DESIGN OF NEURAL NETWORK BASED MODEL PREDICTIVE CONTROL FOR HEAT
EXCHANGERS UNDER PARAMETRIC UNCERTAINTY

By

MISS Ratchaneekorn SROIPETCH

A Thesis Submitted in Partial Fulfillment of the Requirements

for Master of Engineering (CHEMICAL ENGINEERING)

Department of CHEMICAL ENGINEERING

Graduate School, Silpakorn University

Academic Year 2021

การออกแบบการควบคุมเชิงทำนายที่อาศัยแบบจำลองโครงข่ายประสาทเทียมสำหรับ
เครื่องแลกเปลี่ยนความร้อนภายใต้ความไม่แน่นอนของพารามิเตอร์

โดย
นางสาวรัชนีกร สร้อยเพ็ชร์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมเคมี แผน ก แบบ ก 2 ระดับปริญญามหาบัณฑิต
ภาควิชาวิศวกรรมเคมี
บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร
ปีการศึกษา 2564

DESIGN OF NEURAL NETWORK BASED MODEL PREDICTIVE CONTROL FOR

HEAT EXCHANGERS UNDER PARAMETRIC UNCERTAINTY

By

MISS Ratchaneekorn SROIPETCH

A Thesis Submitted in Partial Fulfillment of the Requirements

for Master of Engineering (CHEMICAL ENGINEERING)

Department of CHEMICAL ENGINEERING

Graduate School, Silpakorn University

Academic Year 2021

Title	Design of neural network based model predictive control for heat exchangers under parametric uncertainty
By	Ratchaneekorn SROIPETCH
Field of Study	(CHEMICAL ENGINEERING)
Advisor	Assistant Professor Veerayut Lersbamrungsuk , D.Eng.

Graduate School Silpakorn University in Partial Fulfillment of the Requirements for the Master of Engineering

---Dean of graduate school

(Associate Professor Jurairat Nunthanid, Ph.D.)

Approved by

---Chair person

 (Assistant Professor Tarawipa Puangpetch , Ph.D.)

---Advisor

 (Assistant Professor Veerayut Lersbamrungsuk , D.Eng.)

---Committee

 (Nutchapon Chotigkrai , D.Eng.)

---External Examiner

 (Assistant Professor Pornsiri Kaewpradit , D.Eng.)
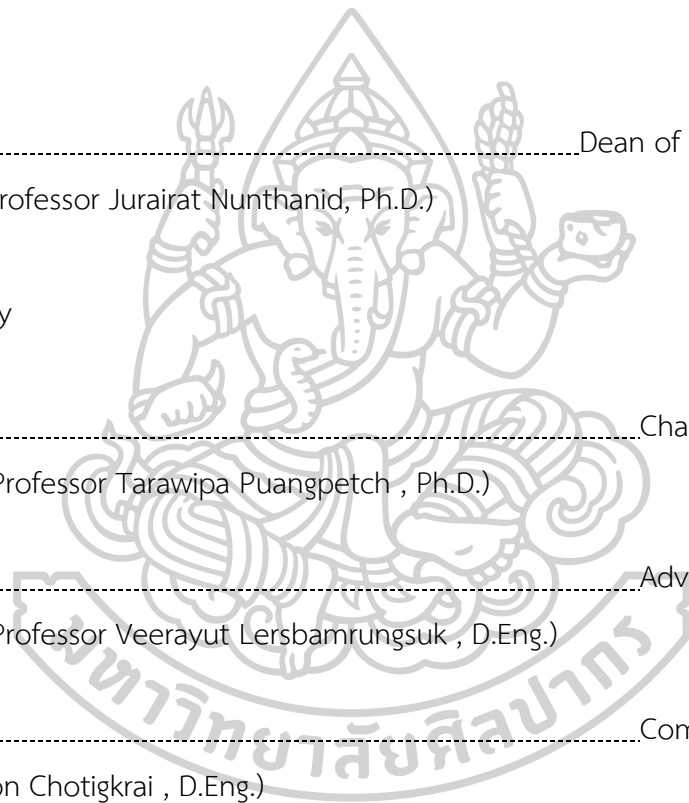
61404206 : Major (CHEMICAL ENGINEERING)

Keyword : Artificial neural network, Neural network-based model predictive control, Heat exchangers, Parametric uncertainty

MISS RATCHANEEKORN SROIPETCH : DESIGN OF NEURAL NETWORK BASED MODEL PREDICTIVE CONTROL FOR HEAT EXCHANGERS UNDER PARAMETRIC UNCERTAINTY THESIS ADVISOR : ASSISTANT PROFESSOR VEERAYUT LERSBAMRUNGSUK, D.Eng.

Fouling is one of the main problems that often arise during the operation of heat exchangers. The presence of fouling results in a reduction of heat transfer efficiency and can cause a temperature control problem. To overcome this problem, the design of the controller for the heat exchanger under fouling is therefore essential. In this study, an artificial neural network (ANN) was used to predict the fouling factor and identify a system for heat exchangers under parameter uncertainty in a neural network-based model predictive control (NNMPC) design with a nonlinear autoregressive network with exogenous inputs (NARX). The ANN training dataset was obtained from a cell-based dynamic heat exchanger model integrated with the threshold fouling model. The inputs of the ANN model included the flow rates of hot and cold streams and the inlet temperatures of hot and cold streams while the outputs of the ANN model included the outlet temperatures of hot and cold streams and fouling factors. Effects of the number of hidden neurons and training algorithms to ANN topology were also studied. The statistical indices used to determine the best topology include the mean square error (MSE), the regression coefficient (R^2), and processing time. In the controller design, the NNMPC and PID controller are used to control the temperature of the heat exchanger system. It has been shown that the NNMPC can control the temperature for the heat exchanger under fouling build-up better than the PID controller in terms of setpoint tracking and disturbance rejection. It can also use to predict the effect of fouling factors.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Motivation

Heat exchangers are equipment used for transferring the thermal energy between two or more fluids directly or indirectly for heating or cooling system. Heat exchangers are widely used in industries including petroleum refineries, power engineering, chemical industries, food factories, and others. The most common types of heat exchangers in industries are shell-and-tube type, suitable for systems that require large mass flows and the required pressures.

One of the major problems that often arise during the operation of heat exchangers is fouling formation which is generally defined as the accumulation of undesired materials such as bacteria, algae, solid particles, corrosion agent, etc., on the heat transfer surface, resulting in reduced heat exchange efficiency as increase of the fouling layer, decrease of the thermal conductivity of the metal surface, and increase of the resistance to heat transfer. In terms of economy, there is a need for maintenance and cleaning, indicating that the process has to be shut down. This introduces a loss of production for a while, a loss of return on investment in production equipment and a decrease in profitability.

The mechanisms of fouling is very complex. There was a large force acting that results in a more fouling layer in the heat exchanger. These are gravity, drag, buoyancy, Van der Waals forces, thermophoresis, Brownian motion, and adhesion/cohesion. These action forces were dependent on properties such as foulant diameter, foulant density, foulant type, Reynolds number, Prandtl number, etc. It also involved the shape and structure of the aggregates, such as the flow conditions of the heat exchanger and various environmental factors [1].

There are several methods to detect a fouling including experimental methods such as simultaneous observations of pressure drops, temperature

measurements, electrical measurements, weighing of heat exchanger plates, etc. but these methods have limitation such as the value of interest must be constant and long enough for measurement or having an expensive cost. Other methods are numerical model-based methods such as extended kalman filter (EKF), artificial neural network (ANN).

Several studies attempted to explain the fouling in heat exchangers in industries and to predict fouling factor. [2], [3], [4], and [5] proposed a method for predicting the crude oil fouling rate, known as the threshold model, by presenting it as a form of the Arrhenius equation. Some research groups [6], [7], and [8] proposed to use artificial neuron networks (ANN) to predict fouling factor. Experimental data collected from literatures were used in the ANN training. It has been proved that ANN is a powerful tool in the domain of prediction and analysis of data.

As fouling can affect heat transfer efficiency in heat exchangers, this can make the temperature control of heat exchangers more difficult. Furthermore, fouling changes along operation time and this implies that temperature control of heat exchangers can be considered as control problem under parametric uncertainty. To handle this problem, a model-based control such as model predictive control is needed.

In this research, an artificial neural network (ANN) for fouling estimation was developed. The data for ANN model was obtained from simulation of heat exchangers with a fouling model under various operating conditions. Then the proposed ANN model was incorporated into model predictive control (MPC), so called neural network model predictive control (NNMPC) for temperature control of heat exchangers under parametric uncertainty.

1.2 Objective of Research

1.2.1 To propose artificial neural network (ANN) model for fouling factor estimation.

1.2.2 To design a neural network based model predictive control for temperature control heat exchangers under parametric uncertainty.

1.3 Scope of Research

1.3.1 Perform simulation of heat exchangers under the threshold fouling model.

1.3.2 Develop ANN model for fouling estimation in heat exchangers under fouling condition.

1.3.3 Design PID controller for temperature control of heat exchangers under fouling condition.

1.3.4 Design ANN based MPC control for temperature control of heat exchangers under fouling condition.

1.4 Contribution of Research

1.4.1 A proposed ANN model for fouling estimation in heat exchangers under fouling condition.

1.4.2 A proposed neural networks-based model predictive control for temperature control of heat exchangers under fouling condition.

1.5 Nomenclature

Symbols

A_{Cell}	heat transfer surface (m^2)
C_p	specific heat capacity (J/kg K)
dR_f/dt	fouling rate (m^2 K/W)
E	activation energy (KJ/mol)
h	heat transfer coefficient (W/m^2 K)
h_f	heat transfer coefficient include thermal resistance of fouling (W/m^2 K)
m	mass flow rate
mh_{Cell}	mass holdup of the fluid in a modeling cell tank
mh_{Tank}	mass holdup of the fluid in a modeling cell tank

Pr	Prandtl number
Q_{Cell}	the rate of heat transfer through the cell wall into or out from a modeling cell tank
R	gas constant (J/mol K)
Re	Reynolds number
t	time (h)
T	temperature (°C)
v	volumetric flow rate (m^3/h)
V_{Cell}	volume (m^3)

Greek letters

α	constant value
α_{Cell}	film heat transfer coefficient in a modeling cell tank
β	constant value
γ	constant value
ρ	density (kg/m^3)
τ_w	the shear stress at the heat transfer surface

Subscripts

b	bulk
C	cold stream
f	film
H	hot stream
In	Inlet of the fluid in a modeling cell tank
Out	outlet of the fluid in a modeling cell tank
W	tube wall

CHAPTER II

LITERATURE REVIEWS

2.1 Dynamics of heat exchanger model

Mathisen et al. [9] investigated the process control and controllability of heat exchangers and heat exchanger networks using a cell modeling technique. However, critical model characteristics must be supplied. It distinguishes between significant and minor model characteristics based on the sequence of arguments, the comparison of controllability measures, and dynamic simulation. The series order, wall capacity, and compressibility of the fluid are critical elements of models for single heat exchangers, whereas the flow arrangement and temperature propulsion have little impact on the dynamics. For a heat exchanger network, the most critical model characteristic is the break time of the linked pipes.

Varga et al. [10] used cell modeling as a heat exchanger network model to investigate the structural control properties of the heat exchanger network using time-varying parameters such as control properties, i.e., stability, observability, and controllability, as determined from qualitative data on heat exchangers and their network topology. The extended Kalman type rating criteria for linear systems with time-varying parameters are used to derive the required and sufficient requirements for structural and observable control of the heat exchanger network. Determining a heat exchange network's structural control and observability needs just examining the network's input and output connectivity capabilities for static and time-varying parameter scenarios. The results are expanded to more realistic situations in which bypass ratios are utilized as control variables and state-space matrices contain many time-varying parameters.

Georgiadis et al. [11] used a sophisticated dynamic model to offer a mathematical model and simulation of a complicated plate heat exchanger configuration subjected to milk fouling. Complex fouling model based on reaction/mass transfer theory and a conventional plate heat exchanger thermal

dynamic model. The final model is composed of a sequence of partial differential equations, integrals, and a small amount of algebra. The analysis of parameters is conducted by the dynamic optimization solution. The simulation findings are consistent with those obtained from previous studies. To define fouling behavior, three distinct approaches involving complicated flow manipulation were investigated. The simulation results shed light on the critical aspects that contribute to milk fouling.

Roetzel et al. [12] established a generic model for a one-dimensional flow heat exchanger that uses the Laplace transform and a numerical inverse method to solve issues. This model applies to multi-stream multi-pass shell-and-tube and plate heat exchangers, as well as their networks. The time delays in the pipe connection are included in this model on the assumption that they are constant and the heat capacity of the pipes is insignificant. To circumvent these assumptions, one can use the pipe connection as an extra exchanger channel. To anticipate the temperature response of heat exchangers, a simplified form of the transfer function resulting from the lumped parameter model is proposed, where parameters are determined by experimental data or the findings of the distributed parameter model.

Ansari et al. [13] used a numerical technique based on an analytical solution of the energy equation to model the dynamic response of a counter-current heat exchanger. A simulation of a counter-current heat exchanger was created using the work of [14]. The heat exchanger consists of N cells. Each cell has components for the hot and cold fluids, as well as the tube wall. At all intervals, it is assumed that the energy equation coefficient of each cell is constant. The temperature distribution of hot and cold liquids and walls at various periods in all cells may be determined by solving the analytical power equations. A numerical approach may be used to apply this temperature distribution across all cells to all heat exchangers. Additionally, in two circumstances, the system's transitory behavior is evaluated. The first scenario examines the system's response to changes in the two fluids' intake

temperatures. The second scenario analyzes the effect of changing the incoming mass flow rate on the system's behavior. During the dynamic behavior of the system, the time constant is taken into account.

It demonstrates that in the flow direction, the fluid time constant at which the mass flow rate changes grows linearly. On the other hand, an undisturbed fluid's mass flow rate displays two distinct sorts of reactions. When the step change is minor, the time constant function is homogeneous across the heat exchanger. However, when the number of changes is increased, the time constant function becomes a line with an increasing slope in the direction of the flow. The results were compared to those obtained via analysis and to those obtained through experimentation. These findings demonstrated that the approach was valid and that the heat exchanger's transient behavior could be replicated.

Dobos et al. [15] employed a cell model technique to simulate the heat exchanger's dynamic behavior. Their objective was to model the spatial heating network while also incorporating it into a model predictive controller. It is utilized to satisfy the heating requirements of network consumers. It will prioritize two primary objectives. The first is dynamic modeling, which can be used to illustrate the primary characteristics of a decentralized heating network; the second is to develop a nonlinear model predictive controller (NLMPC) capable of meeting consumer heating needs in heat exchanger networks. Due to the model's predictive controllers' limited usage of objective functions, it is ideal to develop strategies to decrease redundant energy consumption and maximize the benefit of freely deployed industrial waste heat. Along with environmental concerns, conserving energy may also cut operational expenses. Following that, [16] must lower the transition time in DHN predicted by the nonlinear model stated above by modifying the parameters of the nonlinear MPC and simulating a heat exchanger using the cell model approach. The cost function is used to determine the controller's efficiency while considering the needed system restrictions. Utilize the simplex approach, a well-known test design

technique, to optimize this cost function. This optimization approach is capable of solving mixed-integer optimization problems, which are necessary since prediction and control horizons must be integer values. As a result of the periodic nature of heat demand, the suggested technique may readily be integrated into an iterative learning control system.

Varbanov et al. [17] proposed cell-based dynamic model to describe the dynamic heat exchanger and the visualization technique for determining the number of the modelling cells and their size. A simple heat exchange cell in their work consists of the two perfectly mixed tank, exchanging heat only with each other that are divided by the wall. Cell-based model proposed in their research was acquired from the energy balance of tank in a cell by some assumptions for easy to calculating. This method can calculate the cell number and heat transfer coefficient for a more general case, and also provide thermodynamic reasoning, which is important for implementing engineering tasks.

Zhang et al. [18] suggested a dynamic model that is suited for diagnostic purposes. The HEX/reactor under study and this model can assure the model's correctness by accounting for the effect of mass flow rate and fouling on the total heat transfer coefficient. As a model HEX/reactor, a cell is employed. This approach is capable of detecting, isolating, and identifying sensor, actuator, and process parameter-related faults. This is because it enables the source of a mistake in the total heat transfer coefficient to be determined (fouling or fluid flow rate). The simulation results indicate that the approach is capable of providing enough detection and diagnostic capabilities for the safety of HEX/reactor.

2.2 Models for predicting fouling

Based on an analysis of Exxon crude oil fouling data, Ebert and Panchal [2] proposed a ground-breaking approach called the threshold prediction model. They predicted the rate of deposition using film temperature, but in more recent studies, Heat

Transfer Research Incorporated (HTRI) established the rate of deposition using skin temperature because it shows better trending than film temperature. High variability in activation energy, according to the study, is a challenge. Various researchers have proposed various figures. They tested a specific crude oil with an activation energy of 69 kJ/mole. The modification proposed by Ebert and Panchal is written as

$$\frac{dR_f}{dt} = \alpha Re^\beta exp\left(\frac{-E}{RT_f}\right) - \gamma \tau_W \qquad (1)$$

The first term is related to chemical reaction and film temperature, while the second term depends on the velocity and surface shear stress. Film temperature (T_f) is described as follows.

$$T_f = T_b + 0.55(T_w - T_b) \qquad (2)$$

where T_b and T_w are the bulk temperature and the wall temperature, respectively.

Panchal [3] assumed that fouling removal is a matter of mass transfer rather than surface shear stress, as Crittenden [19] established in his work. Panchal adds the Prandtl number Pr to Ebert and Panchal's model to increase the content of specific heat and thermal conductivity. According to the heat transfer relationship, the power value of Pr is set at -0.33. Panchal's modification is denoted by the following:

$$\frac{dR_f}{dt} = \alpha Re^\beta Pr^{-0.33} exp\left(-E/RT_f\right) - \gamma \tau_w \qquad (3)$$

Polley et al. [4] compared Panchal's model predictions to Knudsen's [20] experimental fouling data and concluded that the anticipated threshold temperature is greater than the actual value due to the effect of flow velocity. The disparity between the anticipated and measured values of the threshold temperature increases fast as the flow velocity increases. As a result, they modify the Panchal model to create a new correlation. It's as follows:

$$\frac{dR_f}{dt} = \alpha Re^\beta Pr^{-0.33} exp(-E/RT_W) - \gamma Re^{0.8} \qquad (4)$$

In the Arrhenius equation, the wall temperature T_W is used to replace the film temperature. This study is an attempt to model measured physical parameters that resulted in no observable fouling. $\gamma Re^{0.8}$ is used to replace the surface shear stress in the mass transfer mechanism. Polley validated the predictive function by comparing the predicted value to the experimental results obtained by [20], [3], and [21] under the same operating conditions.

Brahim et al. [22] contributed to the prediction of fouling by predicting crystal formation on flat and structured surfaces of heat transfer equipment using a Computational Fluid Dynamics (CFD) solver. According to sources, the simulation does not validate what was referred to as the fouling process calculation. The numerical simulation yielded a forecast of fouling resistance over a specified period. However, the simulation tends to underestimate the fouling behavior in the majority of situations. The reason for this is that the CFD solver used computes temperatures at the phase boundary crystal layer/salt solution at unacceptably low values.

Liporace and de Oliveira [23] created a methodology for assessing the real-time performance of a crude oil preheat train. The tool made extensive use of complex process simulations and intricate heat exchanger design to guarantee that the global heat transfer coefficient (operational and clean modes) and fouling factor could be analyzed accurately. This was accomplished with the use of a process simulator called Petrox from Petrobras and HTRI, which resulted in fouling factors that were similar to real-world values. This has increased the performance of heat exchanger design.

