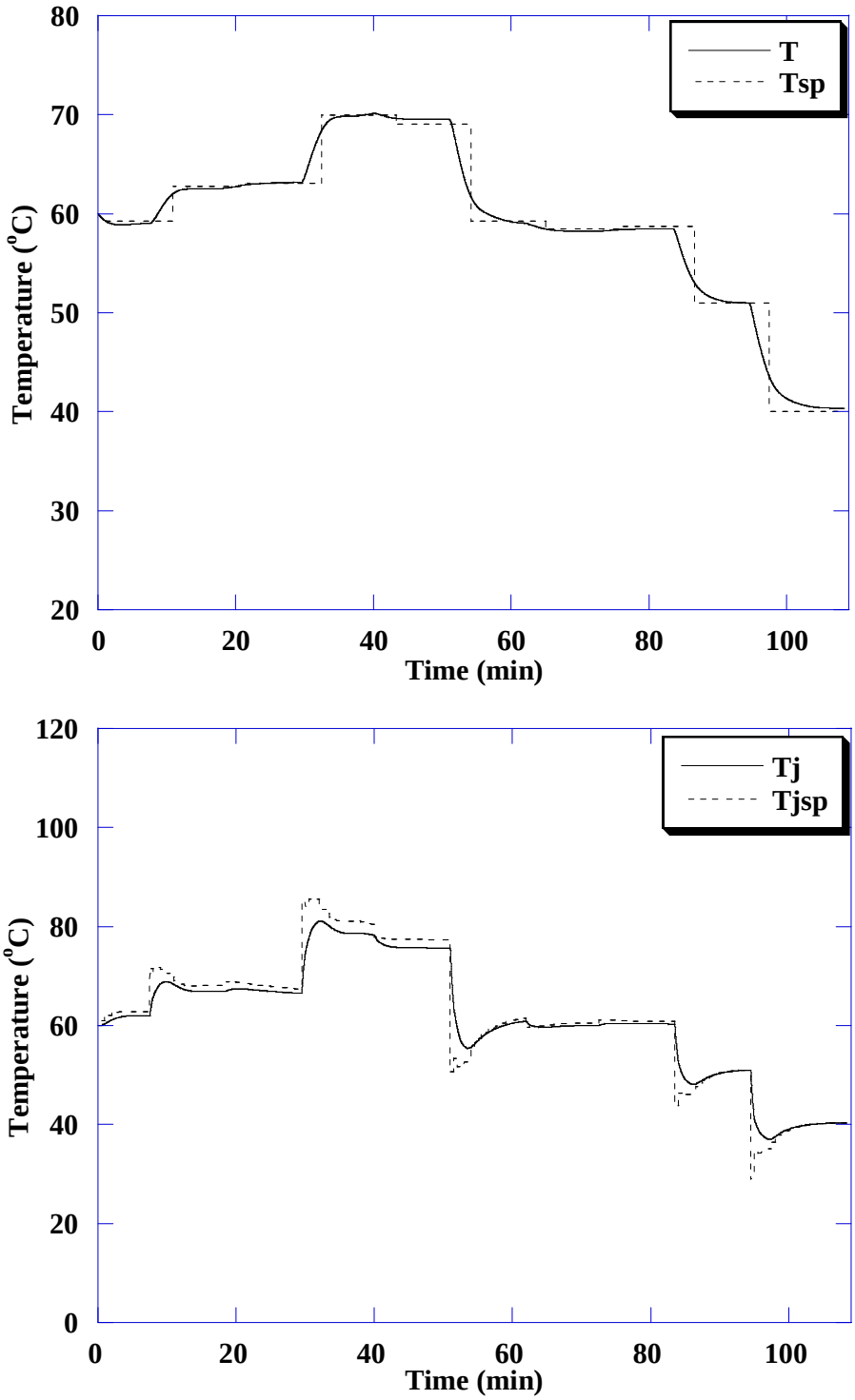


Figure 6.11 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (+30%H_{crys}) using NNMPC controller

Table 6.1 summarizes the control performance of the PID, NNDIC and NNMPC controllers in terms of the integral absolute error (IAE). It can be seen that the NNMPC provides best control performance in all parameter mismatch cases. The robustness of the NNMPC can be explained by the fact that the obtained neural network which is applied with the MPC controller is trained with the wide range of operating conditions.

6.3.2. Comparisons of the Best Controller Performance

In this section, the best performance of the PID and NNMPC controllers are compared. It consists of 2 cases which are the nominal case and the parameter mismatch case that are same as the section 6.3.1.

Nominal case

In this case, the PID and NNMPC controller are tuned as the best performance of each controller. Table 6.4 shows the best performance of the PID and NNMPC controllers. Figures 6.12-6.13 show the response profiles of the crystallizer temperature and the jacket temperature set point. It can be seen that the both controllers can bring the temperature closely to set point with overshoot and oscillation. However, the PID controller gives more overshoot and oscillation than the NNMPC controller does.

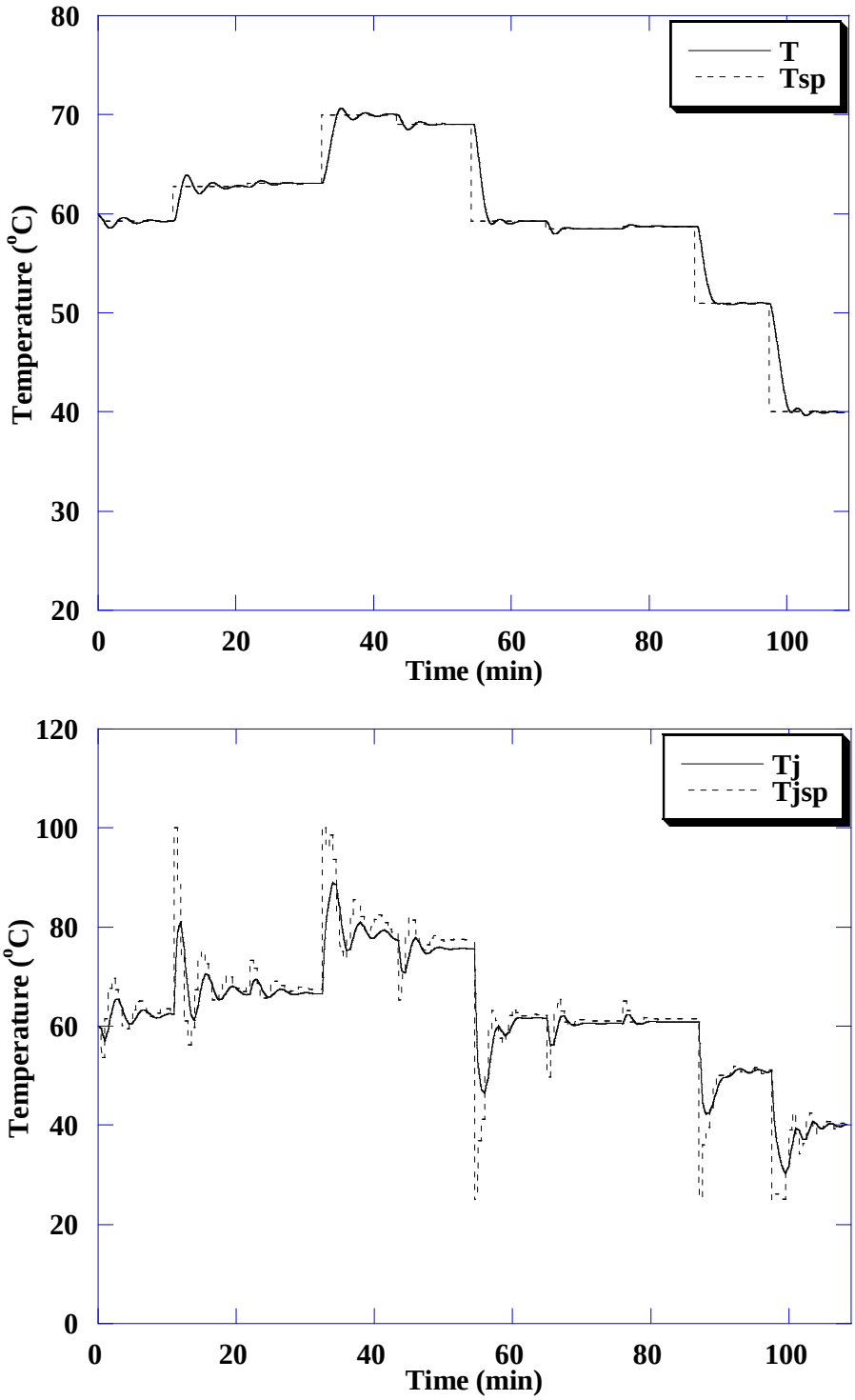


Figure 6.12 The crystallizer temperature control and the manipulated variable (T_{jsp}) in nominal case based on the best performance of PID controller