Nasr et al. [5] present a novel model for crude oil fouling in crude distillation units' preheat exchangers. We used experimental data from Australian light crude oil with tube side surface temperatures ranging between 200 and 260 °C and fluid velocity ranging between 0.25 and 0.4 m/s. The activation energy required has been computed as a function of the surface temperature. A novel model was suggested

that included terms for fouling development and fouling removal caused by chemical and tube wall shear stress, respectively. The main superiority of the model are independent to Pr number, thermal fouling removal and determination of β based on experimental tests. Finally, the fouling rate of Australian light crude oil was computed using the suggested model, and threshold curves for identifying fouling and non-fouling formation zones were created. The following is a modified correlation:

$$\frac{\mathrm{d}R_f}{\mathrm{d}t} = \alpha Re^{\beta} Pr^{-0.33} exp\left(-E/RT_f\right) - \gamma Re^{0.4} \qquad (5)$$

The model performs significantly better than others in forecasting deposit rates for various kinds of crude oil and under various scenarios. By providing a model, designers may readily establish whether the heat exchanger is positioned in the fouling zone and build threshold curves. Additionally, this curve aids in determining the location of the heat exchanger inside the fouling zone and how to push it out of there.

Haghshenasfard et al. [24] developed a two-component, multi-component fluid model in conjunction with a heat-flux partition wall model for the prediction of CaSO4 fouling in sub-cooled flow boiling circumstances. The authors investigate the effects of bulk and surface temperatures, roughness, and fluid velocity on fouling behavior, heat transfer coefficient, and fouling resistance. The models' predictions were evaluated and compared to the experimental data from [25]. The working conditions are as follows: fluid velocity between 0.5 and 2 m/s, surface temperature between 95 and 140oC, total temperature between 65 and 95°C, and concentration between 1.6 and 2.7 g/L.

2.3 Artificial neural network modelling

2.3.1 Estimation of heat exchangers parameters

Jambunathan et al. [26] utilized a feed-forward neural network to estimate the convective heat transfer coefficients at a site in hot air-heated pipes based on

experimental data obtained using liquid crystal thermography (LCT). Three inputs are used in the network: non-dimensional temperature, thermal diffusivity, and time, while the output is the heat transfer coefficient. As a network training algorithm, the Gradient Descent approach is utilized. The configuration with 3-6-3-1 was chosen as a network for predicting heat transfer coefficients because it had a percent deviation of ±2.7% from the experimental values.

Malayeri and Müller-Steinhagen [27] demonstrated how to employ neural network analysis to forecast fouling behavior during sub-cooled flow boiling. It discussed the benefits and drawbacks of neural network design when applied to fouling data. The predictability of the network has been demonstrated to be promising when dominating factors are understood and also appropriate data is supplied to the network.

Ghajar et al. [28] created a feed-forward neural network for the prediction of horizontal heat exchanger heat transfer coefficients (with three inlet configurations, i.e., reentrant, square-edged, and bell-mouth inlet under uniform wall heat flux boundaries condition). The network accepts five inputs: the Reynolds number, the Prandtl number, the Grashof number, the x/D value, and the μ ratio, and outputs the heat transfer coefficient. A total of 1290 data points were used as training data for the network (441 for reentry, 416 for rectangular edges, and 433 for bells). The 5-11-1 feed-forward neural network is capable of accurately predicting the heat transfer coefficient to within a 5% deviation.

Romeo et al. [29] employed a hybrid approach of artificial neural networks and fuzzy logic to manage the fouling factor and maximize boiler performance while reducing the influence of fouling in biomass boiler heat exchangers. This hybrid system employs many neural networks for various tasks, including boiler monitoring, soiling prediction, boiler behavior prediction, and cleaning outcomes if a soot-blowing cycle is started. The authors demonstrated that by utilizing a hybrid ANN

technique to regulate the fouling rate in boilers, power production may be increased by around 3.5%. Between the output of the hybrid system and the data supplied by the biomass boiler, the validation reveals considerable energy savings.

Aminian et al. [6] compared the ANN used to estimate the fouling factor to the fouling rates predicted by the three threshold models. The ANN model is trained using data from the literature. The training data set was created using experimental results published in the literature. Three inputs are used in the network: the tube diameter (d_t), the crude velocity, and the tube surface temperature. To anticipate the fouling factor, they picked a network design with 3-5-6-1 structures. The overall mean relative error (OMRE) of the ANN model was determined to be the lowest when compared to the three threshold model fouling rates presented in *Table 1*. Additionally, they compared the ANN used to estimate the fouling factor to the fouling rates predicted by the threshold model. The fouling factor in crude oil may be predicted using three inputs: the Reynolds number (Re), the Prandtl number (*Pr*), and the tube surface temperature. The performance was attained using a three-layer network with 3-8-1 structures. The value of OMRE was determined to be 14.05%, 22.47%, and 15.83%, respectively, for training, testing, and experimental findings [7].

Lalot et al. [30] demonstrated how to use ANN to identify fouling in a cross-flow heat exchanger. A numerical model was employed in their research to produce data for training under clean and fouling circumstances. The network's five inputs are the inlet and outlet temperatures of cold fluids, the inlet temperature of hot fluids, the mass flow rate of cold fluids, and the mass flow rate of hot fluids, with just the fouling factor as an output. It was revealed that ANN modeling is highly sensitive for forecasting the fouling factor while requiring less computing time than the standard technique.

Table 1. Comparison of the OMRE value between the three threshold fouling rates and ANN model.

Authors	Model	OMRE
Panchal et al., 1997	Panchal model fouling rate: $$\frac{\mathrm{d}R_f}{\mathrm{dt}} = \alpha Re^\beta Pr^{-0.33} exp(-E/RT_f) - \gamma\tau_w$$	47.90%
Polley et al., 2002	Polley model fouling rate: $$\frac{\mathrm{d}R_f}{\mathrm{dt}} = \alpha Re^\beta Pr^{-0.33} exp(-E/RT_w) - \gamma Re^{0.8}$$	75.36%
Nasr et al., 2006	Nasr model fouling rate: $$\frac{\mathrm{d}R_f}{\mathrm{dt}} = \alpha Re^\beta Pr^{-0.33} exp(-E/RT_f) - \gamma Re^{0.4}$$	60.68%
Aminian et al., 2008	ANN model structure: 3-5-6-1	26.23%

Garcia [31] developed an ANN-based supervision technique to reduce the disadvantage of fouling in heat exchangers. Based on static and dynamic ANN approaches, the suggested supervision approach can identify, segregate, and accommodate defects in the closed-loop temperature management of a heat exchanger. The plan that was designed was broken into three sections. The first module examines the surveillance system's consistency. The second module continuously monitors the heat exchanger for fouling conditions and is capable of diagnosing potential fouling reasons. A third module forecasts the remaining operating time under acceptable circumstances, which is then used to arrange the supervisory flow chart. The developed supervision technique is a unique application of artificial neural networks for accurately detecting, separating, and forecasting heat exchanger fouling.

Vaferi et al. [32] introduced an artificial neural network (ANN) model for calculating the convection coefficient (HTC) of nanofluids flowing through spherical

tubes with varying wall conditions in various flow systems. A backpropagation learning approach was used to update the parameters of the ANN model using a variety of experimental datasets. The constructed ANN model has a mean squared error (MSE) of 1.7×10^{-5}, a percent mean absolute relative deviation (AARD%) of 2.41, and a regression coefficient (R^2) of 0.99966 when used to simulate the whole convective HTC experimental dataset. The suggested approach's prediction performance is compared to authoritative correlations proposed in the literature. The constructed MLP network's better predictive ability and empirical correlation superiority demonstrated that the suggested technique could be utilized to measure nanofluid convection HTC within a circular tube in real space.

Biyanto [33] modeled fouling resistance in shell-and-tube heat exchangers using Artificial Neural Networks (ANN) Multilayer Perceptron (MLP) with nonlinear autoregressive with exogenous (NARX) input structures. The input data includes the present flow rate and temperature of the heat exchanger, the physical parameters of the product, and coarse mixing data. This model is a prediction tool for optimizing shell and tube heat exchanger operating conditions and preventative maintenance. The findings indicated that the model accurately captured the complexity of fouling features in heat exchangers as a function of thermodynamic circumstances and variations in crude oil qualities (mixture). Root Mean Square Error (RMSE) was discovered to be an excellent indicator of the nonlinearity and complexity of shell-and-tube heat exchangers that cause fouling throughout the training and validation processes.

Davoudi et al. [8] predicted the fouling factor in the yearly heated probe and single tube heat exchangers under a variety of operating situations using a multi-layer perceptron network. The ANN model was constructed and verified using a massive database consisting of 11626 experimental datasets from six distinct pieces of literature. Density, surface temperature, fluid temperature, the width of the fluid channel, the velocity of the fluid, the concentration of dissolved oxygen in fluid, and

time are the seven inputs to the network. To estimate the fouling factor, the arrangement with 7-10-1 structures had the best performance. With an overall AARD% of 5.4, an MSE of 0.0013, an RMSE of 0.0355, and an R^2 of 0.977819, this ANN model predicts the experimental values of the fouling factor. The simplicity of the generated ANN model and its low error rates across a large experimental dataset are two of their model's important characteristics.

2.3.2 Modeling of heat exchanger

Wang et al. [34] developed a feed-forward neural network model to forecast the performance of shell-and-tube heat exchangers with segmented or continuous helical baffles using sparse experimental data. The network contains eight inputs: Reynolds number on the oil and watersides, inlet temperature on the oil and watersides, tube and baffle count, baffle pitch, and tube center diameter, while the output is the heat transfer rate. For network training, backpropagation is utilized. The network configuration with the identifier 8-6-5-1 is chosen. The difference between anticipated and observed values was less than 2%. The results indicated that the ANN-predicted heat transfer rate was extremely similar to the experimental data. This demonstrates that the neural network methodology is superior to empirical correlations for forecasting heat transfer rates.

Xie et al. [35] employed ANN to simulate shell-and-tube heat exchangers (with oil and water as working fluids), comparing the findings and correlations. The network contains eight inputs, including the Reynolds number on the oil and watersides, the inlet temperature on the oil and watersides, the tube and baffles count, baffle pitch, and tube center diameter, and three outputs: heat transfer rate, the temperature differential between the oil and watersides. For network training, backpropagation is utilized. As the network configuration, 8-6-5-3 is chosen. This ANN model predicts experimental values with an evaluation factor of 1.089 for average accuracy and 0.1387 for scattering accuracy, with a maximum relative error of less than 1.5% when compared to experimental findings.

Pandharipande et al. [36] modeled a shell-and-tube heat exchanger using a feed-forward neural network. The network comprises four inputs: temperatures of hot and cold fluids and the flow rates of hot and cold streams, and two outputs: temperatures of hot and cold streams at their outlets. For network training, the backpropagation algorithm is utilized. Network training is implemented in architecture 4-15-15-15-2, which may achieve a high degree of accuracy (98-99.5%) for predicted values of training and test datasets. When applied to model heat exchangers, ANNs have been found to yield acceptable results.

Mandavgane et al. [37] modeled a shell-and-tube heat exchanger with a feed-forward neural network. The network has four inputs: temperatures of hot and cold fluids and flow rates of hot and cold streams, and two outputs: outlet temperatures of hot and cold fluids. As with [36], the datasets are divided into two cases for training and testing: water-20% glycerin system and water-40% glycerin system. Backpropagation is used to train networks. The improved ANN design, 4-15-15-15-2, can predict the values of training and test datasets with a high degree of accuracy (98-99.5%). It has been demonstrated that when used to model heat exchangers, ANNs may yield satisfactory results.

Duran et al. [38] developed a feed-forward neural network for shell-and-tube heat exchanger cost calculation. This network topology has an input layer with five neurons representing tube pitch, shell diameter, tube diameter, rear head factor, and stationary head factor, and an output layer with one neuron representing the cost per exchange area. Their approach made use of the Levenberg-Marquardt algorithm and the log sigmoid transfer function. The network configurations 5-10-10-1 were found to be highly predictable, with an R-value of 0.97.

Fadare et al. [39] modeled the heat transfer coefficient of a staggered multi-row, multicolumn, cross-flow, tube-type heat exchanger using a feed-forward neural network. This airflow via copper pipes staggered in five rows and four columns at

varying air flow rates with a throttle opening of 10% to 100%. A network accepts two inputs: the Reynolds number and the row number, and produces the Nusselt number. It was stated that a feed-forward neural network with a 2-5-5-1 architecture was capable of predicting the Nusselt number and producing less than 1% and 4% MARE for the training and test datasets, respectively.

Kashani et al. [40] devised a model that enables operators and process engineers to forecast the complicated dynamic behavior of crude oil fouling in CDU industrial heat exchangers using only a few input variables and ANN moving windows. A critical component of this modeling methodology is that the ANN model is updated in real-time whenever a new block of data becomes available, which allows for effective capturing of the dynamics of slow-changing processes. It is quantifiable every day in all CDUs. Three inputs are used in the network: the tube side of crude oil flow rate and the tube and shell side input temperatures, while the output is the fouling factor. In-network training, factor backpropagation is used to modify weights so that the output vector's response to the input vector is as near to the intended response as feasible. They minimize computational burden by utilizing a modified leave-one-out cross-validation methodology combined with an early stop idea. The network was configured in a 3-5-1 to train the network. With an MRE of around 8%, this network is capable of anticipating the fouling factor two days in advance. These models' outputs were compared to relevant experimental data sets. The training and prediction subsets have a mean relative error (MRE) of roughly 6.61% and 8.06%, respectively. MRE has an accuracy of roughly 8% in forecasting fouling rates over the following 50 hours.

2.4 Neural network based model predictive control

Ławrynczuk [41] employed artificial neural networks to manage the temperature of a yeast fermentation biochemical reactor using model predictive control. To begin, the process's neural model was trained utilizing existing datasets obtained by the underlying model. Adjustment employs the optimal brain damage method to

simplify neural stimulation and enhance the neural network's prediction capabilities. Following that, a computationally efficient nonlinear model predictive control (MPC) method with nonlinear prediction and linearization (MPC-NPL) has been created. This approach requires an online solution for a quadratic programming problem. It was demonstrated that the approach achieves closed-loop control performance comparable to that of nonlinear MPCs using complete online non-convex optimization. The computational complexity of the MPC-NPL algorithm proved control precision, as well as the capacity to reject interference in the presence of noise and the resultant interference process.

Vasikaninová et al. [42] utilized a neural network predictive control (NNPC) structure to maintain a target outlet temperature and reduce energy consumption in a co-current tubular heat exchanger. The neural network was employed as a nonlinear process model in their work to forecast the future behavior of a controlled process with dispersed characteristics. Offline training of a neural network model with one hidden layer and six neurons is used to identify the system and forecast the future hot output temperature. To train the network, we used the Levenberg-Marquardt algorithm. To compute the appropriate control input, a predictive control was constructed. Additionally, we compared neural network predictive control to traditional PID control. The outcome established that the NNPC was an effective instrument for controlling the heat exchanger successfully. In comparison to standard PID control, NNPC enables more stable control responses with a shorter settling period. Additionally, this strategy benefited from regulated input and limitations, as well as serving as a tool for almost perfect control process behavior. As a result, consumption is reduced as compared to conventional PID control.

Pazhooh et al. [43] employed an adaptive neural network predictive controller (ANNPC) to regulate a Vinyl Acetate monomer (VAM) production process. The framework of the proposed controller incorporates an observer-based neural network model for assessing the unknown state. After then, adaptive predictors are

customized for the observer and utilized to forecast non-measurable states. For plant-wide process control, a new partly centralized structure has been devised. Furthermore, the suggested controller's efficiency, notably in mitigating the influence of measurement delay, is demonstrated in silicon through numerical simulations of the VAM plant under control. The findings are compared to those obtained using a straightforward PI control system for a VAM process.

Kimaev et al. [44] developed a multi-scale thin-film coating process using a neural network predictive control (NNPC) framework. They created multi-input, single-input, multi-input, nonlinear neural networks with external inputs to forecast the reaction of a multi-scale stochastic chemical engineering system and then manage it (NMPC). The reactant temperature and molar reactant percentage were both considered as factors. Thin-film roughness and growth rate were selected as response variables. The generated profiles were tested with a stochastic multi-scale system and found to be highly consistent with the ANN predictions. ANNs appear to be a potential solution for online optimization and control of computationally intensive multi-scale process systems due to their reported computational efficiency, accuracy, and interference rejection capabilities.

Shin et al. [45] built a neural network model predictive control (NNMPC) system and compared it to a PID controller on a depropanizer column at an offshore facility. Initially, using Aspen HYSYS, a mathematical model of the distillation column was built. A dynamic convolutional neural network (DNN) model was utilized to forecast the propeller reducer's future behavior. Finally, the generated model is connected with the Model Predictive Control System (MPC), therefore replacing the old mathematical model and enabling the control system to achieve stable, rapid, and accurate optimization outcomes. The investigation revealed that the MPC performed better (i.e., had a faster settling time and a shorter rising time) than the PID control group.

Jamil et al. [46] used the neural network predictive control (NNPC) as a novel technique for vibration control of tall structures employing a single degree of freedom active tuned mass damper (ATMD). The suggested approaches were compared to two contemporary control systems: adaptive neural inference regulators and terminal alignment regulators. The validity of the control methodology is confirmed using a scaled-down laboratory model. Nonlinear ARX models are used to develop neural network prediction controllers. The linear ARX model is used to construct polynomial and state-space pole-placement controllers. An adaptive neuro-fuzzy inference system is used to construct a fuzzy logic controller for structure and training (ANFIS). Each of these controllers was evaluated for vibration control using MATLAB and hardware-in-the-loop (HIL). The results indicated that the neural network prediction regulator significantly shortened the big damper's shock suppression time. Additionally, compared to the pole position controller, it takes extremely less control effort. As evidenced by the peak-to-peak control signal voltage being double that of the pole position regulator.

CHAPTER III

THEORY

3.1 Cell-based dynamic heat exchanger models

The dynamic model with cells [17] is a simplified depiction of a heat exchanger, described as two completely mixed tanks that exchange heat solely through a wall, as shown in *Figure 1*). The cell model may generate an infinite number of equations. However, this method establishes a consistent framework and modeling flexibility that enables the incorporation of any type of surface heat exchanger in any flow arrangement. The model's complexity may be adjusted by the number of cells, allowing for trade-offs between accuracy and the model's capacity to handle large and complicated process systems, such as heat exchanger networks. The following assumptions were used in general when developing a cell-based dynamic model:

(1) The two tanks in the heat exchanger cells are perfectly mixed, meaning that the temperature in each tank can be considered constant for space.

(2) The fluid densities are constant.

(3) The tank is filled with relevant or interest liquid.

(4) This is because the model mainly aims at controlling the temperature of the fluid. Therefore, the stream is effectively assumed to have a limited constant specific heat capacity, excluding pure vaporization or condensation process streams.

(5) The wall resistance to heat transfer is neglected as its temperature is considered uniform within the cell volume.
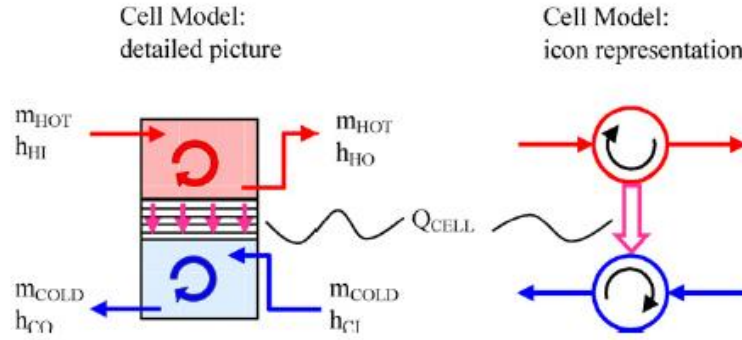
Figure 1. Modeling cell-based dynamic heat exchanger [17].