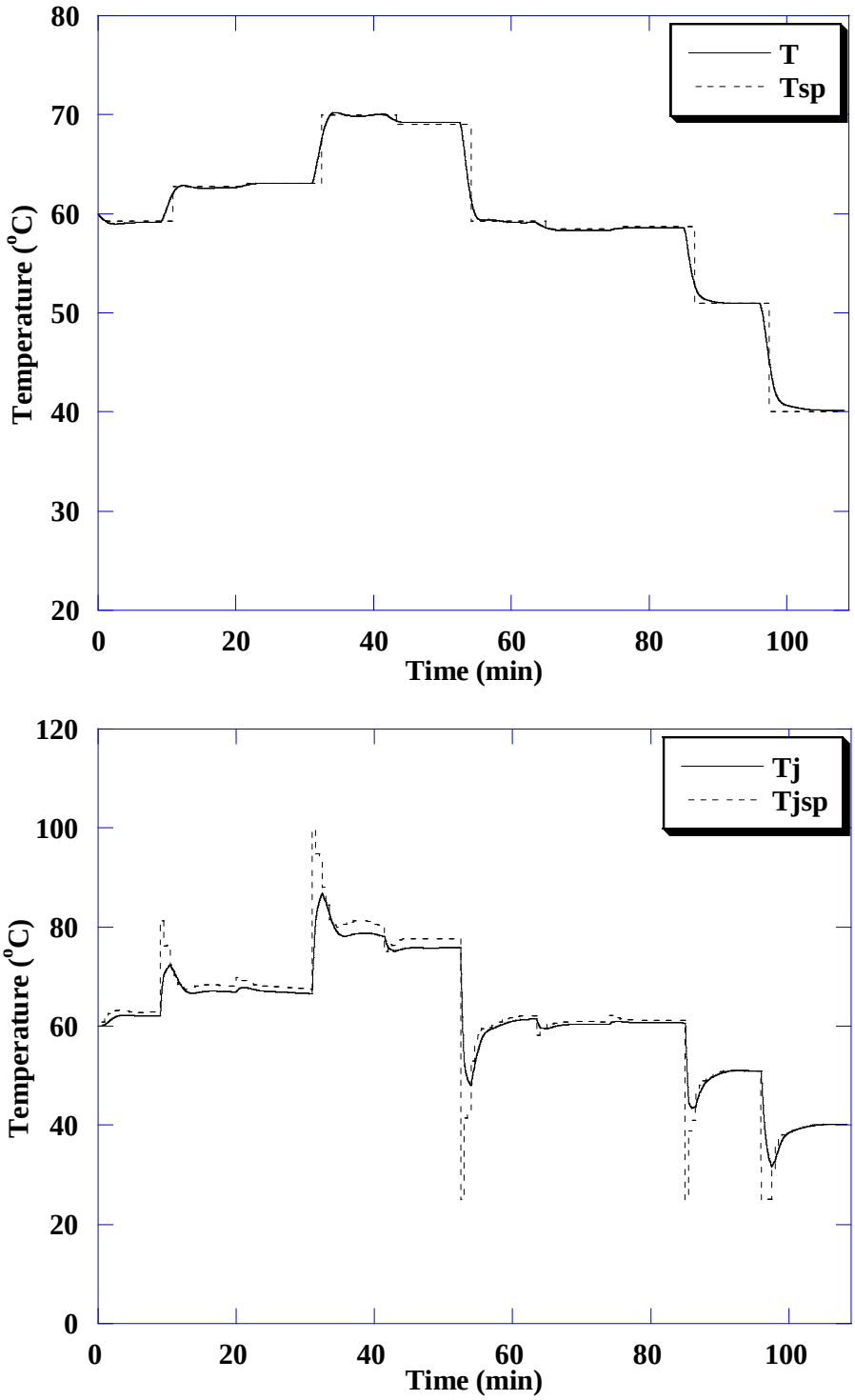


Figure 6.13 The crystallizer temperature control and the manipulated variable (T_{jsp}) in nominal case based on the best performance of NNMPC controller

Parameter mismatch case

For this case, the robustness of the controller is studied by mismatch the parameters which are same the section 6.3.1. The parameters mismatch are divided into 5 parameters consisting of decreasing 30% of U, increasing 30% of H_{crys}, decreasing 30% of H_{evap}, increasing 30% of k_p and increasing 30% of k_g from its nominal value. The parameters mismatch cases are divided 6 cases which are shown in Table 6.4. Figures 6.14-6.17 show the response profiles of crystallizer temperature and the jacket temperature set point.

Table 6.2 Performance of the best performance controller for nominal and parameters mismatch cases

Cases	IAE values		
	NNMPC	PID	NNDIC
Nominal	41.21	67.90	79.85
-30%U	44.96	81.34	93.76
+30%k_p	41.43	67.91	80.29
+30%k_g	41.40	68.00	81.12
+30%H_{crys}	42.13	68.75	80.87
-30%H_{evap},	41.49	69.00	80.01
-30%U, +30%H_{crys}, -30%H_{evap}, +30%k_g, +30%k_p	45.42	84.70	98.13

Table 6.4 summarizes the control performance of the best performance of the PID and NNMPC controllers in terms of the absolute error (IAE). It can be seen that the NNMPC controller provides superior control performance in all cases.

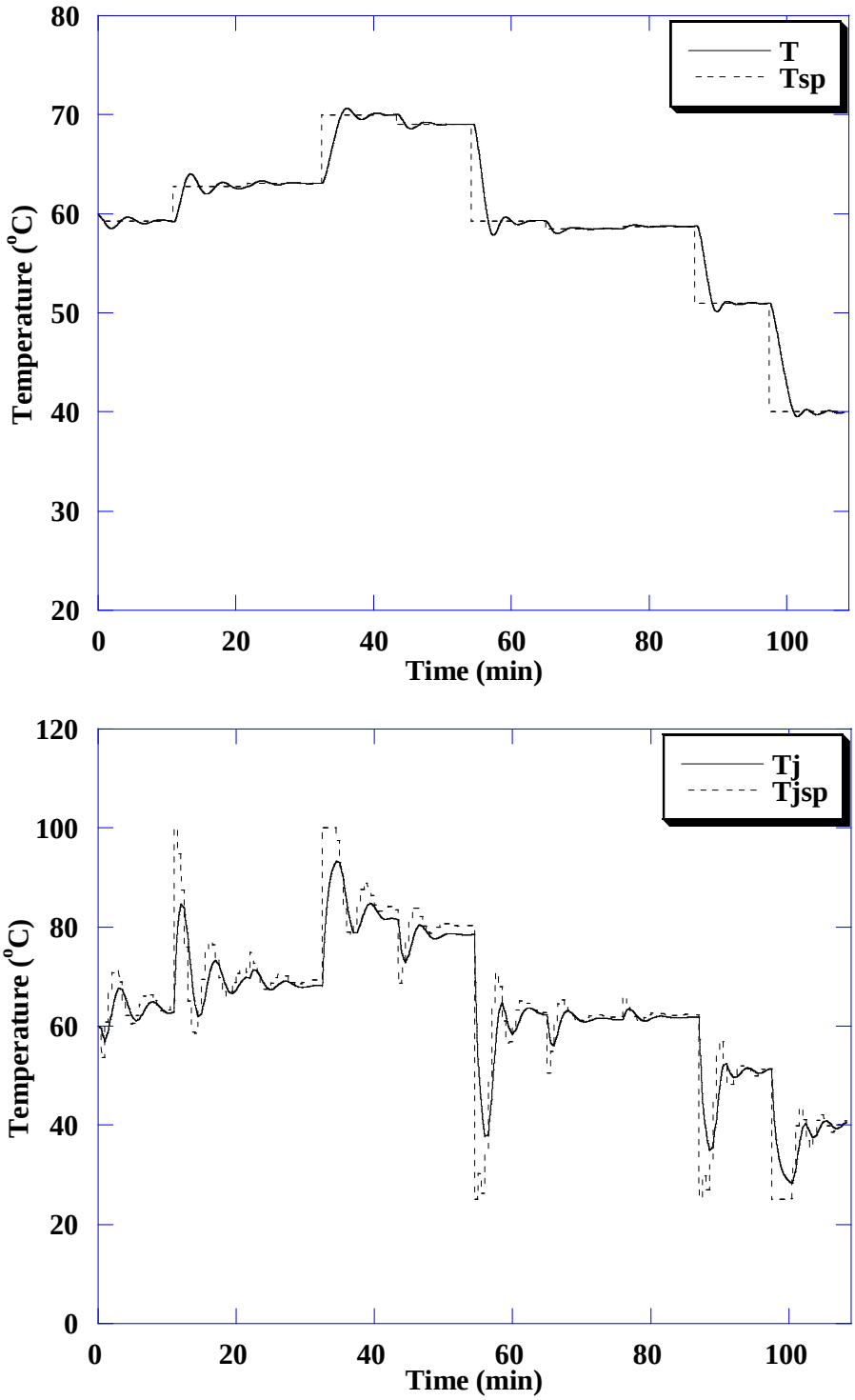


Figure 6.14 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (-30%U) based on the best performance of PID controller