The energy balance of the tank within the cell, assuming that assumptions (2) and (3) are irrelevant, therefore obviating any changes in the amount of liquid retained. Which is equivalent to Eq. 6 [17].

$$mh_{Tank} \cdot C_{p,Fluid} \cdot \frac{dT_{Fluid,Out}}{dt}$$
$$= m_{Fluid} \cdot C_{p,Fluid} \cdot T_{Fluid,In} - m_{Fluid} \cdot C_{p,Fluid} \cdot T_{Fluid,Out} \pm Q_{Cell} \quad (6)$$

Q_{Cell} represents the rate of heat transfer through the cell wall between two tanks with the hot side tank lowered and the cold side tank raised. The heat transfer rates for the hot and cold side tanks are written as follows [17].

$$Q_{H,Cell} = \alpha_{H,Cell} \cdot A_{Cell} \cdot (T_{H,Out} - T_W) \quad (7)$$

$$Q_{C,Cell} = \alpha_{C,Cell} \cdot A_{Cell} \cdot (T_W - T_{C,Out}) \quad (8)$$

By substituting Q_{Cell} in Eq. 6, the following energy balances for hot, cold, and wall tanks are obtained:

$$mh_{H,Cell} \cdot C_{p,H} \cdot \frac{dT_{H,Out}}{dt}$$
$$= m_H \cdot C_{p,H} \cdot T_{H,In} - m_H \cdot C_{p,H} \cdot T_{H,Out} - \alpha_{H,Cell} \cdot A_{Cell} \cdot (T_{H,Out} - T_W)$$

$$(9)$$

$$mh_{C,Cell} \cdot C_{p,C} \cdot \frac{dT_{C,Out}}{dt}$$
$$= m_C \cdot C_{p,C} \cdot T_{C,In} - m_C \cdot C_{p,C} \cdot T_{C,Out} + \alpha_{C,Cell} \cdot A_{Cell} \cdot (T_W - T_{C,Out})$$

$$(10)$$

$$mh_W \cdot C_{p,W} \cdot \frac{dT_W}{dt} = \alpha_{H,Cell} \cdot A_{Cell} \cdot (T_{H,Out} - T_W) - \alpha_{C,Cell} \cdot A_{Cell} \cdot (T_W - T_{C,Out}) \quad (11)$$

Since a single cell cannot adequately depict the complete heat exchanger, numerous cells are often mixed according to the device's real flow sequencing. As a result of dividing all heat exchangers, the cell size, film heat transfer coefficient, and other characteristics will be computed. As a result, referring to the volume of the cell tank rather than the mass is more convenient. Equations 12-14 were used to denote the following [10].

$$\frac{dT_{H,Out}}{dt} = \frac{v_H}{V_{H,Cell}} \cdot (T_{H,In} - T_{H,Out}) - \frac{\alpha_{H,Cell} \cdot A_{Cell}}{V_{H,Cell} \cdot \rho_H \cdot C_{p,H}} (T_{H,Out} - T_W) \tag{12}$$

$$\frac{dT_{C,Out}}{dt} = \frac{v_C}{V_{C,Cell}} \cdot (T_{C,In} - T_{C,Out}) + \frac{\alpha_{C,Cell} \cdot A_{Cell}}{V_{C,Cell} \cdot \rho_C \cdot C_{p,C}} (T_W - T_{C,Out}) \tag{13}$$

$$\frac{dT_W}{dt} = \frac{\alpha_{H,Cell} \cdot A_{Cell}}{V_W \cdot \rho_W \cdot C_{p,W}} (T_{H,Out} - T_W) - \frac{\alpha_{C,Cell} \cdot A_{Cell}}{V_W \cdot \rho_W \cdot C_{p,W}} (T_W - T_{C,Out}) \tag{14}$$

3.2 Threshold fouling model

The threshold fouling model is used for predicting fouling rates. It has been developed to several models for predicting the fouling rate at various crude types and conditions. The models are controlled by two competing mechanisms, the first involved in the chemical reaction as the term of fouling formation and the second involved in the fluid velocity on the fouling removal. In this work, the researchers are interested in the threshold fouling model developed by Nasr and Givi because it can predict the fouling rate for different crude types and conditions better than other models. It can explain the equation as follows [5].

$$\frac{dR_f}{dt} = \alpha Re^\beta exp\left(\frac{-E}{RT_f}\right) - \gamma Re^{0.4} \tag{15}$$

where α=0.01098 m^{2}K/J

β=-1.547

γ=0.96×10^{-13} m^{2}K/J

E=22.618 kJ/mol

These parameters are the constant values calculated from experimental data of the Australian crude oil measured by Seleh and colleagues whose laboratory experimental data is an experiment with a recirculation system with an annular test probe [47]. Experimental conditions were determined using the surface temperature between 200 and 260 ℃ and fluid velocity between 0.25 and 0.4 m/s. Film temperature (T_f) is described as follows [4].

$$T_f = T_b + 0.55(T_w - T_b)$$
(16)

where T_b and T_w are the bulk temperature and the wall temperature, respectively.

The fouling model in the equation 15 are controlled by two competing mechanisms, the first involved in the chemical reaction as the term of fouling formation and the second involved in the fluid velocity on the fouling removal. This shows when the fluid velocity increases, the fouling deposition decreases while when the wall temperature increases, the fouling formation increases. This threshold fouling model is integrated with the cell-based dynamic heat exchanger model for simulation of heat exchangers.

3.3 Artificial neural network

The ANN model is a technique to imitate the human brain's work and process information [48]. The fundamental concept of ANN is learning from data and making a prediction using the relationships between inputs and outputs data. By comparing the outputs with targets, the system can adjust the weights to match its target. With sufficient number of data, the network can be well trained. ANN model is widely used as the model to solve difficult and complex problems.

3.3.1 Feed-forward neural network

Feed-forward neural networks were the first type of artificial neural networks constructed, and they function more efficiently than other forms of neural networks. They are a widely used type of artificial neural network that incorporate an input layer for receiving external inputs. The weight of each input is compounded. The

input product is then multiplied by the bias and sent to the activation function, which computes and stores the signals in a hidden layer. The signals from the concealed layer are sent to the final output layer. As seen in *Figure 2,* the output layer is in charge of calculating the output signal.
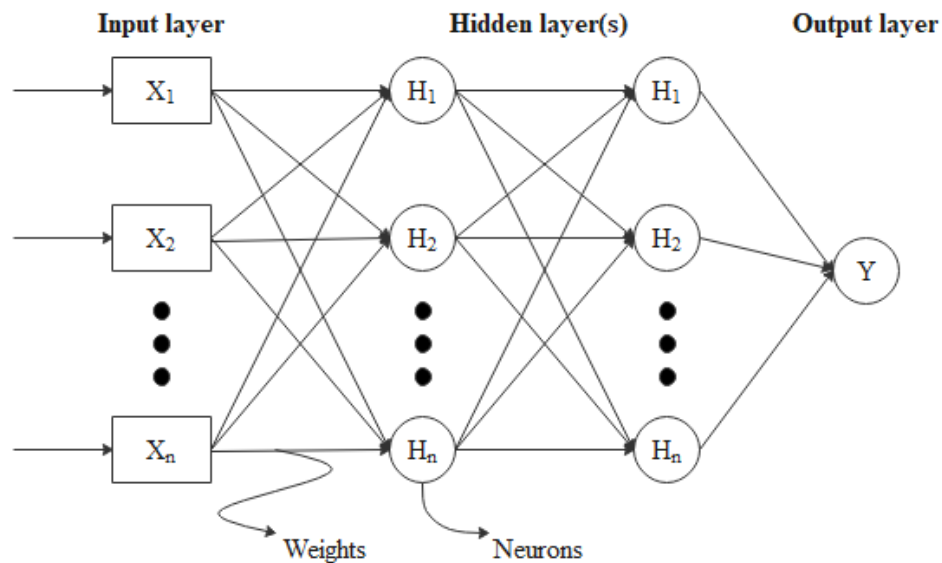
Figure 2. Feed-forward neural network architecture.

The procedure for calculating the output signal (y) for an artificial neuron can be expressed by Eq. 17.

$$y_j = f\left(\sum_{i=1}^{n} w_{ij}x_i + b\right) \tag{17}$$

w_{ij} refers the weight between the input neuron x_i and jth output neuron and b is the bias coefficient of neuron. The net output of each neuron is passed to activation or transfer function ($f(\cdot)$). y_j represents the output signal.

3.3.2 Nonlinear autoregressive artificial neural network with exogenous inputs (NARX-ANN)

The NARX artificial neural network is a recurrent dynamic network with feedback links encompassing various levels of the network, as seen in *Figure 3*. The NARX-ANN

model is derived from the linear ARX model, which is frequently used in time series analysis. The NARX-ANN model's defining equation is as follows:

$$\hat{y}(t) = f\left(y(t), y(t-1), \ldots, y(t-n_y), u(t), u(t-1), \ldots, u(t-n_u)\right) \tag{18}$$

where $y(t)$ is the goal and $\hat{y}(t)$ is predicted output variables. $u(t)$ is the input variable of the network. The time delays of output and input variables are denoted by n_y and n_u. The input is created from time t to $t-n_y$ (output of interesting process), whereas the control value is created from time t to $t-n_u$ (input of interesting process). The dependent output signal $y(t)$ is regressed on its prior values and those of an independent (exogenous) input signal.

The hidden layer output at time t is obtained as [49]

$$H_i(t) = f_1\left(\sum_{r=0}^{n_u} w_{ir} u(t-r) + \sum_{l=1}^{n_y} w_{il} y(t-l) + b_i\right) \tag{19}$$

where w_{ir} denotes the weight of the connection between the input neuron $u(t-r)$ and the ith hidden neuron, and w_{il} denotes the weight of connection between the ith hidden neuron and output feedback neuron $y(t-l)$. b_i and $f_1(\cdot)$ denote the bias of the ith hidden neuron and the activation function of hidden layer, respectively.

The final prediction can be given by [49]

$$\hat{y}_j(t) = f_2\left(\sum_{i=1}^{n_h} w_{ji} H_i(t) + b_j\right) \tag{20}$$

where w_{ji} is the weight associated with the link between the ith hidden neuron and jth predicted output n_h. b_j is the bias of the jth predicted output. n_h is the number of hidden neurons, and $f_2(\cdot)$ is the activation function for the output layer.
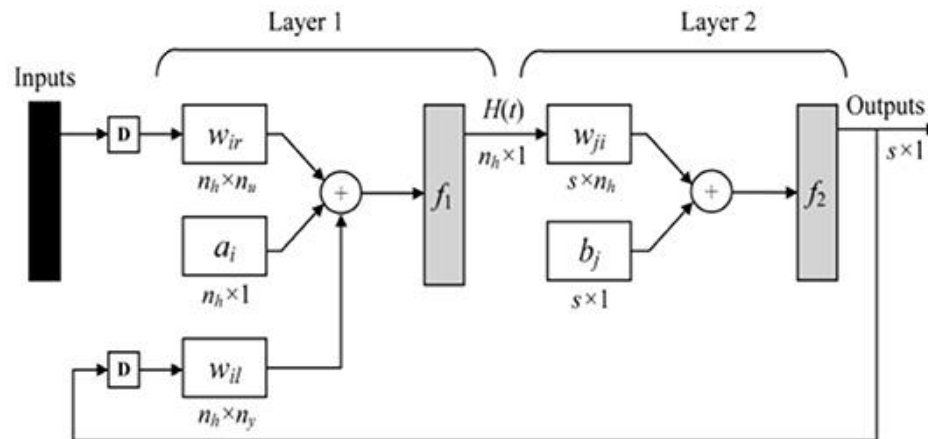
Figure 3. NARX artificial neural network architecture [49].

NARX artificial neural network has a variety of applications. It is helpful as a predictor, predicting the value of the input signal when it is received again. It is an example of nonlinear filtering where the output is noise-free. The NARX-ANN model is demonstrated in another significant application, that of modeling nonlinear dynamic systems. Creating a NARX-ANN model with a series-parallel architecture (NARX-SP network; open-loop) is extremely beneficial for training (see *Figure 4*), as it allows for the use of the real output, the known goal value, rather than feeding back the estimated output in conjunction with exogenous inputs. This offers two benefits. The first advantage is that the feed-forward network's input is more precise. The second advantage is that the resultant network is entirely feed-forward and can be trained using static backpropagation. The NARX-ANN model is closed once the training is complete. The projected output from the series-parallel architecture is linked to the network's earlier point, referred to as the parallel architecture (see *Figure 5*), progressively quenching the prior input and output. Since series-parallel architectures can only anticipate one step ahead, they should be changed to parallel architectures to forecast many time steps.
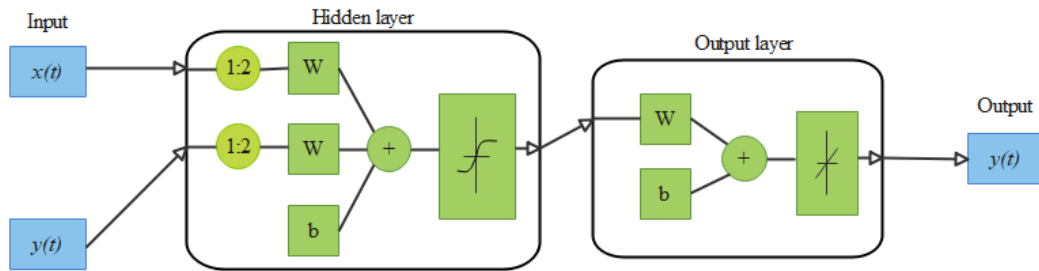
Figure 4. Series-parallel architecture.
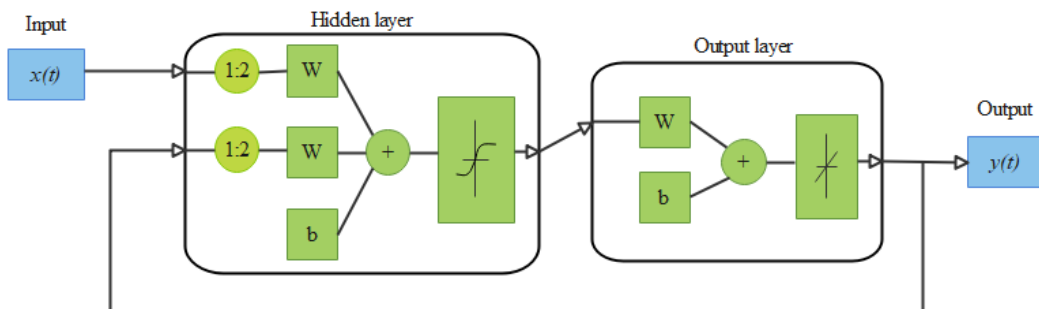
Figure 5. Parallel architecture.

3.3.3 Activation function

Activation functions, also known as transfer functions, are a critical component of neural network architecture because they are utilized to determine the output of nodes in the network layer by converting the sum of all processing from all inputs to an output signal. The activation function used has a considerable influence on the neural network's capacity and efficiency. Technically, the activation function is called after each node in the network has completed its internal processing, even though the network is designed to employ the same activation function on all nodes in the layer. Two types of activation functions exist:

1) Linear function

- The linear activation function, also known as identity, is a simple function because the linear activation function does not change the weighted sum of the inputs in any way and returns a value directly. Both input and output values are in the range $[-\infty, \infty]$. It is often used as an activation function in the output layer.

2) Non-linear function

- The Sigmoid activation function, also known as the logistic function, is a function s curve, commonly used as the activation function in both the hidden and output layers, and is the same function used in the logistic regression classification algorithm. Both input and output values are in the range $[\,0,1\,]$ (see in *Figure 6*).
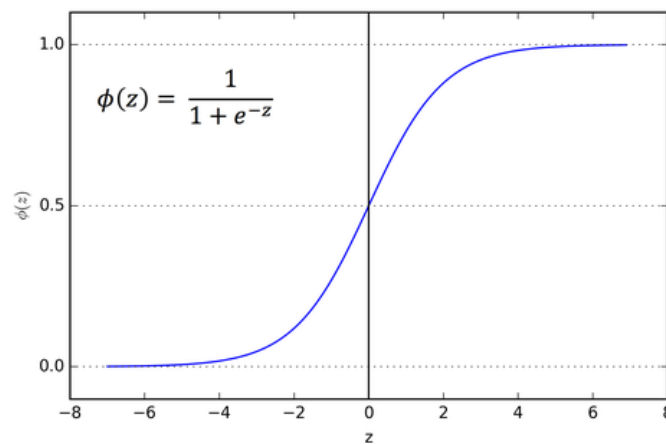
Figure 6. Sigmoid activation function.

- The hyperbolic tangent activation function, also known as tangent sigmoid activation function, is similar to sigmoid activation function and has the same s-shape. Both input and output values are in the range $[-1,1]$. It is commonly used as an activation function in the hidden layer (see in *Figure 7*). Which is mathematically written as follows:

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \qquad\qquad (21)$$

- The rectified linear activation function, or ReLU activation function, is frequently used in hidden layers because of its ease of use and efficiency in network design, which outputs direct input if positive (see in *Figure 8*).
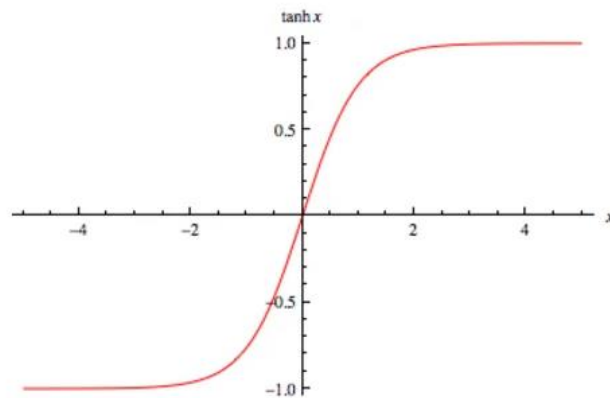
Figure 7. Hyperbolic tangent activation function.
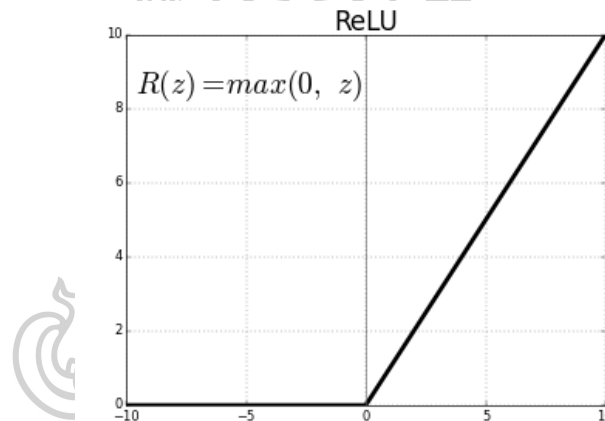
Figure 8. Rectified linear activation function.

- The softmax output activation function is a function that is different from other functions because the probability distribution is calculated where the sum of the results is equal to 1. Thus, the function's input is a vector of actual values, and the output is a vector of equal length with a sum of 1 . The softmax function is calculated as follows:

$$f(x) = \frac{e^x}{\sum e^x}$$

(22)

3.3.4 Training algorithm

The training algorithm is a sequential technique for determining the connections in a neural network. In supervised training, the network is presented with the proper

expected outcome for the training set's input vector. Additionally, it may be required to modify the weight via several repetitions based on training data. In general, an ANN model learns a dataset using backpropagation, which adjusts the weights of neurons depending on the error rate between the goal and anticipated output. Numerous optimization strategies are utilized to train the ANN model, as shown below.

1) Levenberg-Marquardt backpropagation

The Levenberg-Marquardt algorithm sometimes referred to as the damped least-squares technique, is a numerical approach for solving nonlinear function minimization problems. It is quick and steady in its convergence. This approach is well-suited for training neural networks on small and medium-sized issues. The Levenberg-Marquardt algorithm's fundamental principle is to perform a hybrid train process, i.e., around an area of complicated curvature. The Levenberg-Marquardt method shifts to the steepest descent mode when the local curvature is sufficient to provide a quadratic approximation. Then it is transformed into the Gauss-Newton method, which converges substantially faster.