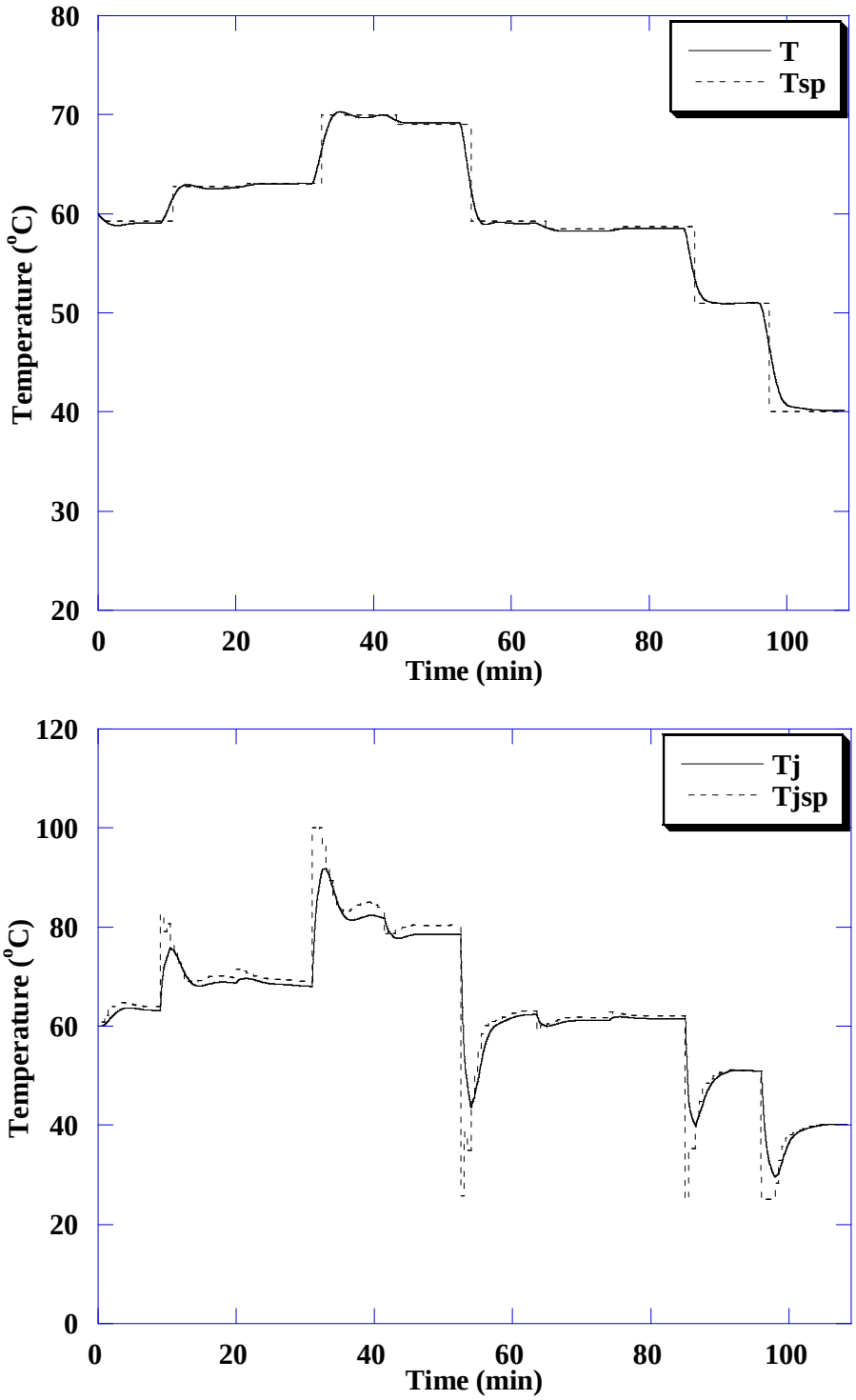


Figure 6.15 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (-30%U) based on the best performance of NNMPC controller

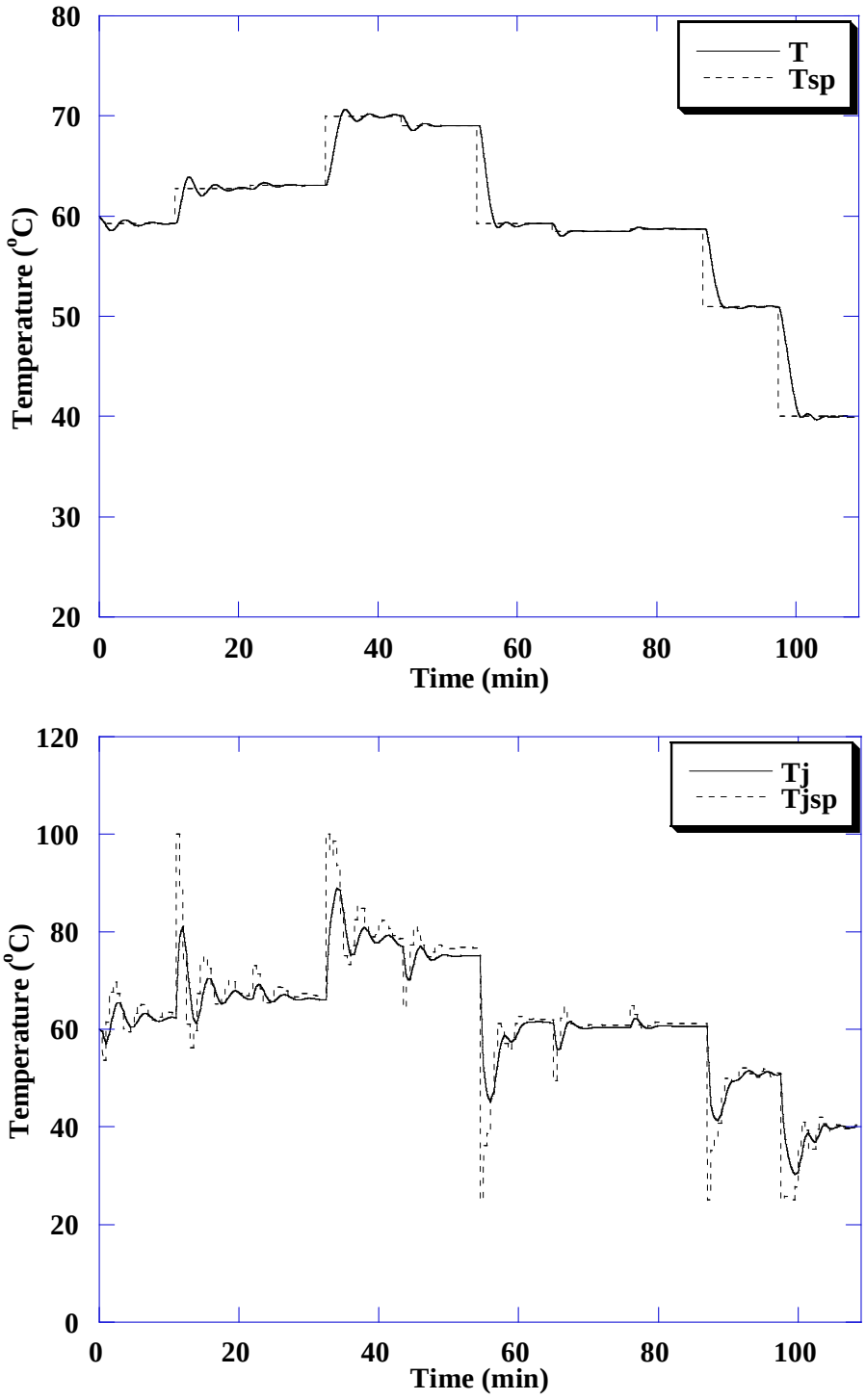


Figure 6.16 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (+30%H_{crys}) based on the best performance of PID controller

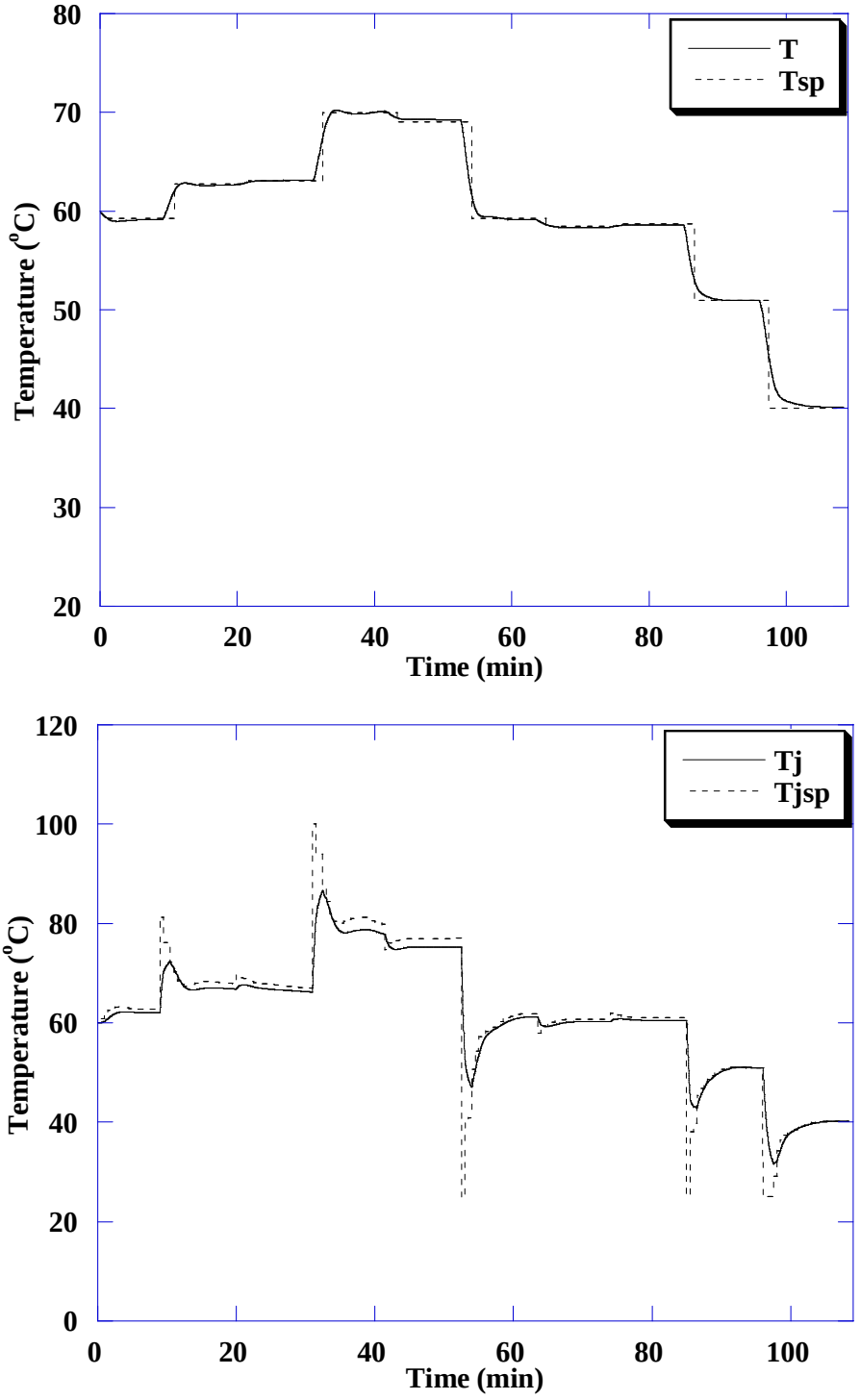


Figure 6.17 The crystallizer temperature control and the manipulated variable (T_{jsp}) in parameter mismatch case (+30%H_{crys}) based on the best performance of NNMPC controller