The Levenberg-Marquardt algorithm is optimized for use with a loss function, alternatively referred to as a cost function, which is a sum of squared errors. It operates without calculating the actual Hessian matrix; instead, it uses gradient vectors and the Jacobian matrix. Eq. 23 is a representation of the loss function.

$$f = \sum_{i=1}^{n} e_i{}^2 \tag{23}$$

where n is the number of training sample and e_i is the error between actual and predicted output.

A loss function's Jacobian matrix can be defined as follows:

$$J_{i,j} = \frac{\partial e_i}{\partial w_j} \tag{24}$$

for $i = 1, \dots, m$ and $j = 1, \dots, n$.

where m is the number of training sample and n is the number of parameters in the neural network. Therefore, the size of the Jacobian matrix is $m \times n$.

The gradient vector of the loss function can be calculated as follows.

$$\nabla f = 2J^T \cdot e \tag{25}$$

where e is the vector of all error terms.

Levenberg-Marquardt algorithm introduces another approximation to Hessian matrix:

$$H = 2J^T \cdot J + \lambda I \tag{26}$$

where λ is always positive, called the combination coefficient and I is the identity matrix.

By combining Eq. 25 and 26, the update rule of Levenberg–Marquardt algorithm can be presented as

$$w_{k+1} = w_k - (2J^T \cdot J + \lambda I)^{-1} \cdot (2J^T \cdot e) \tag{27}$$

When the damping parameter λ is zero, this is just Newton's method using the Hessian matrix to estimate the value. Conversely, when λ is large, this becomes gradient descent with a small training rate. The parameter λ is initialized to be large so that the first update is a small step in the direction of the gradient descent. If any iteration results in failure, λ is increased by some factor. Otherwise, as the loss decreases, λ is decreases so that the Levenberg-Marquardt algorithm approaches Newton's method, this process generally accelerates convergence to a minimized the sum of squared errors.

2) Bayesian regularization backpropagation

Bayesian regularization backpropagation is a network training algorithm that utilizes Levenberg-Marquardt optimization to update the weight and bias variables. The learning rule is to continually modify the network weight and bias using the steepest

descent approach via backpropagation (BP) while minimizing the sum of squared errors. Backpropagation with Bayesian regularization consists of two components: the Bayesian approach and the regularization approach [50].

Mackay developed a Bayesian neural network technique based on the Gaussian approximation, which was extensively used afterward. Bayesian networks offer two distinct benefits over traditional neural networks. The first is the training process's automated normalizing of coefficients. The second is to use probabilistic computations to modify the weights of the input variables automatically. The primary concept is to evaluate the model by utilizing current knowledge and data. The distinction between Bayesian and traditional statistics is the use of prior information [51], for which Bayesian formulae are as follows:

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)} \tag{28}$$

The $P(\theta|X)$ is the posterior probability of θ conditional on X, $P(X|\theta)$ is the prior of X conditional on θ, and $P(X)$ is the non-zero prior probability of event X. The Bayesian technique derives the goal function for network training from the probability function of the sample data used as input. Furthermore, the weight modification is determined using the probability distribution of the preceding weights and bias. Thus, using the input sample data, ensuing probabilistic distribution of weights and bias will be gradually adjusted. The parameters of the Bayesian network are named after the probability distribution of the input sample data. Thus, it is theoretically feasible to increase forecast accuracy.

The regularization approach is meant to increase the general ability of neural network training functions by modifying their performance. Normally, the neural network performance training function employs the mean squared error function, indicated by E_D, to describe the neural network model's training set, where $D = (x_i, t_i)$ with $i = 1, 2, \dots, n$ where n is the total number of training samples. The

network error E_D is defined as the error sum of squares under the specified network framework H and starting value of the network parameter W.

$$E_D = \frac{1}{2}\sum_i^n (y(x_i, W, H) - t_i)^2 \tag{29}$$

To avoid overfitting during the network learning process, the standard way is to add the E_W attenuation expression after the error function.

$$E_W = \frac{1}{2}\sum_i^m W_i{}^2 \tag{30}$$

where W is the network parameter and m is the number of parameters. Therefore, all error functions can be defined as follows:

$$M_W = \beta E_D + \alpha E_W \tag{31}$$

where α and β represent the hyperparameter, which is used to control the distribution of other parameters.

The objective function Eq. 31 must be minimized to determine the ideal weight area. As with Eq. 32, this is equal to maximizing the posterior probability function.

$$P(\alpha, \beta | D, M) = \frac{P(D|\alpha, \beta, M)P(\alpha, \beta | M)}{P(D|M)} \tag{32}$$

where D is the weight distribution, M is the particular neural network architecture.

3) Scaled conjugate gradient backpropagation

A scaled conjugate gradient backpropagation (SCG) [52] is a technique for avoiding time-consuming line searches. This method is far too sophisticated to summarize in a few sentences. The core concept is to combine the regional approach of the Levenberg-Marquardt algorithm's trustworthy model with a conjugate gradient technique. Equation 33 [52] illustrates how to estimate the Hessian matrix(s).

$$s_k = \frac{E'(w_k + \sigma_k p_k) - E'(w_k)}{\sigma_k} + \lambda_k p_k \tag{33}$$

where E' is the gradient of E which is the total error function, λ_k is scaling factors, and σ_k are introduced to approximate the Hessian matrix and initialized by user at the beginning of the algorithm such that $0 < \lambda_k < 10^{-6}$ and $0 < \sigma_k < 10^{-4}$.

For SCG, factor calculation (β_k) and direction of the new search are written as in Eq. 34 and 35 [52].

$$\beta_k = \frac{(|g_{k+1}|^2 - g_{k+1}{}^T g_k)}{g_k{}^T g_k} \tag{34}$$

$$p_{k+1} = -g_{k+1} + \beta_k p_k \tag{35}$$

3.4 Proportional-Integral-Derivation (PID) controller

In industrial applications, the PID controller is a commonly used instrument for controlling process variables such as pressure, flow, temperature, and speed. It's simple to grasp and effective. A feedback control system is one of the most prevalent types of control systems. The error value utilized in the computation is the difference between the process variables' actual and desired values. By changing the process input signal, the controller will seek to reduce the error value. The PID controller employs three fundamental control characteristics, which are explained below: proportional, integral, and derivation.

Proportional controllers, or P-controllers, are proportional to the difference between the setpoint and process variable. The distinction is in the definition of the term "erroneous." The proportional gain (K_C) specifies the output response's ratio to the error signal. The downside of this controller is that when used alone, the process variable will never reach the setpoint, i.e., there will always be an offset between the process variable and the setpoint, but the error will always remain stable. By and large, raising the proportional gain speeds up the reaction time of the control system. However, if the proportional gain is set to a value that is too great, the process variable will fluctuate. If K_C is increased more, the oscillation becomes greater, the system becomes unstable, and it may oscillate out of control.

The integral controller, or I-control, continuously analyzes all previous error histories by aggregating the areas under the error curve or summarizing the error terms across time. When a minor mistake occurs, which results in a gradual rise in the I-control, the integral response gradually grows over time. The controller drives the steady-state error signal to zero unless the error approaches zero. When a negative error occurs, the I-control typically reduces the output. Reduce the integral gain (K_I) to increase the response speed.

The derivative controller, abbreviated D-control, is capable of forecasting future error behavior. Its output is proportional to the rate at which the time error changes multiplied by the derivative constant. By increasing the derivative parameter (T_D), the control system will be able to respond more harshly to changes in the error range and will also enhance the overall control system's reaction speed. Because the derivative response is very susceptible to noise in the process variation signal, the majority of practical control systems employ a modest amount of derivative time (T_D). If the sensor feedback signal is noisy or the control loop rate is too slow, the response derivative may become unstable, causing the control system to fail.

A standard PID controller, commonly used in control, is shown by Eq. 36.

$$u(t) = u_0 + K_C\left[e(t) + \frac{1}{T_I}\int_0^t e(t)dt + T_D\frac{de(t)}{dt}\right] \tag{36}$$

Transfer function of the PID controller is given by:

$$G(s) = K_C\left(1 + \frac{1}{T_I s} + T_d s\right) = K_C + K_I\frac{1}{s} + K_D s \tag{37}$$

where K_C = proportional gain, K_I = integral gain, K_D = derivative gain, T_I = integral time constant, and T_D = derivative time constant.

3.4.1 Tuning method

Before the PID controller, the controlled process's dynamics had to be adjusted, since the default P, I, and D values did not offer the necessary performance and

occasionally resulted in control instability and delayed response. Numerous tuning techniques need to be created to determine the optimal PID controller tuning, requiring operator attention to pick the optimal proportional, integral, and derivative gains.

- The Zeigler-Nichols method is a widely used tuning technique. The PID controller works similarly to the trial and error technique, with I and D set to zero and P increasing incrementally until the loop begins to oscillate. When the oscillation begins, the final gain K_U and the oscillation time, denoted by T_U, are recorded. P, I, and D are then modified in accordance with the table below.

Table 2. Zeigler-Nichols table.

Control type	K_P	T_I	T_D	$K_I = K_P/T_I$	$K_D = T_D/K_P$
P	$0.5K_U$	-	-	-	-
PI	$0.45K_U$	$T_U/1.2$	-	$0.54K_U/T_U$	-
PD	$0.8K_U$	-	$T_U/8$	-	$0.1K_UT_U$
PID (classic)	$0.6K_U$	$T_U/2$	$T_U/8$	$1.2K_U/T_U$	$0.075K_UT_U$

- The internal model control (IMC) can be utilized to determine the optimal tuning value for the PID controller, as seen in *Table 3*. τ_c is a crucial decision parameter when the IMC technique is used to construct PID controllers. In general, when K_C drops and T_I grows, the inclusion of τ_c results in a more conservative controller.

Numerous IMC recommendations have been given for choosing τ_c in the first order plus time delay (FOPTD) model.

1) $\tau_c/\theta > 0.8$ and $\tau_c > 0.1\tau$ (Rivera et al., 1986 [53])

2) $\tau > \tau_c > \theta$ (Chien and Fruehauf, 1990 [54])

3) $\tau_c = \theta$ (Skogestad, 2003 [55])

where τ is time constant of a process and θ is time delay.

Table 3. IMC-based PID controller settings.

Case	Model	$K_C K$	T_I	T_D
A	$\dfrac{K}{\tau s + 1}$	$\dfrac{\tau}{\tau_c}$	τ	-
B	$\dfrac{K}{(\tau_1 s + 1)(\tau_2 s + 1)}$	$\dfrac{\tau_1 + \tau_2}{\tau_c}$	$\tau_1 + \tau_2$	$\dfrac{\tau_1 \tau_2}{\tau_1 + \tau_2}$
C	$\dfrac{K}{\tau^2 s^2 + 2\xi\tau s + 1}$	$\dfrac{2\xi\tau}{\tau_c}$	$2\xi\tau$	$\dfrac{\tau}{2\xi}$
D	$\dfrac{K(-\beta s + 1)}{\tau^2 s^2 + 2\xi\tau s + 1}, \beta > 0$	$\dfrac{2\xi\tau}{\tau_c + \beta}$	$2\xi\tau$	$\dfrac{\tau}{2\xi}$
E	$\dfrac{K}{s}$	$\dfrac{2}{\tau_c}$	$2\tau_c$	-
F	$\dfrac{K}{s(\tau s + 1)}$	$\dfrac{2\tau_c + \tau}{\tau_c{}^2}$	$2\tau_c + \tau$	$\dfrac{2\tau_c \tau}{2\tau_c + \tau}$
G	$\dfrac{K e^{-\theta s}}{\tau s + 1}$	$\dfrac{\tau}{\tau_c + \theta}$	τ	-
H	$\dfrac{K e^{-\theta s}}{\tau s + 1}$	$\dfrac{\tau + \dfrac{\theta}{2}}{\tau_c + \dfrac{\theta}{2}}$	$\tau + \dfrac{\theta}{2}$	$\dfrac{\tau\theta}{2\tau + \theta}$
I	$\dfrac{K(\tau_3 s + 1)e^{-\theta s}}{(\tau_1 s + 1)(\tau_2 s + 1)}$	$\dfrac{\tau_1 + \tau_2 - \tau_3}{\tau_c + \theta}$	$\tau_1 + \tau_2 - \tau_3$	$\dfrac{\tau_1 \tau_2 - (\tau_1 + \tau_2 - \tau_3)}{\tau_1 + \tau_2 - \tau_3}$

3.5 Model predictive control

Model predictive control is a type of control that utilizes dynamic model data from the system to forecast future reactions and select the optimal control signal to feed the system. The calculation finds the best value of the target function using the future forecast value and the measurement value at the present moment of the calculation. MPC's objective function is denoted by Eq. 38.

$$J = \sum_{j=N_{min}}^{N_{max}} \left(y_r(k+j) - \hat{y}(k+j)\right)^2 + \rho \sum_{j=1}^{N_u} (u(k+j-1) - u(k+j-2))^2 \qquad (38)$$

where N_{min}, N_{max} are minimum and maximum prediction horizons, N_u is the control horizon. The u variable is the control signal, y_r is the desired response and $\hat{y}$ is the network model response. The ρ value determines the contribution that the sum of the squares of the control increments has on the performance index. The process constraints on the manipulated and controlled variables as follows:

$$u_{min} \le u(k+j-1) \le u_{max}, \quad j = 1, 2, \ldots, N_u$$

The length of the control horizon N_u must satisfy following constraints:

$$0 < N_u \le N_{u_{max}}$$

The value of $N_{u_{max}}$ should cover the important part of the step response curve.

Output variable constraint is given as:

$$y_{min} \le \hat{y}(k+j) \le y_{max}, \quad j = N_{min}, \ldots, N_{max}$$
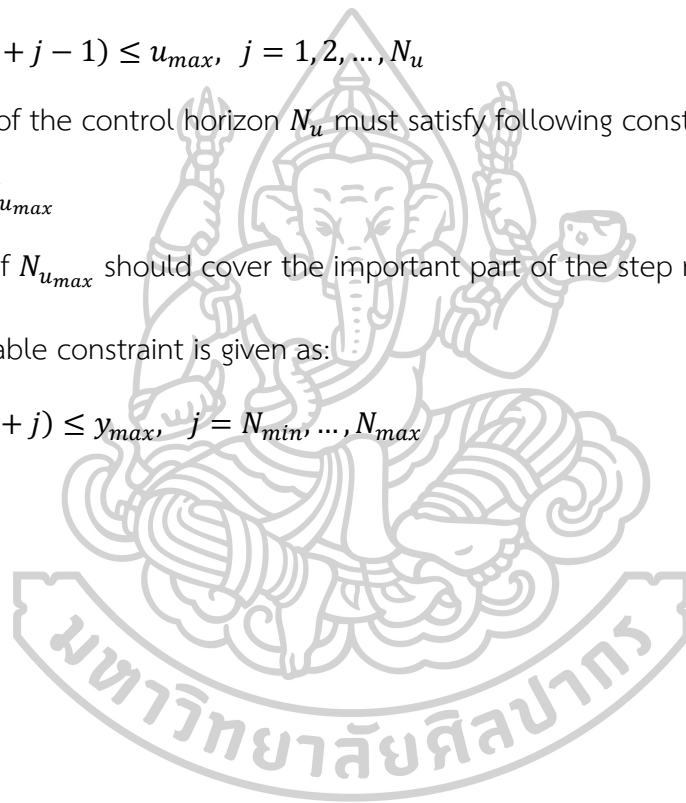
CHAPTER IV

RESEARCH METHODOLOGY

4.1 Equipment and software

4.1.1 Laptop (ASUS Intel® Core™ i5-6200U CPU @ 2.30GHz 2.40 GHz, 12.00 GB)

4.1.2 MATLAB

4.2 Heat exchanger model

4.2.1 Dynamic model of heat exchangers

In simulation of shell-and-tube heat exchangers, the concept of cell-based dynamic models [17] was used. In the cell-based model as shown in *Figure 9*, a heat exchanger is assumed to composed of a series of mixing tanks where in each tank, heat is exchanged between hot and cold streams. Under the following assumptions: 1) physical properties including density and heat capacity of fluid are constant, 2) all tank cells are perfectly mixed and full with the corresponding fluids, and 3) heat loss to the environment is neglected, cell-based model of heat exchangers can be written as the equations 39-41 with process parameters in Table 4, which these parameters are calculated from [56].
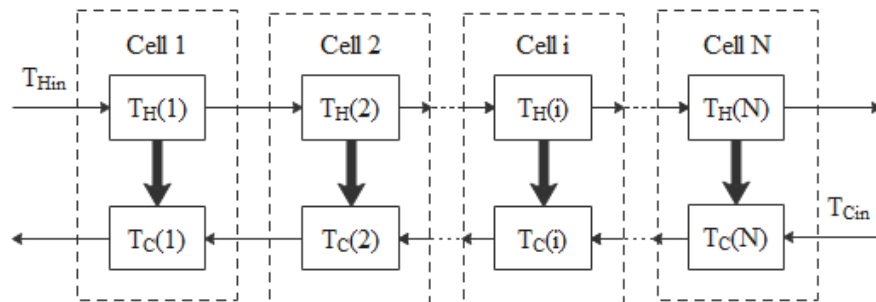
Figure 9. Cell-based model of heat exchangers.

$$\frac{dT_H(i)}{dt} = \frac{v_H}{V_{H,Cell}} \cdot \left(T_H(i-1) - T_H(i)\right) - \frac{h_{fH} \cdot A_{Cell}}{V_{H,Cell} \cdot \rho_H \cdot C_{p,H}} \cdot \left(T_H(i) - T_W(i)\right) \qquad (39)$$

$$\frac{dT_C(i)}{dt} = \frac{v_C}{V_{C,Cell}} \cdot \left(T_C(i+1) - T_C(i)\right) + \frac{h_{fC} \cdot A_{Cell}}{V_{C,Cell} \cdot \rho_C \cdot C_{p,C}} \cdot \left(T_W(i) - T_C(i)\right) \qquad (40)$$

$$\frac{dT_W(i)}{dt} = \frac{h_{fH} \cdot A_{Cell}}{V_W \cdot \rho_W \cdot C_{p,W}} \cdot \left(T_H(i) - T_W(i)\right) - \frac{h_{fC} \cdot A_{Cell}}{V_W \cdot \rho_W \cdot C_{p,W}} \cdot \left(T_W(i) - T_C(i)\right) \qquad (41)$$

For determining the influence of fouling on the heat transfer of a heat exchanger system, the heat transfer coefficient was calculated using a formula including the thermal resistance of the fouling layer in cold and hot streams as follows:

$$h_{fH} = \frac{h_H}{1 + h_H \cdot R_{fH}}$$ (42)

$$h_{fc} = \frac{h_c}{1 + h_c \cdot R_{fc}}$$ (43)

Table 4. The process parameters of the cell-based dynamic model for shell-and-tube heat exchanger.

Process parameters	Hot stream	Cold stream
	Kerosene (42° API)	Crude oil (34° API)
Volumetric flow rate (m³/h)	20	35
Inlet temperature (°C)	150	53
Density (kg/m³)	730	820
Specific heat capacity (J/kg K)	2470	2050
Viscosity (Pa·s)	0.00043	0.0032
Volume (m³)	0.0568	0.0777
Heat transfer coefficient, clean (J/m² K)	940	680
Heat transfer surface area (m²)	26.93	

4.2.2 Threshold fouling model

In this work, the threshold fouling model developed by [5] as shown in the equation 44 was used for predicting fouling rates.

$$\frac{dR_f}{dt} = \alpha Re^\beta exp\left(\frac{-E}{RT_f}\right) - \gamma Re^{0.4}$$ (44)

where α=0.01098 m²K/J

β=-1.547

γ=0.96×10⁻¹³ m²K/J

E=22.618 kJ/mol

These constants were calculated from experimental data of the Australian crude oil measured [47]. Film temperature (T_f) is described by [4],

$$T_f = T_b + 0.55(T_w - T_b) \tag{45}$$

where T_b and T_w are the bulk temperature and the wall temperature, respectively.

The fouling model in the equation 44 are controlled by two competing mechanisms, the first involved in the chemical reaction as the term of fouling formation and the second involved in the fluid velocity on the fouling removal. This shows when the fluid velocity increases, the fouling deposition decreases while when the wall temperature increases, the fouling formation increases. This threshold fouling model is integrated with the cell-based dynamic heat exchanger model for simulation of shell-and-tube heat exchangers. The shell and tube sides represent the hot and cold streams, respectively.

4.3 Modeling with ANN

In this research, the ANN model based on NARX artificial neural network was used to predict and control the dynamics of the heat exchanger under parameter uncertainty. The NARX artificial neural network is a recurrent dynamic network with reverse connections surrounding various layers of the network. The prediction of the independent output signal is regressed to the previous value of the output signal and the previous value of the exogenous input signal. In the architecture of the NARX-ANN model, a series-parallel architecture (open-loop architecture) was used to train the model by relying on the target output to estimate the output signal along with the exogenous input. And once the train is successful, the architecture is closed, i.e., the output signal from the series-parallel architecture is reconnected back to the network's earlier point or the network's feedback input to enable the NARX-ANN model to predict multiple steps. This step is called parallel architecture (closed-loop architecture) and is then used in the next step of NNMPC. In this work, the activation

function used in the hidden layer is the sigmoid tangent function, while the output layer is the linear transfer function, which is the default in MATLAB software.

Due to dynamic neural networks are delayed. The input in the network must be a sequence of input vectors occurring in a certain order of time. The order in which the vector appears is critical to the approximation of the output signal, as the appropriate number of delays result in an accurate NARX-ANN model. In addition, the number of hidden neurons is significant for prediction as well. Although increasing the number of hidden neurons will result in a more accurate NARX-ANN model, it takes a lot of processing time and results in the model becoming overly complex due to a large number of parameters and its ineffectiveness. This event is called overfitting. To prevent this problem, finding the optimal number of delays and the optimal number of hidden neurons would allow the model to be effectively applied to previously unseen data.

In this work, trial-and-error was used as a method to study the effect of hyperparameters, i.e., the number of delays and the number of hidden neurons, wherein the delay numbers varied from 0-4 for both the exogenous input and the feedback input, and the number of hidden neurons varied from one to twenty. Moreover, in the ANN training process, an optimization algorithm is required. Several algorithms, Bayesian regulation backpropagation, Levenberg-Marquardt backpropagation, and Scaled conjugate gradient backpropagation, were tested. Determination of the number of delays, the number of hidden neurons, and the training algorithm are decided to take into account the effect of a statistical index (such as MSE).

4.3.1 Data generation

Data are important for ANN modeling. In this research, the data for training and testing NARX-ANN model were collected from the simulation of cell-based heat exchangers integrated with the threshold fouling model as mentioned in section 2.

Under the simulation runs with random of the inlet temperatures (140-160 $^{\circ}$C and 40-60 $^{\circ}$C for hot and cold streams) and the flow rates (0-30 m^3/h and 30-40 m^3/h for hot and cold streams), changes of outlet temperatures of hot and cold streams, and fouling factors of hot and cold sides along the time were observed. The data used for training is generated in four sets. Each set took the initial values of the fouling occurrences of each interval: clean, 1-month, 2-month, and 3-month intervals. Each interval of data took 4 hours for running. After the data were generated, the inlet temperatures and the flow rates were used as exogenous input of NARX-ANN model while outlet temperatures and fouling factors was used as the output. Therefore, the NARX-ANN model used to predict and control the dynamics of shell-and-tube heat exchangers under parameter uncertainty in the design of NNMPC has four models.

4.3.2 Training

The NARX-ANN model is used to predict future values based on previous inputs. Training is an iterative process that attempts to adjust the weights and bias of the ANN model based on optimization algorithms to reduce errors between target and actual data. This is the process of learning data of the model.

The data generated in section 4.3.1 is divided into three sections: 70% of training, 15% of validation, and 15% of testing datasets before starting to train the NARX-ANN model. During the training process, the NARX-ANN model starts with random values for weight and bias coefficients, and the training algorithm attempts to adjust them using the experimental data provided. The training algorithm adjusts these parameters by minimizing the deviation between the experimental value and the predicted output value. The general structure of the NARX-ANN model training is shown in *Figure 10*, where $u(t)$ is the exogenous input, $y(t)$ is the measure value or the plant model, $\hat{y}(t+1)$ is the predicted output or NN model output, the blocks TDL is the tapped delay lines that hold the past value of the input signal, $IW^{i,j}$ is weight matrix from jth input neuron to ith hidden layer and $LW^{i,j}$ is weight matrix from jth output neuron to ith hidden layer. The efficacy of the trained NARX-ANN

model should be verified by experimental datasets such as test subsets that were not used during the training process. The MSE was used to measure the efficiency of the NARX-ANN model since there were four output variables in this study. Therefore, before measuring the performance of the model, it is necessary to all normalization output variables first and then calculates the statistic index. The steps described above can only predict one step. A multistep ahead prediction can be created by reconnecting the output signal back to the network's feedback input of the network.
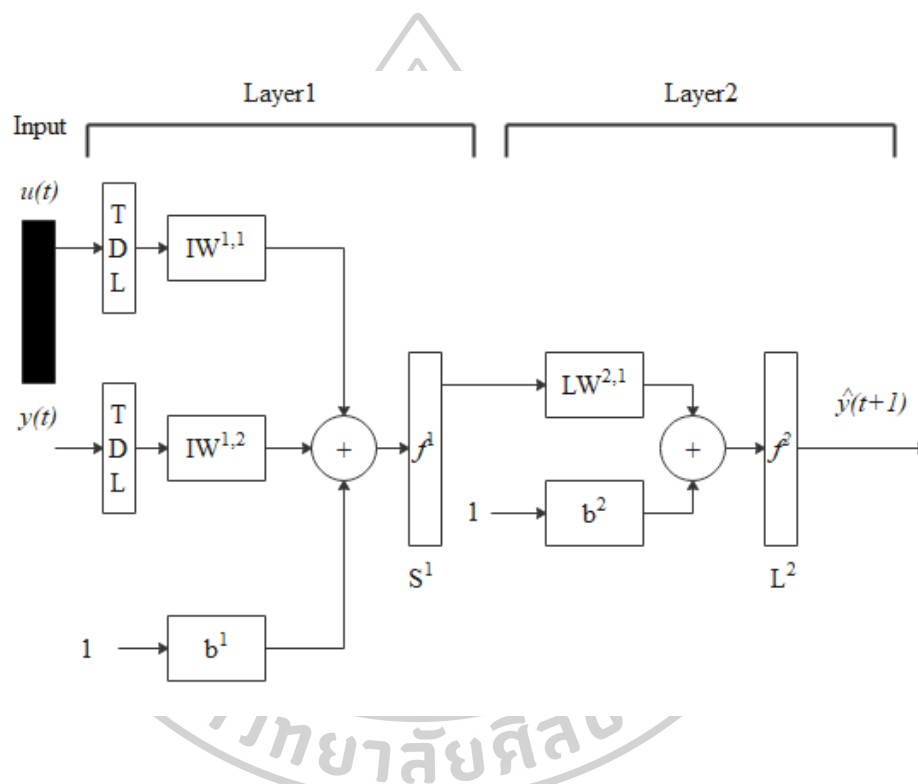
Figure 10. NARX-ANN model training structure.

4.4 Study the effect of fouling on the dynamics of PID-controlled heat exchangers

PID control is a control algorithm commonly used in the industry and internationally recognized in industrial control. The popularity of PID controllers is partly due to their robust performance in a wide range of operating conditions. Another is due in part to the simplicity of use compared to other control algorithms, which allows engineers to operate straightforwardly. There is also no need for an accurate

mathematical model. Therefore, PID controllers are often used first in the design of control systems in various systems. However, the PID controller has to be properly tuned to achieve a good performance.

In the controlled PID of shell-and-tube heat exchangers, outlet temperature at cold stream (T_C) is controlled variable, where the manipulated variable is flow rate at hot stream (v_H) and disturbance variables are flow rate for cold stream (v_C) and inlet temperatures of hot and cold streams (T_{Hin}, T_{Cin})

4.4.1 Identifying the system from the step response data

The First-order plus time delay (FOPTD) model is a simple approximation of the dynamic response (transient response or time response) of a process variable to influence, which is a reasonable approximation of the behavior of the process. It is often widely used in algorithm tuning for PID controllers to obtain default controller tuning constants. The FOPTD can be estimated using the unit step input and uses 2 points in the high change rate region to force the model response to match the actual response. The general form of FOPTD is shown as follows.

$$G_p(s) = \frac{Ke^{-\theta s}}{\tau s + 1} \tag{46}$$

The two points is a simple and convenient method for adjusting the dynamic response of the system of the FOPTD transfer function. Parameter modeling, gain (K), time constant (τ), and time delay (θ), was obtained for Eq. 46 by calculating time constants t_1 and t_2, the response to 28.3% and 63.2% of the difference between the first steady-state value and the last steady-state value. It is shown in Eq. 47-49 and *Figure 11.*

$$K = \frac{\Delta y}{\Delta u} \tag{47}$$

$$\tau = 3/2 \, (t_2 - t_1) \tag{48}$$

$$\theta = t_2 - \tau \tag{49}$$

4.4.2 Tuning the PID controller

In this work, the IMC-based PID controller (see *Table 3*) was utilized for FOPTD systems to determine the optimal tuning value for the PID controller, including K_C, T_I, and T_D, of shell-and-tube heat exchanger under parametric uncertainty. The desired performance with IMC–PID is achieved using only the tuning parameter, τ_c, which is related to the time constant of the process. From *Table 3*, the tuning value of the PID controller can be calculated as follows:

$$K_C = \frac{2\tau + \theta}{K(2\tau_c + \theta)}, T_I = \tau + \frac{\theta}{2}, T_D = \frac{\theta\tau}{2\tau + \theta} \tag{50}$$

Note that τ_c is closed-loop time constant.
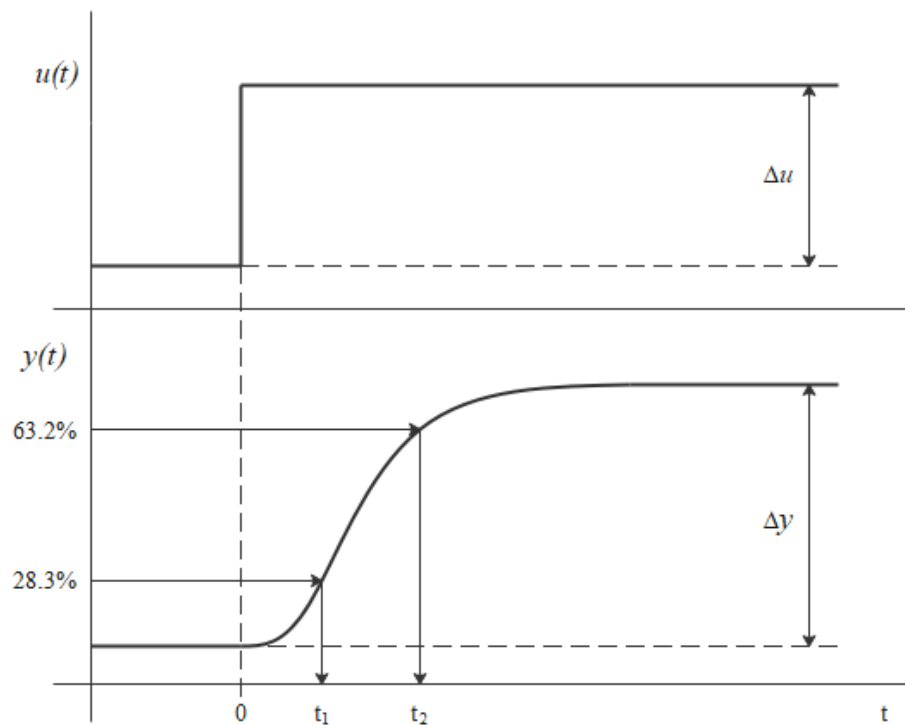
Figure 11. Step response of open loop process.

4.5 Study the effect of fouling on the dynamics of NNMPC heat exchangers

The neural network-based model predictive control uses a non-linear plant neural network model to predict plant performance. The controller then calculates the control inputs that will optimize the plant's performance at a specific time in the

future. The first step in controlling model predictions is to define a neural network plant model. Next, the controller uses a factory model to predict future performance. These steps are described below.

4.5.1 System identification

The identification of the NNMPC system is to train the neural network to represent the plant's feed-forward dynamics. In this study, the network is trained in offline batch mode using data collected from plant operations. The neural network model uses previous input and previous plant output to predict future plant output values. The prediction error between the plant output and the neural network model output is used as a signal to train the neural network. In other words, the prediction error is sent to the training algorithm to adjust parameters, weight, and bias coefficient, with minimal deviations between the experimental value and the predicted output, making the neural network model a more accurate, controllable heat exchanger for effective results. This procedure can be performed as outlined in section 4.3.

4.5.2 Neural network-based model predictive control

The neural network-based model predictive control is used to control the outlet temperature of shell-and-tube heat exchangers under fouling occurrence. The NNMPC consists of a neural network model (NARX artificial neural network model) and numerical optimization software. A neural network model was used to predict plant responses over a specific time. The output futures predicted by the neural network model are then sent to the numerical optimization software to determine the control signal (u) that minimizes the cost function (J) over the specified horizon. MPC's cost function can be calculated from equation 38. The general control structure for the NNMPC is shown in *Figure 12*.
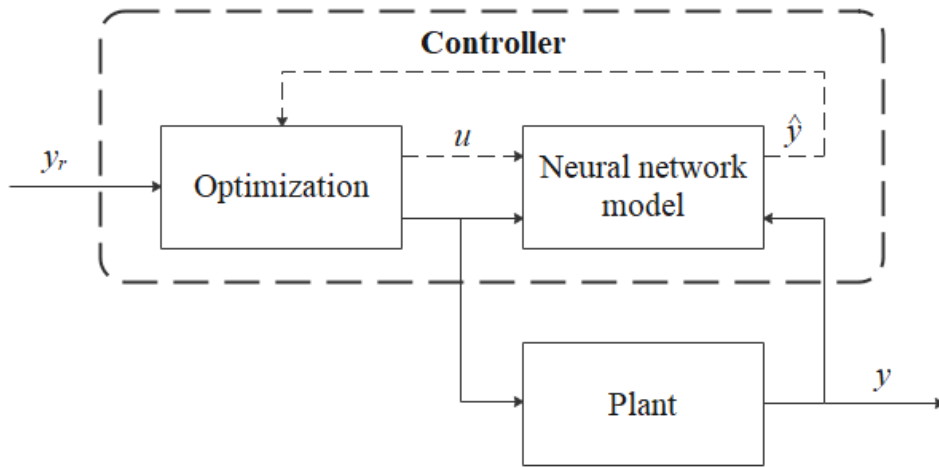
Figure 12. Neural network-based model predictive control.

CHAPTER V

RESULTS AND DISCUSSION

5.1 Open-loop of the heat exchanger with fouling

Cell-based dynamic model of heat exchangers [20] with the threshold fouling model was used for simulation of heat exchanger. In fining the optimal number of cells, the number of cells was increased until there was no significant change in the outlet tempeature of heat exchangers. Based on the result, the 30 cell-model of heat exchangers was chosen and was used to study the behavior of changes in the output temperature of the cold stream from its steady-state value, ΔT_{Cout}.

From *Figure 13a*, it was shown that after the step changes 5% of the flow rate at the hot stream (from 20 to 21 m³/h) was introduced, ΔT_{Cout} increased to 0.5153 °C (from 85.28 °C reach 85.79 °C) with response time to a steady-state at 89.64 seconds for the clean heat exchanger. As the fouling layer increases, the thermal conductivity of the metal surface decreases, and the heat transfer resistance increases, resulting in the output temperature at the cold stream being unable to maintain the same temperature as during the absence of fouling formation. It can be seen that after operating 1, 2 and 3 months of the heat exchanger, ΔT_{Cout} increased 0.4864 °C (from 84.44 °C reach 84.92 °C), 0.4597 °C (from 83.63 °C reach 84.09 °C), and 0.4352 °C (from 82.86 °C to 83.3 °C), respectively.

Reducing the flow rate of hot fluids by 5% (20 to 19 m³/h), ΔT_{Cout} reduced to 0.558 °C (from 85.28 °C to 84.72 °C) for the clean heat exchanger. And after the heat exchanger operated for 1, 2, and 3 months, ΔT_{Cout} is reduced to 0.527 °C (from 84.44 °C into 83.91 °C), 0.4983 °C (from 83.63 °C reach 83.13 °C) and 0.4719 °C (from 82.86 °C to 82.39 °C), respectively, as shown in *Figure 13b*.

When increasing 5% of inlet temperature of the hot stream (from 150 °C to 157.5 °C), ΔT_{Cout} over the clean heat exchanger increased by 2.496 °C (from 85.28 °C to 87.77 °C), with response time to steady-state at 95.44 seconds. After operating the

heat exchanger for 1, 2 and 3 months continuously, ΔT_{Cout} increased 2.431 ℃ (from 84.44 ℃ to 86.87 ℃), 2.368 ℃ (from 83.63 ℃ to 86 ℃) and 2.309 ℃ (from 82.86 ℃ to 85.17 ℃), respectively (see *Figure 14a*). Reduction 5% of the inlet temperature from 150 ℃ to 142.5 ℃ showed that ΔT_{Cout} over the clean heat exchanger reduced to 2.496 ℃ (from 85.28 ℃ reach 87.77℃). The heat exchanger operated over 1, 2, and 3 months, ΔT_{Cout} decreased 2.431 ℃ (from 84.44 ℃ reach 82.01 ℃), 2.368 ℃ (from 83.63 ℃ to 81.26 ℃), and 2.309 ℃ (from 82.86 ℃ reach to 80.55 ℃), respectively, as shown in *Figure 14b*.
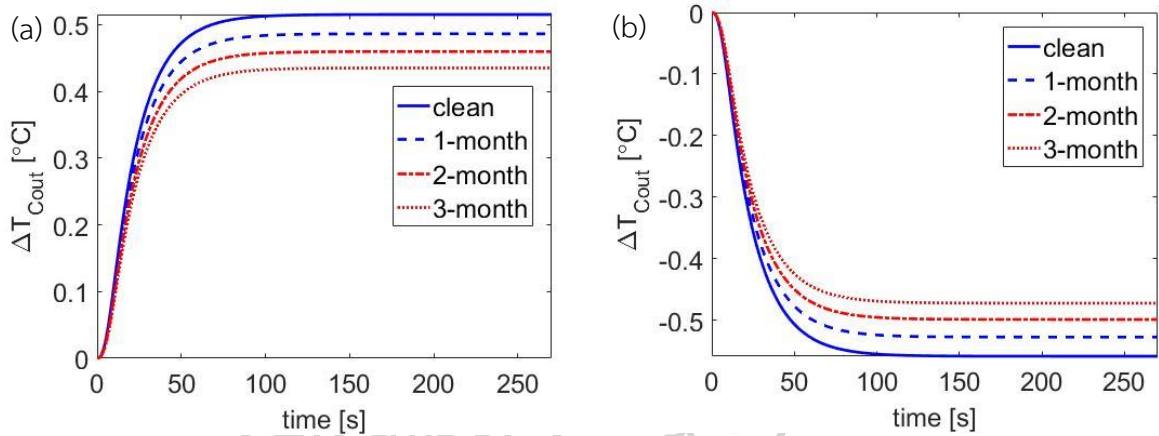
Figure 13. Open-loop responses of the heat exchanger with fouling to step change (a) +5% of flow rate of the hot stream and (b) -5% of flow rate of the hot stream.
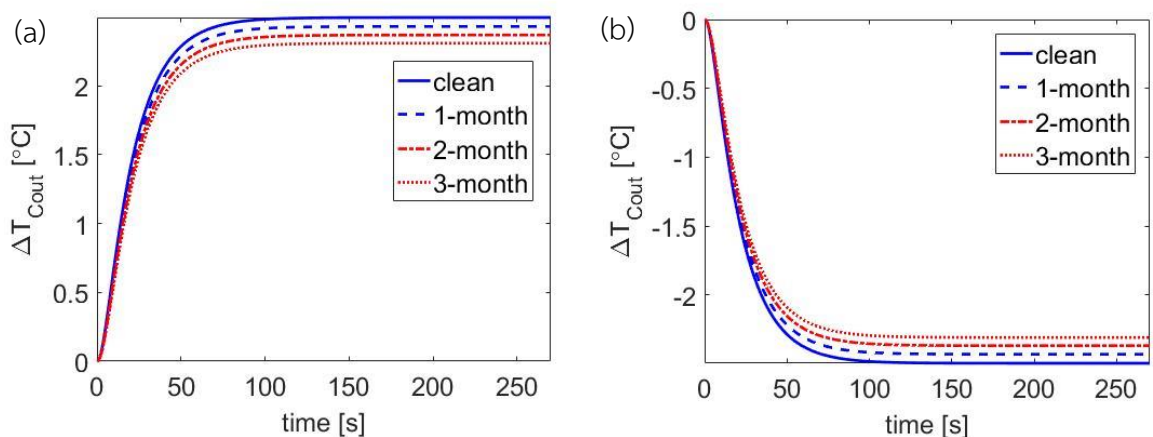
Figure 14. Open-loop responses of the heat exchanger with fouling to step change ± 5% of inlet temperature at hot stream.