CHAPTER VII

CONCLUSIONS

In this study, the implementation of a dynamic optimization and neural network is applied to improve the product quality of a batch heating/cooling evaporative crystallizer for production of citric acid. Combining heating/cooling and evaporation in batch crystallization can produce higher supersaturation, higher crystal yield and larger crystal size. The dynamic optimization computes the optimal operating temperature policy to maximize the crystal yield and the crystal size. The neural network is applied to model which is used in model predictive control and to control. This neural networks use tan-sigmoid and linear as activation functions. The Levenberg-Marquardt algorithm is applied for training the network. The objective of neural network training is to minimize the error function which is mean square error (MSE) between the predicted neural network values and actual targeted value. One goal in production is the control of product properties. In this work, the PID, NNDIC and NNMPC controllers are applied to control the process.

In dynamic optimization, the optimization problems are 2 problems which are to maximize the average crystal size (OPT1) and maximize crystal yield (OPT2). The optimal product qualities in OPT1 provide average crystal size and crystal yield about 451 μm and 1.585 kg, respectively and OPT2 provide average crystal size and crystal yield about 420 μm and 1.641 kg, respectively. The heating/cooling evaporative crystallization method produces the average crystal size of 19% and 30% larger than cooling and evaporation method, respectively as well as the crystal yield obtained from heating/cooling evaporative crystallization method is higher than cooling and evaporation method about 50%.

The neural network forward model is used to predict the future value of the concentration, the crystallizer temperature and the jacket temperature profiles. The

optimal structure of neural network forward model consists of 8 nodes in input layer, 14 nodes in first hidden layer, 10 nodes in second hidden layer, and 3 nodes in output layer. The neural inverse model is used to control the crystallizer temperature by predicting the value of the manipulated variable (the jacket temperature set point, T_{jsp}).The optimal structure of neural network inverse model is composes of 11 nodes in the input layer, 12 nodes in the first hidden layer, 8 nodes in the second hidden layer and 1 node in the output layer. Both neural network forward and inverse models show good accuracy for the prediction of the system.

In the controller designing, the PID, NNDIC and NNMPC controllers are compared. In a nominal case, The PID and NNDIC controllers give some overshoot of the controlled variable (T) but the response of the control variable of the NNMPC controller is a little overshoot. The required time for the control variable to reach and remain to the set point of the NNMPC controller is less than the NNDIC and PID controllers. In term of the offset, the NNDIC controller give some offset but the NNMPC and PID controllers give a little offset. In robustness of controller, the parameter mismatch is used to test the performance of controller. It can be seen that the NNMPC controller provides superior control performance in all case.

REFERENCES

Apelblat, A., Dov, M., Wisniak, J., and Zabicky, J., The vapour pressure of water over saturated aqueous solutions of malic, tartaric, and citric acids, at temperatures from 288 K to 323 K, The Journal of Chemical Thermodynamics 27 (1995): 35-41.

Arpornwichanop, A., and Shomchoam, N., Control of fed-batch bioreactors by a hybrid on-line optimal control strategy and neural network estimator, Neurocomputing 72 (2009): 2297–2302.

Bohlin, M., and Rasmuson, A.C., Application of controlled cooling and seeding in batch crystallization, The Canadian Journal of Chemical Engineering 70 (February, 1992): 120-126.

Charoenniyom, T., Kittisupakorn, P., and Daosud, W., Neural network modeling and optimization of the methyl methacrylate production process for esterification reaction in batch reactor, Pure and Applied Chemistry International Conference, Bangkok, Thailand; 5-7 January, 2011

Choong, K.L., and Smith, R., Novel strategies for optimization of batch, semi-batch and heating/cooling evaporative crystallization, Chemical Engineering Science 59 (2004): 329–343.

Choong, K.L., and Smith, R., Optimization of batch cooling crystallization, Chemical Engineering Science 59 (2004): 313–327.

Damour, C., Benne, M., Boillereaux, L., Grondin-Perez, B., and Chabriat, J.P., NMPC of an industrial crystallization process using model-based observers, Journal of Industrial and Engineering Chemistry 16 (2010): 708–716.

Daosud, W., Thitiyasook, P., Arpornwichanop, A., Kittisupakorn, P., and Hussain, M.A., Neural network inverse model-based controller for the control of a steel pickling process, Computers and Chemical Engineering 29 (2005): 2110–2119.

Fagervik, K., Konstari, O., and Schalien, R.V., Control of batch evaporative crystallization of sugar by means of adaptive simulation, American Control Conferences; 15-17 June 1988.

Fevotte, G., Canpont, D., and Lakrori, M., Non linear control of a batch evaporative crystallization using an algorithm of "L/A" type, International conference on analysis and optimization of systems 9, Antibes, France, volume 144: (37-46); 12 June 1990.

Georgieva, P., and Azevedo, S.F. de, Application of feed forward neural networks in modeling and control of a fed-batch crystallization process NN modeling and estimator, Proceedings of world academy of science, engineering and technology 12; march 2006.

Georgieva, P., and Azevedo, S.F. de, Neural network-based control strategies applied to a fed-batch crystallization process, International Journal of Engineering and Mathematical Sciences 3:3 (2007): 224-233.

Gomi, H., and Kawato, M., Learning control for a closed loop system using feedback-error-learning, Proceeding of the 29th Conference on Decision and Control Honolulu. Hawall; December 1990.

Guez, A., Eilbert, J.L., and Kam, M., Neural Network Architecture for Control, IEEE Control Systems Magazine (1998): 22-25.

Hussain, M.A., Kittisupakorn, P., and Daosud, W., Implementation of Neural-Network-Based Inverse-Model Control Strategies on an Exothermic Reactor, ScienceAsia 27 (2001): 41-50.

Keong, K.G., Sha, W., and Malinov, S., Artificial neural network modelling of crystallization temperatures of the Ni–P based amorphous alloys, Materials Science and Engineering A365 (2004): 212–218.

Kittisupakorn, P., Model based control for batch chemical processes. Thailand: Chulalongkorn University Press, 2008

Kittisupakorn, P., Tangteerasunun, P., and Thitiyasook, P., Dynamic neural network modeling for hydrochloric acid recovery process, Korean Journal of Chemical Engineering 22(6) (2005): 813-821.

Kittisupakorn, P., Thitiyasook, P., Hussain, M.A., and Daosud, W., Neural network based model predictive control for a steel pickling process, Journal of Process Control 19 (2009): 579–590.

Konakom, K., Kittisupakorn, P., and Mujtaba, I.M., Batch control improvement by model predictive control based on multiple reduced-models, Chemical Engineering Journal 145 (2008): 129–134.

Kumar, K.V., Martins, P., and Rocha, F., Modelling of the batch sucrose crystallization kinetics using artificial neural networks: comparison with conventional regression analysis, Industrial and Engineering Chemistry Research 47 (2008): 4917-4923.

Ławryńczuk, M., Modelling and nonlinear predictive control of a yeast fermentation biochemical reactor using neural networks, Chemical Engineering Journal 145 (2008): 290–307.

Lin, X., Zhang, H., Wei, L., and Liu, H., Optimal control for industrial sucrose crystallization with action dependent heuristic dynamic programming, Proceedings of the 8th World Congress on Intelligent Control and Automation, Jinan, China; 6-9 July 2010.

Li, Y., and Häuβler, A., Artificial evolution of neural networks and its application to feedback control, Artificial intelligence in Engineering 10 (1996): 143-152

Mesbah, A., Huesman, A.E.M., Kramer, H.J.M., Nagy, Z.K., and Van den Hof, P.M.J., Real-time control of a semi-industrial fed-batch evaporative crystallizer using different direct optimization strategies, AIChE Journal Volume 57, Issue 6, (June 2011): 1557–1569.

Mesbah, A., Kalbasenka, A.N., Huesman, A.E.M., Kramer, H.J.M., and Van den Hof, P.M.J., Real-time dynamic optimization of batch crystallization processes, Proceedings of the 17th World Congress the International Federation of Automatic Control, Seoul, Korea; 6-11 July 2008.

Mujtaba, I. M., Aziz, N., and Hussain, M. A., Neural network based modeling and control in batch reactor, Chemical Engineering Research and Design, 84(A8) (2006): 635–644.

Mukhopadhyay, S.C., and Epsteln, M.A.F., Computer model for crystal size distribution control in a semi-batch evaporative crystallizer, Industrial and Engineering Chemistry Process Design and Development 19 (1980): 358-364.

Nakanishi, J., and Schaal, S., Feedback error learning and nonlinear adaptive control, Neural Networks 17 (2004): 1453–1465.

Neelakantan, P.S., and Mukesh, D., Computer modeling of a continuous evaporative crystallizer, Industrial and Engineering Chemistry Process Design and Development 18 (1979): 56-59.