It can be seen that an increase of 5% of the flow rate and the inlet temperature of the hot stream, the outlet temperature of cold stream response tends to increase accordingly. Meanwhile, when the flow rate and inlet temperature of the hot stream are reduced by 5%, the outlet temperature of the cold stream response tends to decrease as well. But when the heat exchanger was operated for a while, it was found that the formation of fouling resulted in a decrease in the efficiency of the heat exchanger. It can be noted that the outlet temperature on the cold side could not be maintained the same temperature as the clean heat exchanger. Hence, temperature control was needed.

5.2 Topology of the NARX-ANN model

As the fouling changes along the operation or time, NARX-ANN model was chosen in design of ANN model for fouling estimation. Designing an NARX-ANN model requires finding the appropriate number of delays, number of hidden layers, and the number of neurons in each hidden layer to enable the model to predict values accurately. [57] proposed that feed-forward networks with a single hidden layer can approximate continuous functions. Therefore, this research selected one hidden layer to minimize the model complexity. In finding the optimum number of delays and the number of hidden neurons in the hidden layer, various number of delays and number of hidden neurons were tested and ranked using the statistical accuracy indices including mean square error (MSE) and regression coefficient (R^2-value) as shown in the equations 51 and 52, respectively.

$$mse = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{51}$$

$$R^2 = \frac{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2 - \sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2} \tag{52}$$

where N is the number of data points, y_i and $\hat{y}_i$ represent the actual values and the predicted values of fouling factor, respectively. $\bar{y}_i$ is the average value of actual values of fouling factor.

5.2.1 Number of delays

In general, the trial and error method is widely used to quantify the optimal number of delays for constructing the NARX-ANN model because it is simple and easy. In the process of figuring out the proper number of delays, a three-step method were proposed; firstly, the number of delays for both the past exogenous input (n_u) and the past actual output (n_y) were varied from 0-4; secondly, varying structures were trained and tested, and then calculate the statistical accuracy indices; and thirdly, the optimal number of delays was selected from the smallest statistic accuracy indices. All procedures were performed under ten hidden neurons and Levenberg-Marquardt backpropagation as a training algorithm. In this study, four NARX-ANN models were trained for four operation periods (clean, after 1-, after 2-, and after 3-month operation) of heat exchangers. Determining the best topology of the NARX-ANN models by a different number of delays was shown in Appendix A. The results of the NARX-ANN models with the optimal number of delays under fouling formation in the heat exchanger were summarized in *Table 5*.

Table 5. The NARX-ANN models with the optimal number of delays under fouling formation.

Fouling	The number of delays	Train		Test	
	(n_u,n_y)	MSE	R^2	MSE	R^2
clean	3,1	0.00000451	0.99999	0.00000756	0.99999
1 month	2,1	0.00000726	0.99999	0.00000680	0.99999
2 months	2,1	0.00000778	0.99999	0.00000649	0.99997
3 months	2,1	0.00000807	0.99999	0.00000948	0.99999

5.2.2 Number of hidden neurons

A three-step method was proposed to find the optimal number of hidden neurons; firstly, the number of hidden neurons was varied from one to twenty; secondly, varying structures were trained and tested, and then calculate the statistical accuracy

indices; and thirdly, the optimal number of hidden neurons was selected from the smallest number of hidden neurons with an acceptable statistic accuracy indices. Determining the best architecture of the NARX-ANN models with a different number of hidden neurons was shown in Appendix B. Although increasing the number of hidden neurons resulted in a more accurate model, it took a lot of processing time and resulted in too complex models that could not be implemented in real life. *Table 6* summarized the results of the NARX-ANN models with the optimal number of hidden neurons under fouling formation in the heat exchanger. The findings were carried out under the topology, number of delays, as given in section 5.2.1. The training algorithm used was Levenberg-Marquardt backpropagation.

Table 6. The NARX-ANN models with the optimal number of hidden neurons in various fouling.

Fouling	The number of hidden neurons	Train		Test	
		MSE	R^2	MSE	R^2
clean	10	0.00000451	0.99999	0.00000756	0.99999
1 month	15	0.00000452	0.99999	0.00000514	0.99999
2 months	15	0.00000398	0.99999	0.00000433	0.99999
3 months	15	0.00000548	0.99999	0.00000788	0.99999

5.2.3 Training algorithm

Finding the best training algorithm to be used in the training process to create the NARX-ANN models was tested in this section. The steps are as follows: 1) define the training algorithm Bayesian regulation backpropagation, Levenberg-Marquardt backpropagation, and Scaled conjugate gradient backpropagation, 2) various training algorithm was trained and tested, and then calculate the statistical accuracy indices as the accuracy and computational time should be traded-off, the indices used for the comparison of algorithm performance includes MSE, R^2, and processing time, and 3) the best training algorithm was selected from the smallest statistic accuracy indices and low processing times. Determining the best architecture of the NARX-ANN

models with a different training algorithm as shown in Appendix C. The best training algorithm of the NARX-ANN models under fouling formation in the heat exchanger was shown in *Table 7*. The finding is performed under the topology, i.e., the number of delays and the number of hidden neurons, according to getting in section 5.2.1 and section 5.2.2, respectively.

Table 7. The best training algorithm of the NARX-ANN models under fouling formation.

Fouling	The training algorithm	MSE	R^2	Processing time (s/epoch)
clean	Bayesian regulation backpropagation	0.00000237	0.99999	0.12515
1 month	Bayesian regulation backpropagation	0.00000392	0.99999	0.21800
2 months	Bayesian regulation backpropagation	0.00000307	0.99999	0.17946
3 months	Bayesian regulation backpropagation	0.00000380	0.99999	0.19101

5.3 Temperature control of heat exchangers using PID controller

In temperature control of heat exchanger here, the output temperature of the cold stream and the flow rate of hot stream were considered as controlled variable and manipulated variable, respectively. Four PID controllers were designed corresponding with a FOPTD transfer-function model specified from four period operation (clean, after 1-, after 2-, and after 3-month operation). The FOPTD form of heat exchangers was derived from the approximation of the response of the outlet temperature of the hot side under a step-change of the input (flow rate of the cold stream). The two-point method of [58] that use the times that the outputs reach 28.3% and 63.2% of the final value was used for fitting of the FOPTD models.

The values of the tuning parameters of the PID controllers for each period of operation were determined using the IMC method. The results were shown in *Table 8.* Periodic revision of the PID controller was required to compensate for deposits on the heat transfer surface to allow satisfactory of the output temperature.

Table 8. PID controller parameters designed by the internal model control method.

Heat exchanger fouling	K_C	K_I	K_D
clean	2.7211	468.7481	0.0019
1 month	3.3081	564.0387	0.0024
2 months	3.4868	588.9887	0.0026
3 months	3.6638	611.6577	0.0028

PID controller parameters designed according to *Table 8* were implemented for control of heat exchangers under parametric uncertainty. Fouling was considered as parametric uncertainty because it changed along the operation of heat exchangers. The robustness of the designed PID was checked under setpoint tracking and disturbance rejection.

Figure 15a illustrated the response of a step-setpoint-change of the outlet temperature of cold stream (T_{Cout}) of each period of fouling accumulation within the heat exchanger. When the +5% of the setpoint change (from 80 °C to 84 °C) at the time 100 seconds, the control output response could be well controlled to the setpoint. The settling time to the desired setpoint was approximately 106.49 seconds. Despite the overshoot response, they could be controlled to the desired setpoint with the ISE value = 160.18, which refers to the ISE of the clean heat exchanger and 1, 2, and 3 months of the heat exchanger operating period. The associated manipulated variable profile is shown in *Figure 15b*.

Figure 16a shows the T_{Cout} response of each heat exchanger operating period to step change -5% of setpoint change varied from 80 °C reach 76 °C at 100 seconds

with parameter tuning shown in *Table 8*. It shows that the control response obtained from the IMC-based PID controller can be controlled to the desired setpoint, although it took a long time on average convergence compared to step change +5% of setpoint shown in *Figure 15a*. There is an overshoot and oscillation response at all operating intervals. And they tend to increase with the use of heat exchangers for more than three months continuously. The associated manipulated variable profile is shown in *Figure 16b*.
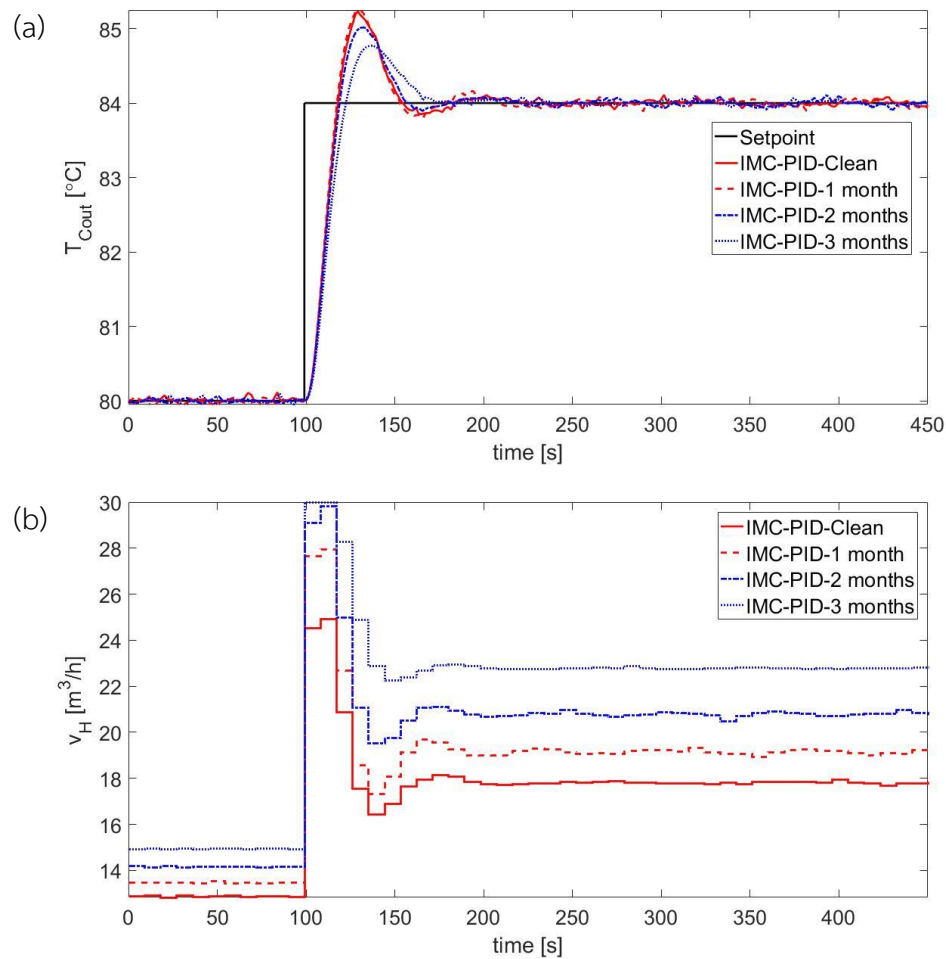
Figure 15. Control response of IMC-PID controller: (a) step change +5% of setpoint tracking in outlet temperature of the cold stream; (b) the associated manipulated variable profile.
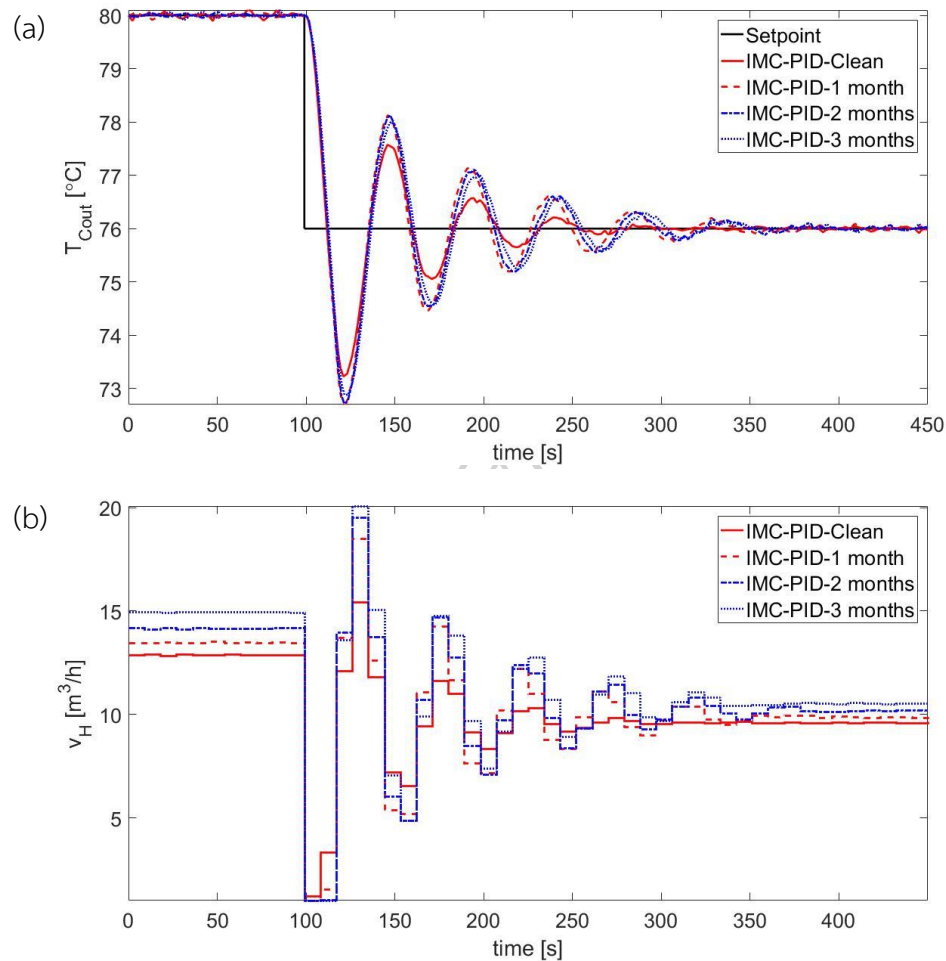
Figure 16. Control response of IMC-PID controller: (a) step change -5% of setpoint tracking in outlet temperature of the cold stream; (b) the associated manipulated variable profile.

Figure 17a and b show the clean heat exchanger's T_{Cout} control response to ±5% disturbance rejection changes, i.e., the flow rate of the cold stream and inlet temperatures of hot and cold streams and manipulated variable, respectively. A +5% increase in the flow rate of the cold stream results in a decrease in outlet temperature at the cold stream. The controller then adjusts the manipulated variable and returns the controlled variable to the setpoint value. When the inlet temperature of the hot stream is increased by 5%, the outlet temperature at the cold stream increased. The controller then reduces the controlled variable to reject the disturbance of inlet temperature at the hot side. The increased inlet temperature

of the cold stream by 5% increased the outlet temperature of cold fluid. The manipulated variable is then adjusted by the controller and the controlled variable is returned to the setpoint value. These results show that the controller can effectively reject disturbance. Since the inlet temperature at the cold stream is one of the disturbance variables that affect control more difficult than others, inlet temperature at the cold stream is used for disturbance rejection for 1, 2, and 3 months of heat exchanger operation period.
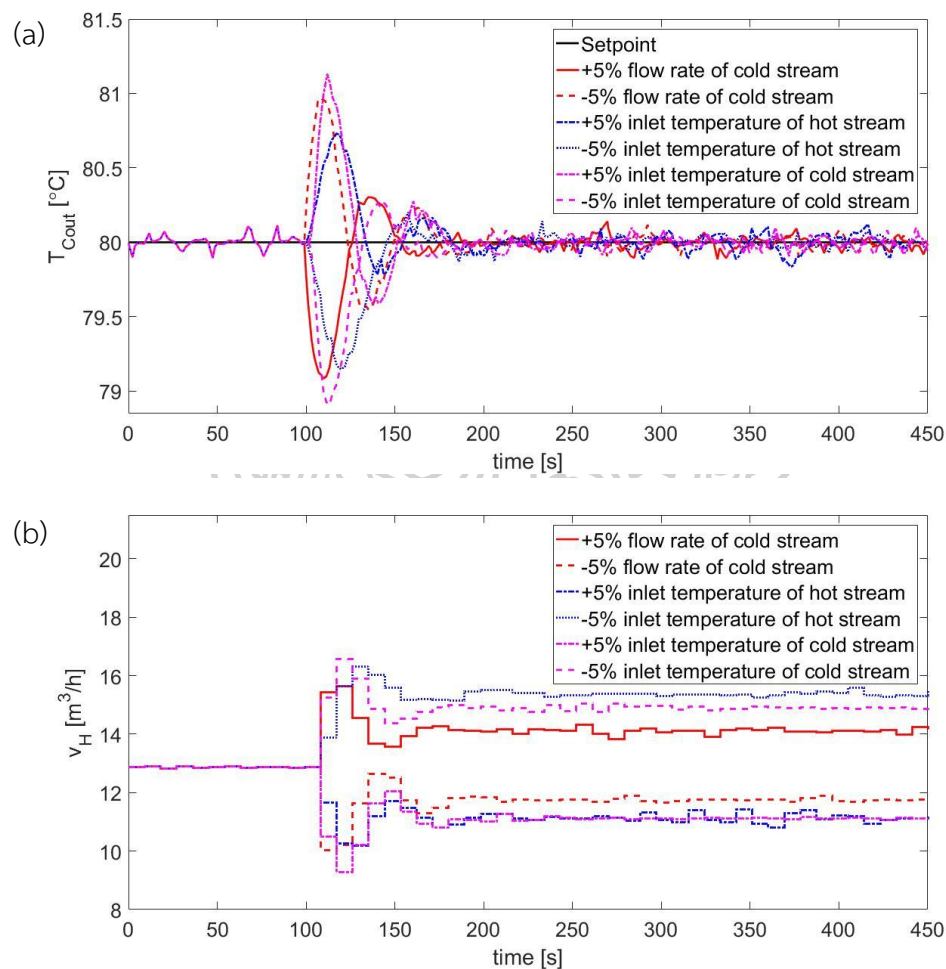
Figure 17. Control response of IMC-PID controller for clean heat exchanger operation: (a) step change ±5% of disturbances rejection in the flow rate of the cold stream and inlet temperatures of hot and cold streams; (b) the associated manipulated variable profile.