Nueaklong, E., Kittisupakorn, P., and Daosud, W., Dynamic and control of heat exchanger system in hard chrome electroplating using neural network, The 7th International Conference on Computing and Information Technology, Bangkok, Thailand; 11-12 May 2011.

Othman, F., and Naseri, M., Reservoir inflow forecasting using artificial neural network, International Journal of The Physical Sciences 6(3) (2011): 434-440

Paengjuntuek, W., Kittisupakorn, P., and Arpornwichanop, A., On-line dynamic optimization integrated with generic model control of a batch crystallizer, Journal of Industrial and Engineering Chemistry 14 (2008): 442–448.

Paengjuntuek, W., Kittisupakorn, P., and Arpornwichanop, A., Optimization and nonlinear control of a batch crystallization process, Journal of the Chinese Institute of Chemical Engineers 39 (2008): 249–256.

Seborg, Dale E., Edgar, Thomas F. and Mellichamp, Duncan A., Process dynamics and control, 2nd edition. United States of America: John Wiley & Sons, 2004.

Sowul, L., and Epsteln, M.A.F., Crystallization kinetics of sucrose in a CMSMPR evaporative crystallizer, Industrial and Engineering Chemistry Process Design and Development 20 (1981): 197-203.

Tung, H., Pual, E., Midler, M., and McCauley, J.A., Crystallization of organic compounds an industrial perspective. New Jersey: John Wiley & Sons, Inc., 2009.

Wang, L.L., Wallace, T.C., Fredrick, SR., Hampel, G., and Steele, J.H., Vacuum evaporation of KCI-NaCI salts: part II. Vaporization-rate model and experimental results, Metallurgical and Materials Transactions B volume 27B (June 1996): 433-443.

Wong, S.Y., Bund, R.K., Connelly, R.K., and Hartel, R.W., Modeling the crystallization kinetic rates of lactose via artificial neural network, Crystal Growth and Design 10(6) (2010): 2620-2628.

Yan, Z., Xiuxia, L., Peng, Y., Zengqiang, C., and Zhuzhi, Y., Modeling and control of nonlinear discrete-time systems based on compound neural networks, Chinese Journal of Chemical Engineering 17(3) (2009): 454-459.

Yang, M., and Wei, H., Application of a neural network for the prediction of crystallization kinetics, Industrial and Engineering Chemistry Research 45 (2006): 70-75.

Yu, D.L., and Gomm, J.B., Implementation of neural network predictive control to a multivariable chemical reactor, Control Engineering Practice 11 (2003): 1315–1323.

Zilouchian, A. and Jamshidi, M., Intelligent control systems using soft computing methodologies. United States of America: CRC Press LLC, 2001.

APPENDICES

APPENDIX A

Proportional-Integral-Derivative Control

A basic closed-loop control system is shown in Figure A.1

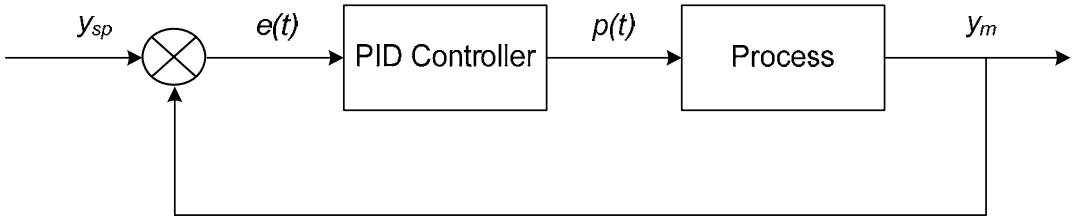


Figure A.1 Block diagram of closed loop control with PID controller

$$e(t) = y_{sp} - y_m$$

The combination of the proportional, integral and derivative control modes as a PID controller is given by (Seborg et al., 2004)

$$p(t) = \bar{p} + K_c\left(e(t) + \frac{1}{\tau_I}\int_0^t e(t)dt + \tau_D \frac{de(t)}{dt} \right)$$

$e(t)$ = error signal

y_{sp} = set point

y_m = measured value of the controlled variable

$p(t)$ = controller output

$\bar{p}$ = bias value

K_c = controller gain

τ_I = integral time or reset time

τ_D = derivative time

APPENDIX B

Tuning Relations Based on Integral Error Criteria

Controller tuning relations have been developed that optimize the closed-loop response for a simple process model and a specified disturbance or set point change. The optimum settings minimize an integral error criterion. Three popular integral error criteria are: (Seborg et al., 2004)

1. Integral of the absolute value of the error (IAE)

$$IAE = \int_0^\infty |e(t)| dt$$

2. Integral of the squared error (ISE)

$$ISE = \int_0^\infty (e(t))^2 dt$$

3. Integral of the time-weighted absolute error (ITAE)

$$ITAE = \int_0^\infty t|e(t)| dt$$

The ISE criterion penalizes large errors, while the ITAE criterion penalizes errors that persist for long periods of time. In general, the ITAE is the preferred criterion because it usually results in the most conservative controller settings. By contrast, the ISE criterion provides the most aggressive settings, while the IAE criterion tends to produce controller settings that are between those for the ITAE and ISE criteria.

Figure B.1 illstrates the characteristics of the step response of a second-order underdamped process. The following terms are used to describe the dynamics of underdamped processes:

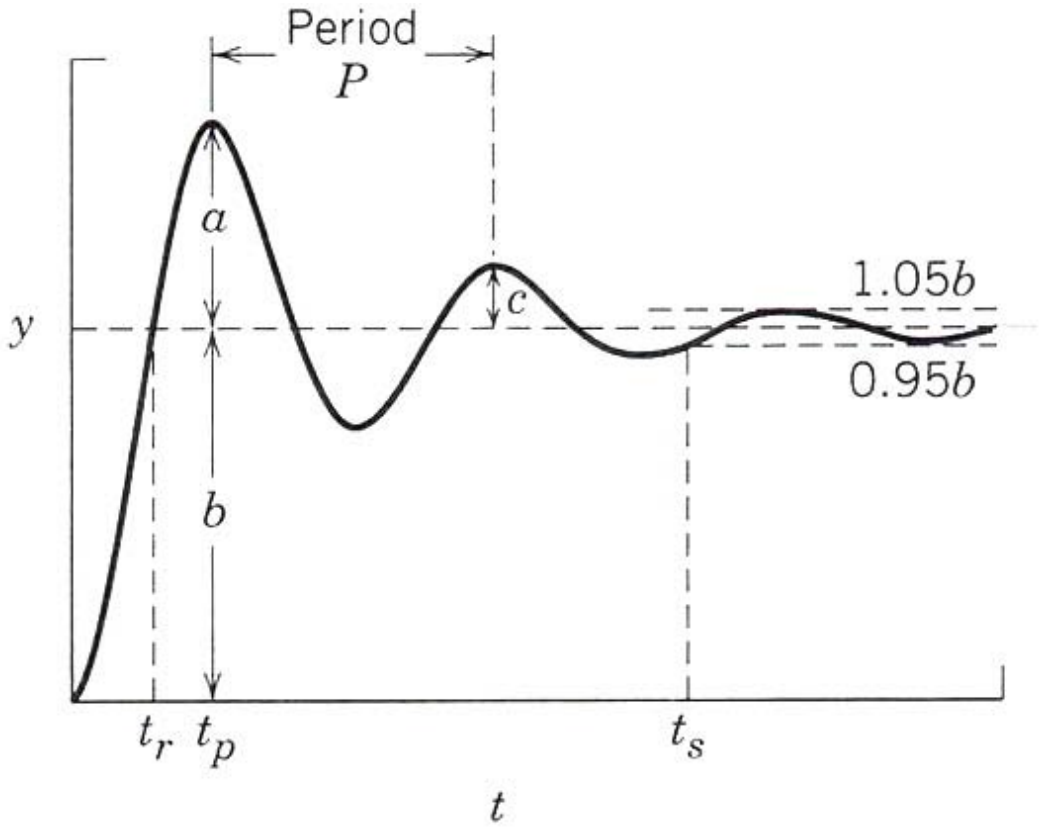


Figure B.1 Performance characteristics for the step response

1. Rise Time. tr is the time the process output takes to first reach the new steady-state value.
2. Time to First Peak. t_p is the time required for the output to reach its first maximum value.
3. Settling Time. ts is the time required for the process output to reach and remain inside a band.
4. Overshoot. OS = a/b
5. Decay Ratio. DR = c/a
6. Period of Oscillation. P is the time between two successive peaks or two successive valleys of the response.

APPENDIX C

Activation Function

The popular functions which are linear, sigmoid and tan-sigmoid are shown as follow.

Linear transfer function

The linear transfer function is utilized in the output layer for output expansion purpose. The result that receives from this transfer function is a linear as show in Figure C.1.

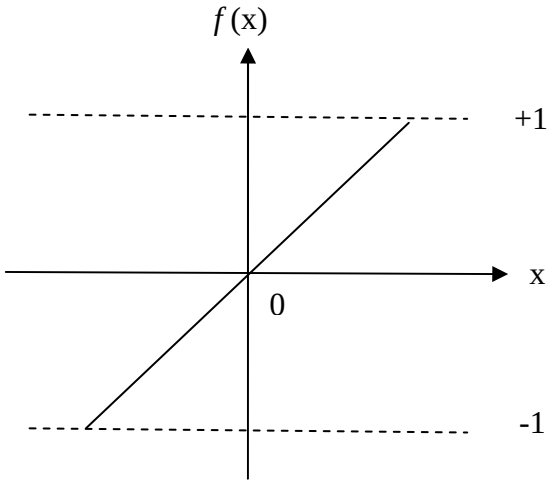


Figure C.1 Linear transfer function

Log-Sigmoid transfer function

The Log-Sigmoid transfer function is a subset of nonlinear transfer functions. This transfer function will convert high positive value into 1 and converted high negative value into 0 that show in Figure C.2.

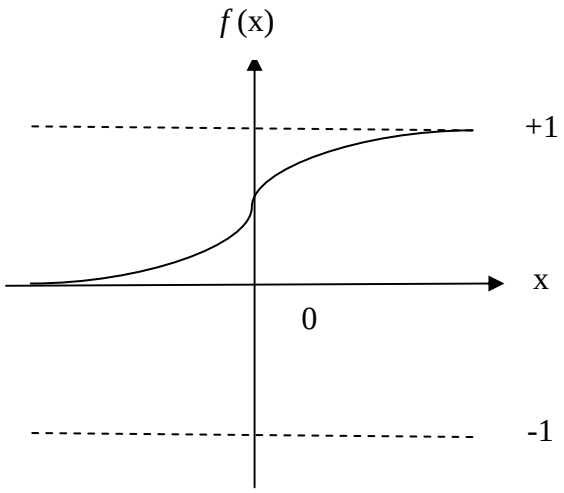


Figure C.2 Log-Sigmoid transfer function

Tan-Sigmoid transfer function

One of nonlinear transfer functions is Tan-Sigmoid transfer function. The transfer function will transform high positive value into 1 and converted high negative value into -1. The figure is shown below:

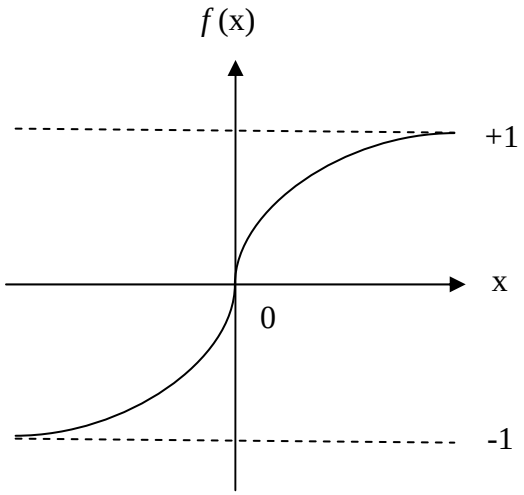


Figure C.3 Tan-Sigmoid transfer function

APPENDIX D

Mean Square Error of Neural Network Modeling

Table D.1 Mean squared error value of the neural network forward model: 1 layer

Number of Nodes in Hidden Layer	Mean square error (MSE)		
	Testing 1	Testing 2	Validating
2	8.8217×10^{-4}	9.7749×10^{-4}	9.9292×10^{-4}
3	6.0828×10^{-5}	9.1509×10^{-5}	1.4244×10^{-4}
4	5.1708×10^{-5}	7.9370×10^{-5}	1.3763×10^{-4}
5	6.9844×10^{-5}	6.9844×10^{-5}	1.3739×10^{-4}
6	1.5297×10^{-4}	2.7864×10^{-4}	1.3451×10^{-4}
7	4.7231×10^{-5}	8.0284×10^{-5}	1.3294×10^{-4}
8	5.9365×10^{-5}	8.8135×10^{-5}	1.3491×10^{-4}
9	1.0115×10^{-4}	1.1341×10^{-4}	1.3468×10^{-4}
10	6.4585×10^{-5}	9.0959×10^{-5}	1.5158×10^{-4}
11	7.6584×10^{-5}	1.1204×10^{-4}	1.2808×10^{-4}
12	6.3076×10^{-5}	1.2005×10^{-4}	1.3302×10^{-4}
13	9.8875×10^{-5}	1.3593×10^{-4}	1.3347×10^{-4}
14	7.6312×10^{-5}	1.0599×10^{-4}	1.3174×10^{-4}
15	9.6580×10^{-5}	1.4022×10^{-4}	1.5279×10^{-4}
16	9.2859×10^{-5}	1.1242×10^{-4}	1.3324×10^{-4}
17	8.1617×10^{-5}	1.0450×10^{-4}	1.3744×10^{-4}
18	6.8858×10^{-5}	9.6370×10^{-5}	1.3644×10^{-4}
19	8.5856×10^{-5}	9.5175×10^{-5}	1.4073×10^{-4}
20	7.9227×10^{-5}	1.4413×10^{-4}	1.4577×10^{-4}
21	5.7180×10^{-5}	8.9381×10^{-5}	1.2848×10^{-4}
22	7.0760×10^{-5}	1.7023×10^{-4}	1.4047×10^{-4}
23	9.2311×10^{-5}	1.0066×10^{-4}	1.3386×10^{-4}
24	1.0974×10^{-4}	1.4240×10^{-4}	1.4870×10^{-4}
25	7.2929×10^{-5}	1.2494×10^{-4}	1.5753×10^{-4}
26	1.1704×10^{-4}	1.7373×10^{-4}	1.3900×10^{-4}
27	8.6239×10^{-5}	1.2038×10^{-4}	1.4418×10^{-4}
28	6.7093×10^{-5}	1.0353×10^{-4}	1.3970×10^{-4}
29	1.3351×10^{-4}	2.2671×10^{-4}	1.6368×10^{-4}
30	8.1444×10^{-5}	1.0393×10^{-4}	1.3094×10^{-4}

Table D.2 Mean squared error value of the neural network forward model: 2 layers

Nodes in 1st Hidden Layer	Nodes in 2nd Hidden Layer	Mean square error (MSE)		
		Testing 1	Testing 2	Validating
2	2	8.8513×10^{-4}	9.7777×10^{-4}	9.0362×10^{-4}
2	4	4.0400×10^{-4}	4.3825×10^{-4}	4.2222×10^{-4}
2	6	3.4685×10^{-4}	3.8635×10^{-4}	3.3236×10^{-4}
2	8	3.6083×10^{-4}	3.6083×10^{-4}	3.4685×10^{-4}
2	10	3.5483×10^{-4}	4.0414×10^{-4}	3.7156×10^{-4}
2	12	3.5405×10^{-4}	4.0845×10^{-4}	3.7979×10^{-4}
2	14	3.6042×10^{-4}	4.0001×10^{-4}	3.4024×10^{-4}
2	16	4.0381×10^{-4}	4.3459×10^{-4}	4.0766×10^{-4}
2	18	3.5154×10^{-4}	3.7802×10^{-4}	3.4029×10^{-4}
2	20	4.0613×10^{-4}	3.8745×10^{-4}	3.7552×10^{-4}
4	2	9.1417×10^{-4}	9.9822×10^{-4}	9.0947×10^{-4}
4	4	7.4624×10^{-5}	6.7500×10^{-5}	5.0375×10^{-5}
4	6	5.3209×10^{-5}	7.6657×10^{-5}	4.7539×10^{-5}
4	8	6.9205×10^{-5}	1.0008×10^{-4}	4.3496×10^{-5}
4	10	6.0294×10^{-5}	9.5339×10^{-5}	4.4318×10^{-5}
4	12	7.0037×10^{-5}	9.0139×10^{-5}	4.1440×10^{-5}
4	14	6.3131×10^{-5}	8.6716×10^{-5}	4.5455×10^{-5}
4	16	6.5444×10^{-5}	8.4459×10^{-5}	4.0871×10^{-5}
4	18	5.3810×10^{-5}	8.6184×10^{-5}	4.0476×10^{-5}
4	20	4.7529×10^{-5}	1.1544×10^{-4}	4.3054×10^{-5}
6	2	9.2238×10^{-4}	1.0123×10^{-3}	8.9907×10^{-4}
6	4	5.6096×10^{-5}	9.0890×10^{-5}	4.2409×10^{-5}
6	6	7.2272×10^{-5}	8.2234×10^{-5}	4.7499×10^{-5}
6	8	8.0994×10^{-5}	1.0693×10^{-4}	4.1457×10^{-5}
6	10	8.8883×10^{-5}	1.0912×10^{-4}	4.4070×10^{-5}
6	12	7.0065×10^{-5}	9.3451×10^{-5}	5.1476×10^{-5}
6	14	6.2777×10^{-5}	1.0129×10^{-4}	4.2674×10^{-5}
6	16	6.6279×10^{-5}	1.0883×10^{-4}	3.4661×10^{-5}
6	18	6.8695×10^{-5}	1.3045×10^{-4}	4.2778×10^{-5}
6	20	7.3656×10^{-5}	8.8758×10^{-5}	3.7497×10^{-5}
8	2	8.8214×10^{-4}	9.7540×10^{-4}	8.9833×10^{-4}
8	4	5.9948×10^{-5}	8.0729×10^{-5}	4.2595×10^{-5}
8	6	7.0770×10^{-5}	9.6855×10^{-5}	4.9445×10^{-5}
8	8	6.9597×10^{-5}	8.8072×10^{-5}	3.5606×10^{-5}
8	10	7.5452×10^{-5}	1.0453×10^{-4}	4.2979×10^{-5}
8	12	7.1768×10^{-5}	9.4698×10^{-5}	3.8778×10^{-5}
8	14	6.6969×10^{-5}	7.2741×10^{-5}	4.7018×10^{-5}
8	16	7.1409×10^{-5}	8.6553×10^{-5}	4.7487×10^{-5}
8	18	6.2238×10^{-5}	7.5661×10^{-5}	4.8296×10^{-5}
8	20	5.9893×10^{-5}	8.8192×10^{-5}	3.5771×10^{-5}
10	2	8.9202×10^{-4}	9.8112×10^{-4}	8.9630×10^{-4}