Figure 18a and *b* show the control response output temperature of the cold stream at 1, 2, and 3 months heat exchanger operation with a step change +5% of inlet temperature at cold fluid and manipulated variable, respectively. When an increase of 5% inlet temperature of cold fluid at 100 seconds resulted in the controlled variable of each operating period increases. The controller then adjusts the manipulated variable and returns the controlled variable to the setpoint value.
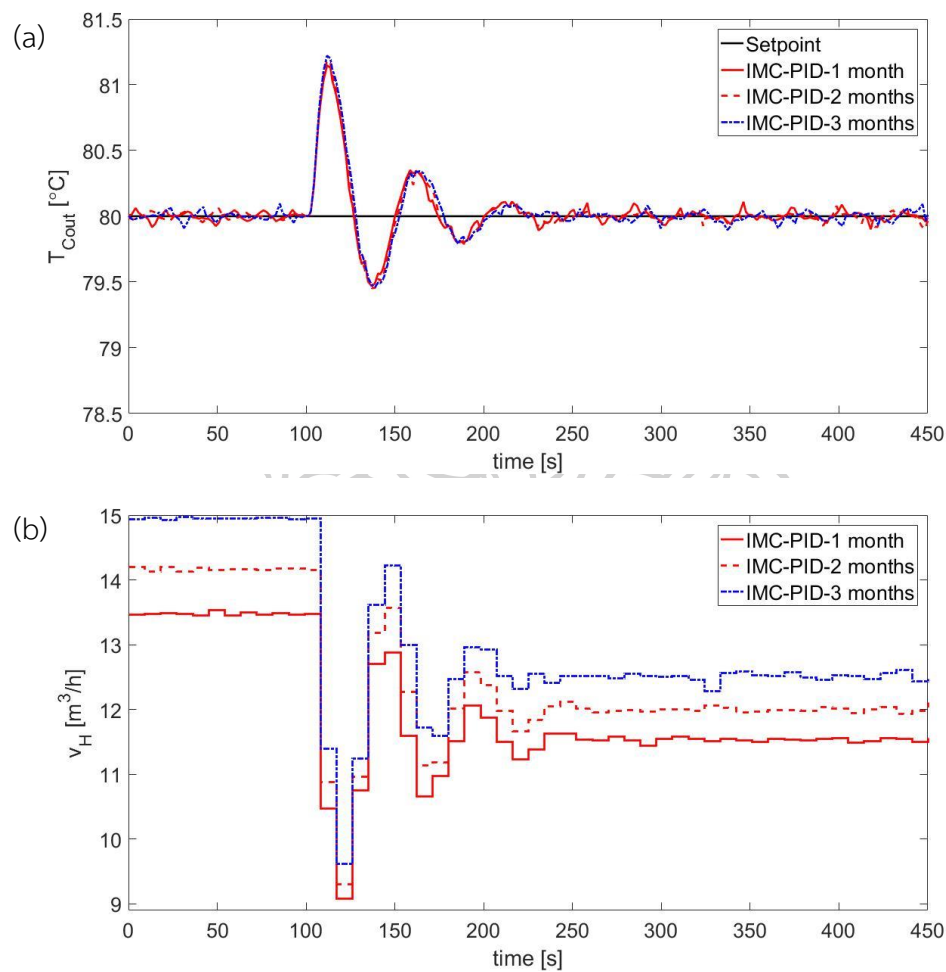
Figure 18. Control response of IMC-PID controller at 1, 2 and 3 months heat exchanger operation: (a) step change +5% of disturbances rejection in the flow rate of the cold stream; (b) the associated manipulated variable profile.

Figure 19a and *b* show the control response output temperature of the cold stream at 1, 2, and 3 months heat exchanger operation with a step change -5% of inlet temperature at cold fluid and manipulated variable, respectively. When the

inlet temperature of the cold stream was reduced by 5% at 100 seconds, the outlet temperature of the cold stream for each period of operation increased. The controller then decreases the controlled variable to reject the disturbance of inlet temperature at the cold fluid. These results indicate that the controller can control the outlet temperature of each operating range even with the disturbance to the system.
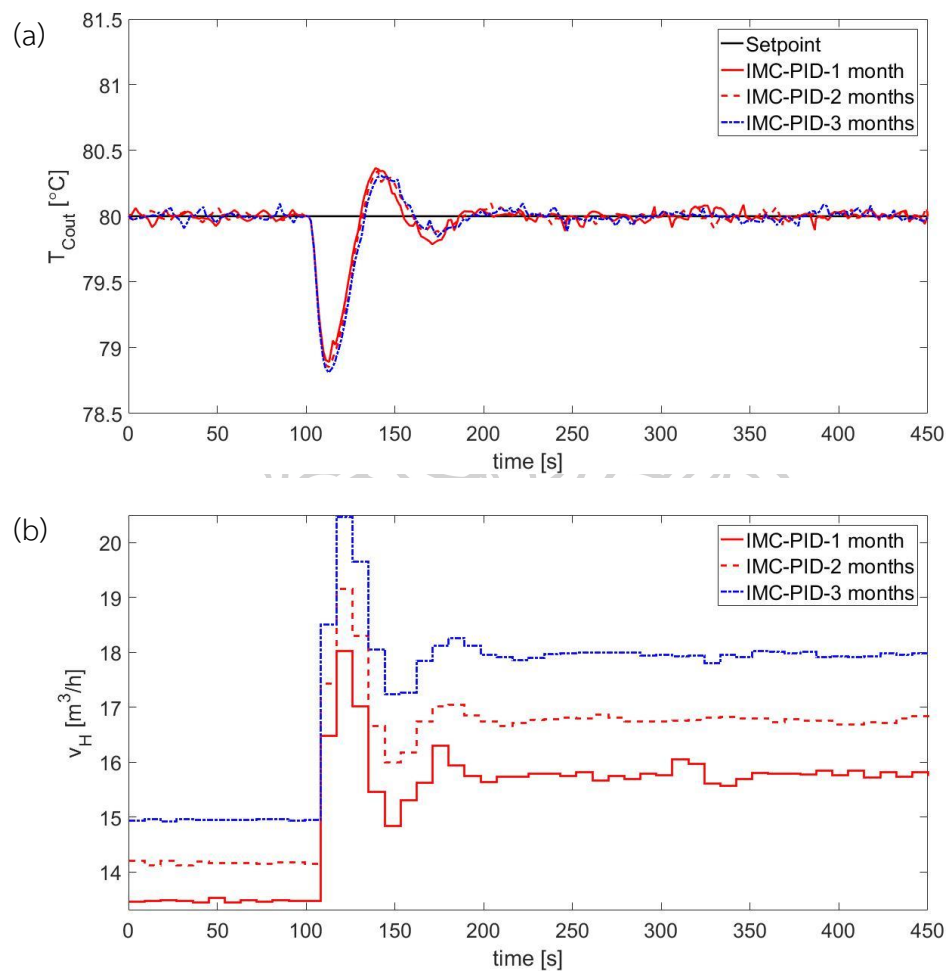
Figure 19. Control response of IMC-PID controller at 1, 2 and 3 months heat exchanger operation: (a) step change -5% of disturbances rejection in the flow rate of the cold stream; (b) the associated manipulated variable profile.

5.4 Temperature control of heat exchangers using NNMPC controller

In the system identification, the NARX-ANN models obtained from section 5.2 were applied in the MPC algorithm. It was shown that the results of network predictions

were nearly close to those generated by the model since the process had few errors in the forecasting and the ANN's output was also suitable for the validation data. The NNMPC started after the ANN model has been sufficiently trained.

The trial and error method is implemented as the procedure for selecting tuning parameters: control horizon (M), prediction horizon (P), and the weight coefficients of the input (w) for NNMPC. In general the control horizon (M) and prediction horizon (P) is suggested as $M \leq P$. The tuning parameters and ISE value of the NNMPC used to control the controlled variable for each heat exchanger operating period are shown in *Table 9*, which is obtained from the setpoint tracking test.

Table 9. The tuning parameters with ISE values of the NNMPC.

Heat exchanger fouling	M	P	w	ISE
clean	6	9	0.01	307.485
1 month	6	10	0.01	324.880
2 months	6	10	0.01	338.963
3 months	6	10	0.01	350.776

The NNMPC parameters designed according to *Table 9* are employed to adjust the controller to test the controls of the heat exchanger system under fouling build-up for setpoint tracking and disturbance rejection. The NNMPC is also used to monitor for fouling inside the heat exchanger. This is an advantage for the NNMPC that can control the heat exchanger meanwhile can monitor fouling.

Figure 20a shows the outlet temperature of cold stream response of each fouling range with step changes +5% of setpoint tracking from 80 ℃ to 84 ℃ with NNMPC tuning parameters shown in *Table 9*. When setpoint tracking is increased by 5% at 100 seconds, the outlet temperature of the cold stream response of each period can be well controlled into the setpoint value. They have the lowest

overshoot response and shortest time to enter the setpoint value compared with the control response from the PID controller shown in *Figure 15a*. Even the average ISE value of the control response obtained with the NNMPC was rather than the PID controller's average ISE value. The associated manipulated variable profile is shown in *Figure 20b*.
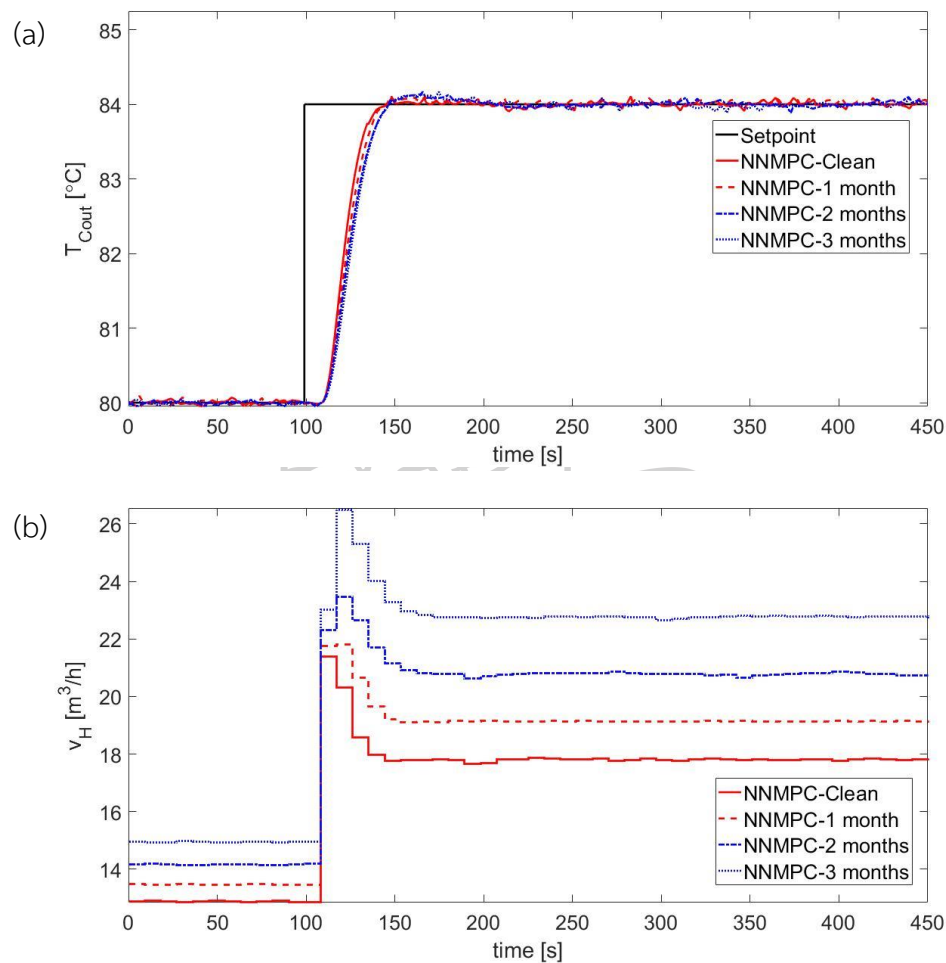
Figure 20. Control response of NNMPC: (a) step change +5% of setpoint tracking in outlet temperature of the cold stream; (b) the associated manipulated variable profile.

Figure 21a and *b* show the outlet temperature of cold stream response of each fouling range with step changes -5% of setpoint and manipulated variable, respectively. When setpoint change is reduced from 80 °C to 76 °C at 100 s, the control response of the controlled variable of each operating range obtained by

NNMPC, shown in *Table 9,* can be controlled into the setpoint value. They have the smallest overshoot and oscillation response compared with the control response obtained with the PID controller shown in *Figure 16a.* They also take a shorter time to enter the setpoint value. These results indicated that NNMPC was able to control the controlled variable better than the PID controller when step the setpoint tracking.
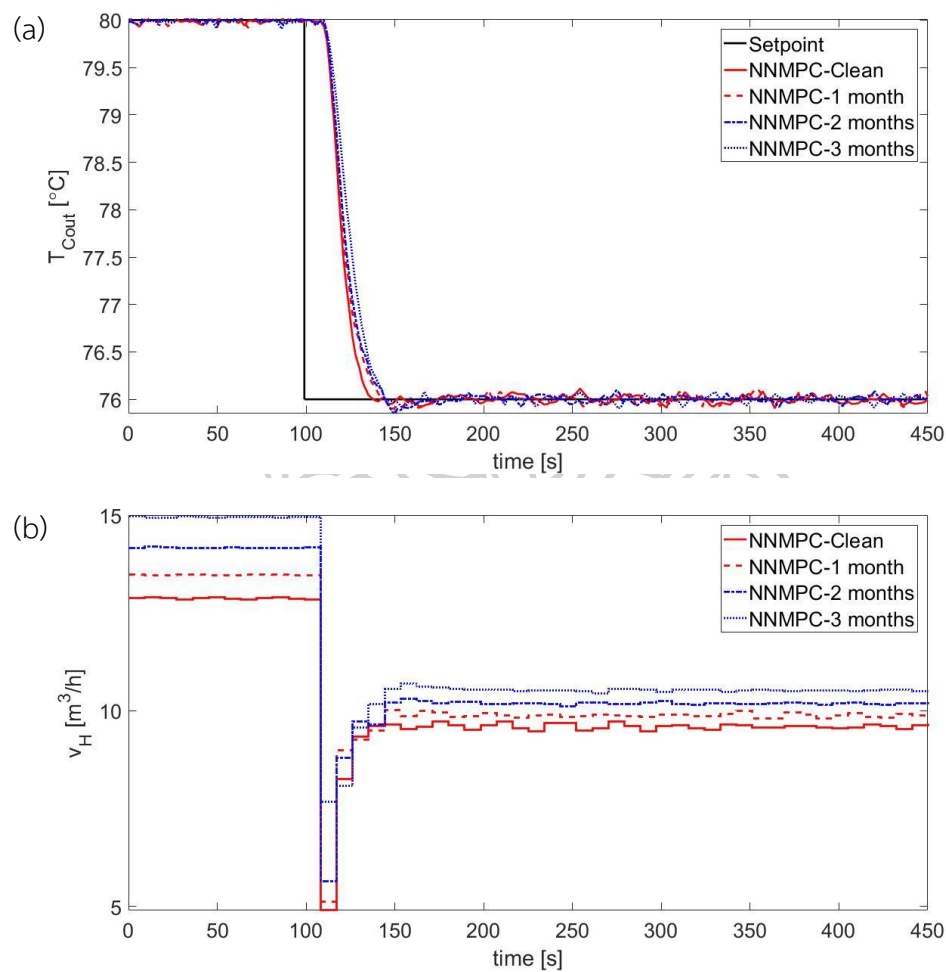
Figure 21. Control response of NNMPC: (a) step change -5% of setpoint tracking in outlet temperature of the cold stream; (b) the associated manipulated variable profile.

Figure 22a and *b* show the control response of the controlled variable for the clean heat exchanger operation with step change ±5% of the flow rate of the cold stream and the inlet temperatures of hot and cold streams and manipulated

variable, respectively. When the flow rate of the cold stream was increased by 5% at 100 s, the controlled variable decreased. NNMPC then adjusted the manipulated variable, and the controlled variable returned to the setpoint with a smaller ISE value than the PID controller. When inlet temperatures of hot and cold streams were increased by 5% at 100 seconds, the outlet temperature of both disturbances increased. The NNMPC then reduces the controlled variable to reject both disturbances in the system with a shorter time reached the setpoint and less ISE than the PID controller.
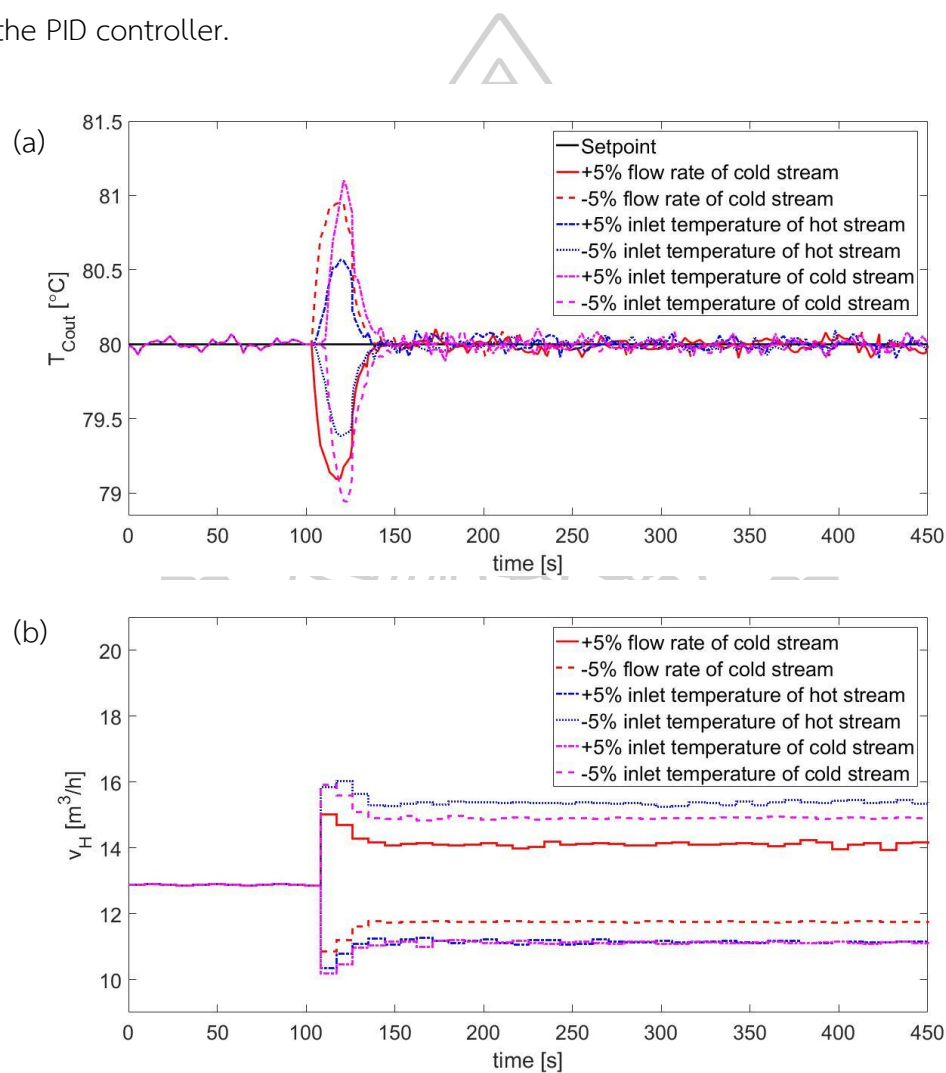
Figure 22. Control response of NNMPC for clean heat exchanger operation: (a) step change ±5% of disturbances rejection in the flow rate of the cold stream and inlet temperatures of hot and cold streams; (b) the associated manipulated variable profile.