10	4	7.0370×10^{-5}	9.1563×10^{-5}	4.2839×10^{-5}
10	6	6.9274×10^{-5}	8.4068×10^{-5}	3.4579×10^{-5}
10	8	6.0260×10^{-5}	1.0029×10^{-4}	4.5900×10^{-5}
10	10	7.0698×10^{-5}	1.0307×10^{-4}	4.3927×10^{-5}
10	12	8.2606×10^{-5}	1.0876×10^{-4}	4.9808×10^{-5}
10	14	7.3310×10^{-5}	1.0154×10^{-4}	4.5668×10^{-5}
10	16	7.9318×10^{-5}	1.0928×10^{-4}	5.3045×10^{-5}
10	18	6.8006×10^{-5}	1.0383×10^{-4}	4.2319×10^{-5}
10	20	6.7937×10^{-5}	8.9056×10^{-5}	4.4358×10^{-5}
12	2	8.9713×10^{-4}	9.8861×10^{-4}	9.0247×10^{-4}
12	4	7.1713×10^{-5}	1.0219×10^{-4}	3.9757×10^{-5}
12	6	7.0710×10^{-5}	9.0062×10^{-5}	4.1332×10^{-5}
12	8	6.9432×10^{-5}	1.1090×10^{-4}	4.4064×10^{-5}
12	10	8.1102×10^{-5}	1.0821×10^{-4}	4.0166×10^{-5}
12	12	7.7474×10^{-5}	8.5441×10^{-5}	5.1478×10^{-5}
12	14	6.0164×10^{-5}	9.9888×10^{-5}	3.3470×10^{-5}
12	16	8.7309×10^{-5}	1.1029×10^{-4}	3.8526×10^{-5}
12	18	7.0153×10^{-5}	9.2921×10^{-5}	3.2759×10^{-5}
12	20	6.3022×10^{-5}	8.5883×10^{-5}	3.9336×10^{-5}
14	2	9.0361×10^{-4}	9.7834×10^{-4}	9.0068×10^{-4}
14	4	7.0976×10^{-5}	1.1512×10^{-4}	4.0620×10^{-5}
14	6	6.2270×10^{-5}	9.8527×10^{-5}	4.5359×10^{-5}
14	8	7.7009×10^{-5}	8.8414×10^{-5}	4.7957×10^{-5}
14	10	5.3528×10^{-5}	8.0095×10^{-5}	4.2852×10^{-5}
14	12	9.1510×10^{-5}	9.9554×10^{-5}	4.8431×10^{-5}
14	14	7.1328×10^{-5}	7.6972×10^{-5}	4.0096×10^{-5}
14	16	7.1333×10^{-5}	1.0117×10^{-4}	3.5078×10^{-5}
14	18	9.3583×10^{-5}	1.0403×10^{-4}	4.1909×10^{-5}
14	20	6.3798×10^{-5}	1.2995×10^{-4}	6.2331×10^{-5}
16	2	8.9254×10^{-4}	1.0723×10^{-3}	9.0572×10^{-4}
16	4	8.9830×10^{-5}	1.0333×10^{-4}	5.0201×10^{-5}
16	6	7.2924×10^{-5}	1.2010×10^{-4}	5.4919×10^{-5}
16	8	1.0226×10^{-4}	9.4781×10^{-5}	5.5904×10^{-5}
16	10	6.3504×10^{-5}	1.3213×10^{-4}	2.5343×10^{-5}
16	12	7.3632×10^{-5}	8.0208×10^{-5}	5.3674×10^{-5}
16	14	6.3055×10^{-5}	9.2006×10^{-5}	4.2061×10^{-5}
16	16	8.8626×10^{-5}	1.0649×10^{-4}	4.1502×10^{-5}
16	18	7.8248×10^{-5}	9.7580×10^{-5}	5.1050×10^{-5}
16	20	8.4419×10^{-5}	9.4878×10^{-5}	4.1260×10^{-5}
18	2	9.2523×10^{-4}	1.016×10^{-4}	9.0409×10^{-4}
18	4	7.2129×10^{-5}	8.9423×10^{-5}	4.2998×10^{-5}
18	6	6.1599×10^{-5}	1.0050×10^{-4}	1.9665×10^{-5}
18	8	5.0633×10^{-5}	1.2682×10^{-4}	3.4712×10^{-5}
18	10	5.7775×10^{-5}	8.5784×10^{-5}	4.7833×10^{-5}
18	12	7.1026×10^{-5}	1.0856×10^{-4}	3.9511×10^{-5}
18	14	7.1900×10^{-5}	1.0833×10^{-4}	4.9672×10^{-5}

18	16	9.3586×10^{-5}	9.8010×10^{-5}	4.6477×10^{-5}
18	18	8.1416×10^{-5}	1.0742×10^{-4}	4.7616×10^{-5}
18	20	9.0775×10^{-5}	9.8666×10^{-5}	4.7085×10^{-5}
20	2	9.1713×10^{-4}	1.0145×10^{-3}	8.9864×10^{-4}
20	4	9.3584×10^{-5}	1.0579×10^{-4}	4.4224×10^{-5}
20	6	5.7440×10^{-5}	1.1677×10^{-4}	5.1896×10^{-5}
20	8	7.9857×10^{-5}	1.0126×10^{-4}	4.9341×10^{-5}
20	10	7.8079×10^{-5}	9.4832×10^{-5}	4.7716×10^{-5}
20	12	6.7162×10^{-5}	9.6116×10^{-5}	4.1185×10^{-5}
20	14	7.8080×10^{-5}	1.1097×10^{-4}	4.5686×10^{-5}
20	16	9.1510×10^{-5}	9.9554×10^{-5}	4.8431×10^{-5}
20	18	9.8837×10^{-5}	8.9693×10^{-5}	4.5404×10^{-5}
20	20	7.5558×10^{-5}	1.0906×10^{-4}	4.4632×10^{-5}

Table D.3 Mean squared error value of the neural network inverse model: 1 layer

Number of Nodes in Hidden Layer	Mean square error (MSE)		
	Testing 1	Testing 2	Validating
2	6.7892×10^{-5}	6.7299×10^{-5}	9.7812×10^{-5}
3	6.3053×10^{-5}	7.7620×10^{-5}	8.6339×10^{-5}
4	4.1435×10^{-5}	4.2175×10^{-5}	4.3889×10^{-5}
5	2.9266×10^{-5}	2.9439×10^{-5}	4.8736×10^{-5}
6	2.9079×10^{-5}	2.4461×10^{-5}	2.8435×10^{-5}
7	2.8766×10^{-5}	2.6343×10^{-5}	3.4507×10^{-5}
8	2.9651×10^{-5}	4.2056×10^{-5}	2.8272×10^{-5}
9	2.8616×10^{-5}	3.6158×10^{-5}	5.9800×10^{-5}
10	2.3356×10^{-5}	2.8835×10^{-5}	3.1025×10^{-5}
11	2.9644×10^{-5}	3.9956×10^{-5}	4.2961×10^{-5}
12	2.5048×10^{-5}	2.7824×10^{-5}	2.2636×10^{-5}
13	2.5265×10^{-5}	3.7118×10^{-5}	3.1295×10^{-5}
14	2.7533×10^{-5}	3.0078×10^{-5}	2.7071×10^{-5}
15	2.5853×10^{-5}	3.9235×10^{-5}	3.7244×10^{-5}
16	2.1289×10^{-5}	4.0106×10^{-5}	3.8934×10^{-5}
17	2.2432×10^{-5}	3.6299×10^{-5}	2.1381×10^{-5}
18	2.5028×10^{-5}	3.6934×10^{-5}	2.9245×10^{-5}
19	2.8252×10^{-5}	2.1007×10^{-5}	4.0422×10^{-5}
20	3.6849×10^{-5}	3.5930×10^{-5}	4.1398×10^{-5}
21	2.3776×10^{-5}	4.4725×10^{-5}	2.6453×10^{-5}
22	4.4342×10^{-5}	3.7888×10^{-5}	2.7078×10^{-5}
23	3.6222×10^{-5}	4.7238×10^{-5}	4.2800×10^{-5}
24	8.4162×10^{-5}	2.7202×10^{-5}	2.8201×10^{-5}
25	4.3825×10^{-5}	2.8991×10^{-5}	4.0402×10^{-5}
26	3.7533×10^{-5}	4.5752×10^{-5}	3.8054×10^{-5}
27	3.1711×10^{-5}	7.0382×10^{-5}	6.2546×10^{-5}
28	4.1585×10^{-5}	2.0565×10^{-5}	3.0077×10^{-5}

29	5.4564×10^{-5}	1.9019×10^{-5}	5.0812×10^{-5}
30	3.5845×10^{-5}	2.6031×10^{-5}	2.6796×10^{-5}

Table D.4 Mean squared error value of the neural network inverse model: 2 layers

Nodes in 1st Hidden Layer	Nodes in 2nd Hidden Layer	Mean square error (MSE)		
		Testing 1	Testing 2	Validating
2	2	1.2854×10^{-4}	1.8138×10^{-4}	2.7845×10^{-4}
2	4	4.7814×10^{-5}	4.6166×10^{-5}	8.0396×10^{-5}
2	6	9.2592×10^{-5}	1.1665×10^{-4}	2.0055×10^{-4}
2	8	6.3215×10^{-5}	3.1069×10^{-4}	3.3293×10^{-4}
2	10	4.7261×10^{-5}	7.1462×10^{-5}	1.0506×10^{-4}
2	12	1.1136×10^{-5}	1.0746×10^{-5}	1.3370×10^{-5}
2	14	2.8641×10^{-5}	2.2946×10^{-5}	2.7543×10^{-5}
2	16	3.8795×10^{-5}	4.5988×10^{-5}	5.7267×10^{-5}
2	18	5.7374×10^{-5}	9.0723×10^{-5}	1.0914×10^{-4}
2	20	7.7564×10^{-5}	3.4993×10^{-4}	3.6291×10^{-4}
4	2	2.3498×10^{-5}	2.1960×10^{-5}	3.5731×10^{-5}
4	4	5.6127×10^{-5}	5.2628×10^{-5}	1.0683×10^{-4}
4	6	3.0501×10^{-5}	3.6601×10^{-5}	5.4781×10^{-5}
4	8	2.3760×10^{-5}	2.4529×10^{-5}	3.1269×10^{-5}
4	10	1.9128×10^{-5}	1.8378×10^{-5}	2.0734×10^{-5}
4	12	2.7579×10^{-5}	2.8297×10^{-5}	2.4806×10^{-5}
4	14	1.9299×10^{-5}	7.0348×10^{-5}	5.7984×10^{-5}
4	16	2.1000×10^{-5}	2.3315×10^{-5}	2.1981×10^{-5}
4	18	2.3915×10^{-5}	2.2139×10^{-5}	1.9696×10^{-5}
4	20	3.8257×10^{-5}	4.2760×10^{-5}	6.1646×10^{-5}
6	2	1.2432×10^{-3}	1.4411×10^{-3}	2.2888×10^{-3}
6	4	1.8704×10^{-5}	1.7533×10^{-5}	2.8070×10^{-5}
6	6	1.6095×10^{-5}	1.4716×10^{-5}	1.5937×10^{-5}
6	8	1.2001×10^{-5}	5.8153×10^{-5}	4.1113×10^{-5}
6	10	2.3047×10^{-5}	2.5435×10^{-5}	4.4809×10^{-5}
6	12	2.1304×10^{-5}	1.4459×10^{-5}	2.1802×10^{-5}
6	14	1.2903×10^{-5}	1.4557×10^{-5}	1.6791×10^{-5}
6	16	1.9970×10^{-5}	1.8929×10^{-5}	2.0920×10^{-5}
6	18	1.4267×10^{-5}	2.3553×10^{-5}	1.6972×10^{-5}
6	20	1.2869×10^{-5}	1.2776×10^{-5}	3.4611×10^{-5}
8	2	2.0426×10^{-5}	2.0179×10^{-5}	1.7998×10^{-5}
8	4	1.5057×10^{-5}	1.0550×10^{-5}	1.2901×10^{-5}
8	6	1.9577×10^{-5}	1.9679×10^{-5}	2.7125×10^{-5}
8	8	1.2838×10^{-5}	7.8442×10^{-5}	5.9341×10^{-5}
8	10	1.6998×10^{-5}	2.1322×10^{-5}	3.2040×10^{-5}
8	12	2.0937×10^{-5}	2.3008×10^{-5}	2.8916×10^{-5}
8	14	1.0787×10^{-5}	3.3123×10^{-5}	3.2644×10^{-5}
8	16	2.8798×10^{-5}	2.1136×10^{-5}	2.9163×10^{-5}

8	18	1.7979×10^{-5}	1.4267×10^{-5}	2.1227×10^{-5}
8	20	2.0990×10^{-5}	1.7203×10^{-5}	1.8555×10^{-5}
10	2	1.2732×10^{-5}	1.3942×10^{-5}	2.0427×10^{-5}
10	4	1.3812×10^{-5}	1.1800×10^{-5}	1.6865×10^{-5}
10	6	1.3385×10^{-5}	3.3773×10^{-5}	3.2884×10^{-5}
10	8	2.1183×10^{-5}	1.8260×10^{-5}	1.4573×10^{-5}
10	10	1.8103×10^{-5}	1.7502×10^{-5}	2.0719×10^{-5}
10	12	1.4852×10^{-5}	1.9474×10^{-5}	4.2248×10^{-5}
10	14	1.5016×10^{-5}	1.8445×10^{-5}	1.6168×10^{-5}
10	16	1.1702×10^{-5}	4.9834×10^{-5}	3.8775×10^{-5}
10	18	1.4220×10^{-5}	1.2310×10^{-5}	1.8520×10^{-5}
10	20	1.1141×10^{-5}	2.6947×10^{-5}	2.3627×10^{-5}
12	2	3.4986×10^{-3}	3.9039×10^{-3}	4.1490×10^{-3}
12	4	1.3022×10^{-5}	2.6250×10^{-5}	2.2613×10^{-5}
12	6	1.4850×10^{-5}	1.5759×10^{-5}	2.0696×10^{-5}
12	8	9.4699×10^{-6}	8.3586×10^{-6}	1.1091×10^{-5}
12	10	1.8461×10^{-5}	2.2259×10^{-5}	1.8449×10^{-5}
12	12	2.9311×10^{-5}	1.9860×10^{-5}	1.1082×10^{-5}
12	14	1.1027×10^{-5}	2.3043×10^{-5}	2.2089×10^{-5}
12	16	7.7131×10^{-6}	6.6468×10^{-6}	1.6977×10^{-5}
12	18	1.7178×10^{-5}	1.5570×10^{-5}	1.2837×10^{-5}
12	20	2.0907×10^{-5}	3.1363×10^{-5}	1.8308×10^{-5}
14	2	2.1356×10^{-5}	2.0351×10^{-5}	1.5224×10^{-5}
14	4	2.0302×10^{-5}	1.9836×10^{-5}	1.2410×10^{-5}
14	6	1.1831×10^{-5}	5.4673×10^{-5}	5.9107×10^{-5}
14	8	1.1311×10^{-5}	3.4807×10^{-5}	3.2663×10^{-5}
14	10	6.4244×10^{-6}	2.5199×10^{-5}	2.4304×10^{-5}
14	12	1.0375×10^{-5}	1.1069×10^{-5}	1.7415×10^{-5}
14	14	1.2070×10^{-5}	1.6976×10^{-4}	1.2480×10^{-4}
14	16	1.5512×10^{-5}	2.5300×10^{-5}	2.9279×10^{-5}
14	18	1.1423×10^{-5}	2.1559×10^{-5}	2.6248×10^{-5}
14	20	1.1249×10^{-5}	2.9493×10^{-5}	2.4623×10^{-5}
16	2	1.2495×10^{-5}	1.1004×10^{-5}	2.2864×10^{-5}
16	4	1.7595×10^{-5}	1.2679×10^{-4}	9.2085×10^{-5}
16	6	1.9690×10^{-5}	1.5665×10^{-5}	1.5324×10^{-5}
16	8	1.4520×10^{-5}	1.3869×10^{-4}	1.1121×10^{-4}
16	10	1.9537×10^{-5}	1.4045×10^{-5}	2.1754×10^{-5}
16	12	1.0274×10^{-5}	3.7666×10^{-5}	2.8254×10^{-5}
16	14	9.4668×10^{-6}	8.4213×10^{-6}	1.2593×10^{-5}
16	16	1.2728×10^{-5}	1.1209×10^{-5}	1.2183×10^{-5}
16	18	1.3387×10^{-5}	1.6241×10^{-5}	1.9925×10^{-5}
16	20	1.2653×10^{-5}	1.8550×10^{-5}	1.7708×10^{-5}
18	2	4.4545×10^{-4}	4.5087×10^{-4}	7.6278×10^{-4}
18	4	1.4713×10^{-5}	1.6903×10^{-5}	2.8948×10^{-5}
18	6	9.7432×10^{-6}	1.6086×10^{-5}	2.2264×10^{-5}
18	8	1.1965×10^{-5}	1.3713×10^{-5}	2.0425×10^{-5}

18	10	1.2086×10^{-5}	2.0246×10^{-5}	1.5270×10^{-5}
18	12	9.4217×10^{-6}	1.7299×10^{-5}	1.6088×10^{-5}
18	14	1.6815×10^{-5}	1.8562×10^{-5}	1.7054×10^{-5}
18	16	1.0268×10^{-5}	1.4965×10^{-5}	1.8288×10^{-5}
18	18	2.0788×10^{-5}	2.0653×10^{-5}	2.2088×10^{-5}
18	20	1.4056×10^{-5}	5.1414×10^{-5}	4.1619×10^{-5}
20	2	2.4115×10^{-5}	1.8761×10^{-5}	3.6002×10^{-5}
20	4	1.8038×10^{-5}	1.7876×10^{-5}	1.7633×10^{-5}
20	6	1.0925×10^{-5}	1.8703×10^{-5}	1.2474×10^{-5}
20	8	1.0809×10^{-5}	5.9887×10^{-5}	5.0402×10^{-5}
20	10	1.6433×10^{-5}	1.4560×10^{-5}	1.2805×10^{-5}
20	12	9.2806×10^{-6}	9.5827×10^{-6}	2.8928×10^{-5}
20	14	1.1638×10^{-5}	2.9143×10^{-5}	4.1149×10^{-5}
20	16	8.9123×10^{-6}	1.5661×10^{-5}	1.7695×10^{-5}
20	18	1.2854×10^{-5}	6.8700×10^{-5}	5.0646×10^{-5}
20	20	8.8964×10^{-6}	1.8117×10^{-5}	1.4916×10^{-5}

VITA

Jedsada Thampasato was born on January 2, 1987 in Chonburi, Thailand. After finished high school from Chonrasadornumrung School in Chonburi, he entered the King Mongkut's University of Technology Thonburi in May 2005 and received his Bachelor's Degree of Chemical Engineering in Chemical Engineering from the Department of Chemical Engineering, Faculty of Engineering in April 2009. He began his graduate studies in May 2010 when he entered the Graduate School of Chulalongkorn University and joined the process control engineering group in the Department of Chemical Engineering.