Figure 23a and b show the control response of the outlet temperature of cold stream in the 1, 2, and 3 months heat exchanger operation with step change +5% of the inlet temperature of the cold stream and manipulated variable, respectively. When the inlet temperature of the cold stream was increased by 5% at 100 seconds for each operating period, the controlled variable decreased and returned to the setpoint value after NNMPC adjusted the manipulated variable. The time entered the setpoint for control response of the NNMPC was shorter than the PID controller.
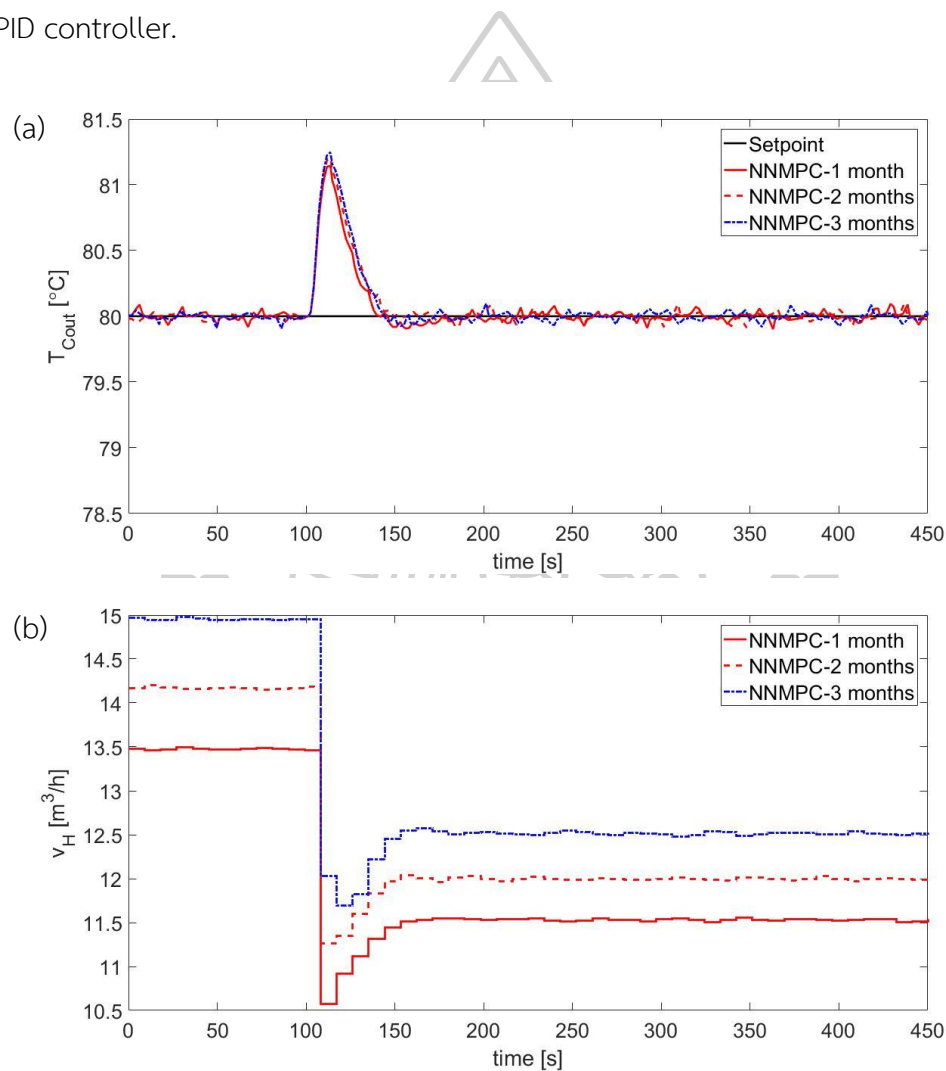
Figure 23. Control response of NNMPC at 1, 2 and 3 months heat exchanger operation: (a) step change +5% of disturbances rejection in the flow rate of the cold stream; (b) the associated manipulated variable profile.

Figure 24a and b show the control response of the outlet temperature of cold stream in the 1, 2, and 3 months heat exchanger operation with step change of -5% in the inlet temperature of the cold stream and manipulated variable, respectively. The controlled variable increased when the inlet temperature of the cold stream of each operating period was decreased by 5% after the manipulated variable was adjusted by NNMPC and the controlled variable returned to the setpoint value. They took a shorter time to enter the setpoint than the control response received from the PID controller. These show that NNMPC can reject disturbances better than the PID controller.
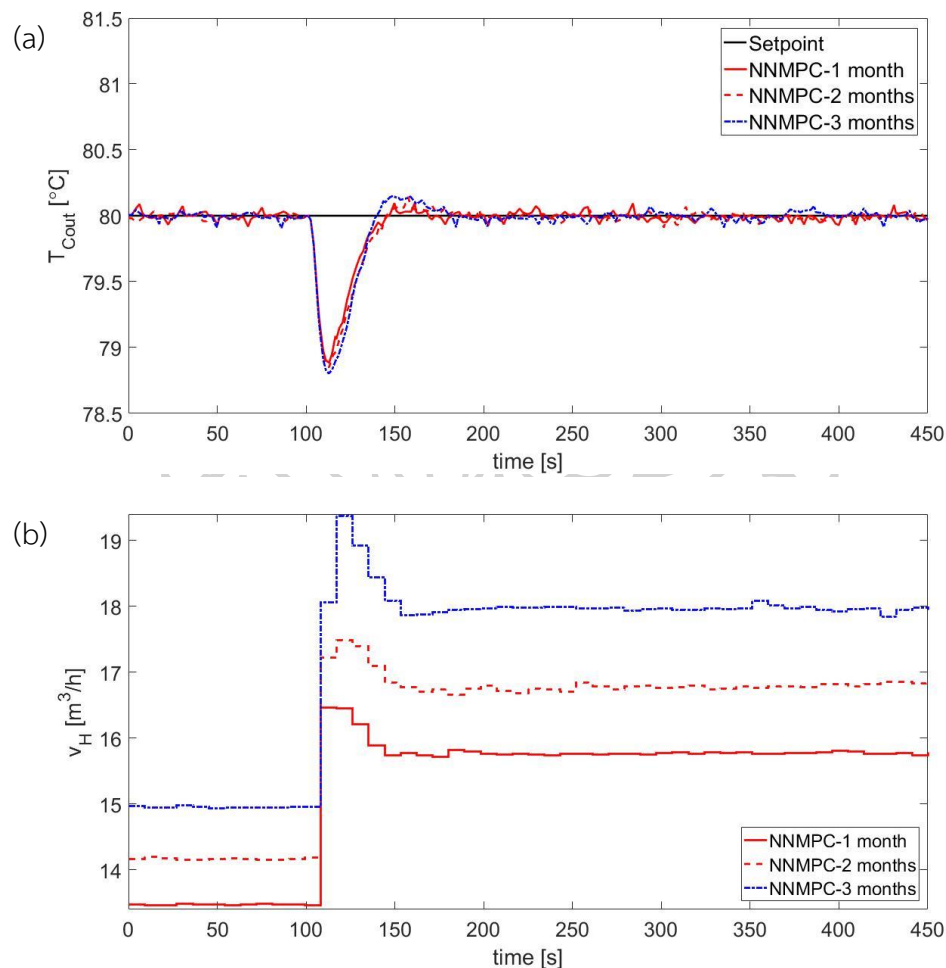
Figure 24. Control response of NNMPC at 1, 2 and 3 months heat exchanger operation: (a) step change -5% of disturbances rejection in the flow rate of the cold stream; (b) the associated manipulated variable profile.

Figure 25 and *Figure 26* show the prediction results of fouling formation at cold and hot sides for each heat exchanger operating period, respectively, based on the prediction of NARX-ANN model in ±5% step the setpoint tracking. It has been shown that NARX-ANN model can use to monitor fouling formation.
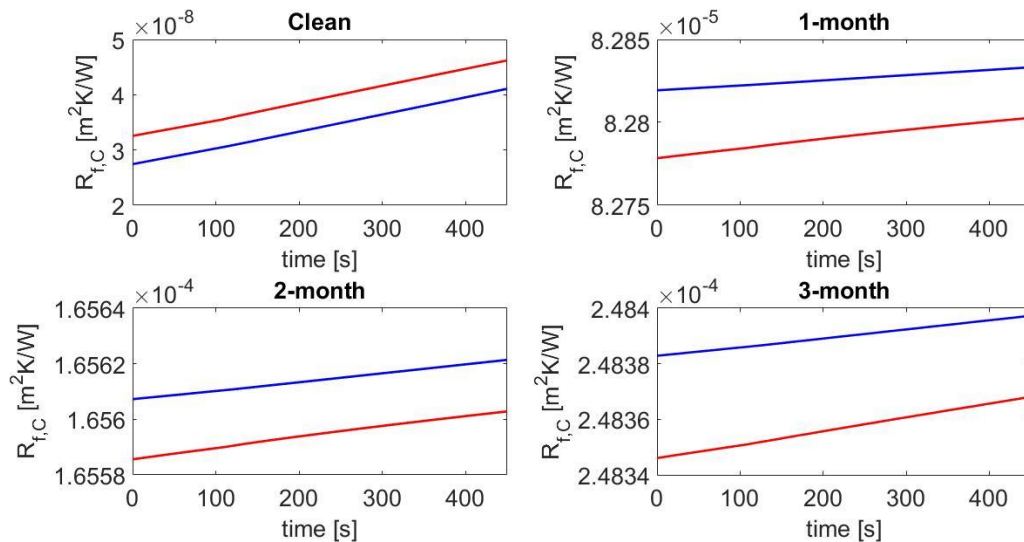
Figure 25. The results fouling formation prediction at the cold side for each heat exchanger operating period between plant (blue) and NNMPC (red).
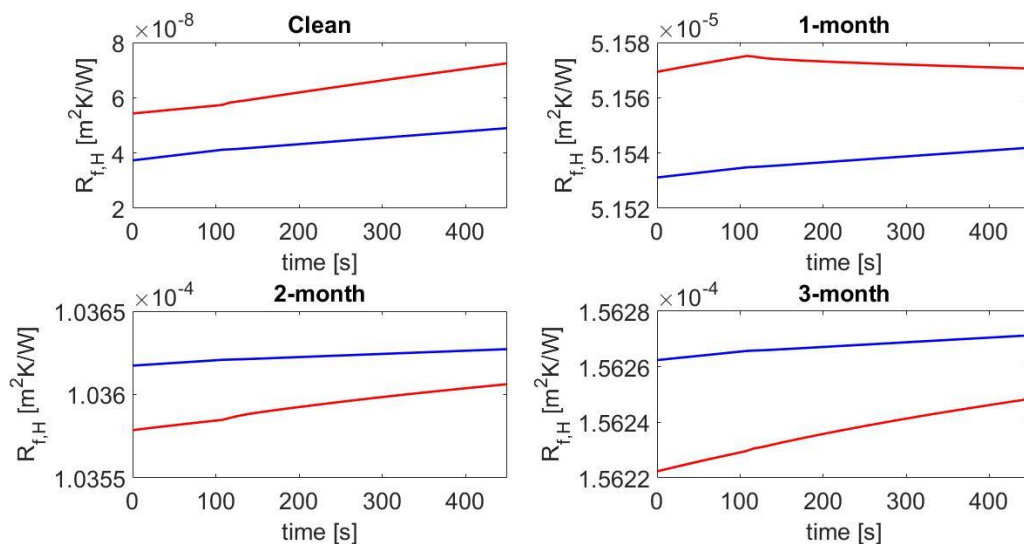
Figure 26. The results fouling formation prediction at the hot side for each heat exchanger operating period between plant (blue) and NNMPC (red).

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

Fouling is one of the main problems that often arise during the operation of heat exchangers. The presence of fouling results in a reduction of heat transfer efficiency and can cause a temperature control problem. To overcome this problem, the design of the controller for the heat exchanger under fouling is therefore essential. In this study, an artificial neural network (ANN) was used to predict the fouling factor and identify a system for heat exchangers under parameter uncertainty in a neural network-based model predictive control (NNMPC) design with a nonlinear autoregressive network with exogenous inputs (NARX). The ANN training dataset was obtained from a cell-based dynamic heat exchanger model integrated with the threshold fouling model. The inputs of the ANN model included the flow rates of hot and cold streams and the inlet temperatures of hot and cold streams while the outputs of the ANN model included the outlet temperatures of hot and cold streams and fouling factors. Effects of the number of hidden neurons and training algorithms to ANN topology were also studied. The statistical indices used to determine the best topology include the mean square error (MSE), the regression coefficient (R2), and processing time. In the controller design, the NNMPC and PID controller are used to control the temperature of the heat exchanger system. It has been shown that the NNMPC can control the temperature for the heat exchanger under fouling build-up better than the PID controller in terms of setpoint tracking and disturbance rejection. It can also use to predict the effect of fouling factors.

6.2 Recommendations

Since the NARX-ANN model is trained offline, it is necessary to create 4 NARX-ANN models, namely NARX-ANN of the clean heat exchanger and 1, 2, and 3 months of heat exchanger operating period, to be used as a prediction model for the MPC. An

online ANN may reduce the time it takes to create and reduce the number of ANN models for the heat exchanger.

REFERENCES

1. Li, S., et al., *Adhesive particulate flow: The discrete-element method and its application in energy and environmental engineering.* Progress in Energy and Combustion Science, 2011. **37**(6): p. 633-668.

2. Ebert, W.A. and C.B. Panchal, *Analysis of Exxon crude-oil-slip stream coking data*, in *Proceedings of Mitigation of Industrial Heat Exchangers*. 1995: California, USA. p. 451-460.

3. Panchal, C.B., et al., *Threshold conditions for crude oil fouling*, in *Proceedings of an International Conference on Understanding Heat Exchanger Fouling and Its Mitigation*. 1997: Pascoli, Italy. p. 273-282.

4. Polley, G.T., et al., *Evaluation of laboratory crude oil threshold fouling data for application to refinery pre-heat trains.* Applied Thermal Engineering, 2002. **22**: p. 777–788.

5. Jafari Nasr, M.R. and M.M. Givi, *Modeling of crude oil fouling in preheat exchangers of refinery distillation units.* Applied Thermal Engineering, 2006. **26**: p. 1572–1577.

6. Aminian, J. and S. Shahhosseini, *Evaluation of ANN modeling for prediction of crude oil fouling behavior.* Applied Thermal Engineering, 2008. **28**: p. 668–674.

7. Aminian, J. and S. Shahhosseini, *Neuro-based formulation to predict fouling threshold in crude preheaters.* International Communications in Heat and Mass Transfer, 2009. **36**: p. 525–531.

8. Davoudi, E. and B. Vaferi, *Applying artificial neural networks for systematic estimation of degree of fouling in heat exchangers.* Chemical Engineering Research and Design, 2018. **130**: p. 138–153.

9. Mathisen, K.W., M. Morari, and S. Skogestad, *Dynamic models for heat exchangers and heat exchanger networks.* Computers and Chemical Engineering, 1994. **18**: p. 459–546.

10. Varga, E.I., K.M. Hangos, and F. Szigeti, *Controllability and observability of heat exchanger networks in the time-varying parameter case.* Control Engineering

Practice, 1995. **3**(10): p. 1409-1419.

11.   Georgiadis, M.C. and S. Macchietto, *Dynamic modelling and simulation of plate heat exchangers under milk fouling.* Chemical Engineering Science, 2000. **55**(9): p. 1605-1619.

12.   Roetzel, W., M. Li, and X. Luo, *Dynamic behaviour of heat exchangers.* Advanced Computational Methods in Heat Transfer, 2003: p. 451-460.

13.   Ansari, M.R. and V. Mortazavi, *Simulation of dynamical response of a countercurrent heat exchanger to inlet temperature or mass flow rate change.* Applied Thermal Engineering, 2006. **26**(17): p. 2401-2408.

14.   Doležal, R., et al., *Solution of the heat-exchanger equation system*, in *Proceedings of the IFAC Workshop Modeling and Control of Electric Power Plants*. 1983: Como, Italy. p. 69–75.

15.   Dobos, L., et al., *Dynamic model and control of heat exchanger networks for district heating.* Hungarian Journal of Industrial Chemistry, 2009. **37**: p. 37-49.

16.   Dobos, L. and J. Abonyi, *Controller tuning of district heating networks using experiment design techniques.* Energy, 2011. **36**(8): p. 4633-4639.

17.   Varbanov, P.S., J.J. Klemeš, and F. Friedler, *Cell-based dynamic heat exchanger models—Direct determination of the cell number and size.* Computers and Chemical Engineering, 2011. **35**: p. 943–948.

18.   Zhang, M., et al., *An Integrated FDD Approach for an Intensified HEX/Reactor.* Journal of Control Science and Engineering, 2018. **2018**.

19.   Crittenden, B.D., S.T. Kolaczkowski, and S.A. Hout, *Modelling hydrocarbon fouling.* Chemical Engineering Research and Design, 1987. **65**: p. 171–179.

20.   Knudsen, J.G., D. Lin, and W.A. Ebert, *The determination of the threshold fouling curve for a crude oil*, in *Proceedings of the International Conference on Understanding Heat Exchanger Fouling and Its Mitigation*. 1997: Pascoli, Italy. p. 265–272.

21.   Scarborough, C.E., et al., *Coking of crude oil at high heat flux levels.* Chemical Engineering Progress, 1979. **75**: p. 41–46.

22.   Brahim, F., W. Augustin, and M. Bohnet, *Numerical Simulation of the Fouling on Structured Heat Transfer Surfaces (Fouling).* Proceedings of Heat Exchanger

Fouling and Cleaning: Fundamentals and Applications, 2004. **17**.

23.   Liporace, F. and S. Oliveira, *Real Time Fouling Diagnosis and Heat Exchanger Performance.* Heat Transfer Engineering, 2007. **28**.

24.   Haghshenasfard, M., et al., *On Numerical Study of Calcium Sulphate Fouling Under Sub-cooled Flow Boiling Conditions.* Applied Thermal Engineering, 2015. **81**.

25.   Najibi, S.H., H. Müller-Steinhagen, and M. Jamialahmadi, *Calcium sulphate scale formation during subcooled flow boiling.* Chemical Engineering Science, 1997. **52**(8): p. 1265-1284.

26.   Jambunathan, K., et al., *Evaluating convective heat transfer coefficients using neural networks.* International Journal of Heat and Mass Transfer, 1996. **39**(11): p. 2329-2332.

27.   Malayeri, M.R. and H. Müller-Steinhagen, *Analysis of Fouling Data Based on Prior Knowledge*. 2003.

28.   Ghajar, A., L.A.P. Tam, and S.I.K. Tam, *Improved Heat Transfer Correlation in the Transition Region for a Circular Tube with Three Inlet Configurations Using Artificial Neural Networks.* Heat Transfer Engineering, 2010. **March 2004**: p. 30-40.

29.   Romeo, L.M. and R. Gareta, *Hybrid system for fouling control in biomass boilers.* Engineering Applications of Artificial Intelligence, 2006. **19**: p. 915–925.

30.   Lalot, S. and H. Pálsson, *Detection of fouling in a cross-flow heat exchanger using a neural network based technique.* International Journal of Thermal Sciences, 2010. **49**: p. 675-679.

31.   Garcia, R.F., *Improving heat exchanger supervision using neural networks and rule based techniques.* Expert Systems with Applications, 2012. **39**: p. 3012-3021.

32.   Vaferi, B., et al., *Artificial neural network approach for prediction of thermal behavior of nanofluids flowing through circular tubes.* Powder Technology, 2014. **267**: p. 1-10.

33.   Biyanto, T.R., *Fouling resistance prediction using artificial neural network nonlinear auto-regressive with exogenous input model based on operating*

conditions and fluid properties correlations. Vol. 1737. 2016. 050001.

34.   Wang, Q., et al., *Prediction of heat transfer rates for shell-and-tube heat exchangers by artificial neural networks approach.* Journal of Thermal Science, 2006. **15**(3): p. 257-262.

35.   Xie, G.N., et al., *Heat transfer analysis for shell-and-tube heat exchangers with experimental data by artificial neural networks approach.* Applied Thermal Engineering, 2007. **27**(5): p. 1096-1104.

36.   Pandharipande, S., et al., *Optimising ANN architecture for shell and tube heat exchanger modelling.* Indian Journal of Chemical Technology, 2004. **11**: p. 804-810.

37.   Mandavgane, S. and S. Pandharipande, *Application of optimum ANN architecture for heat exchanger modeling.* Indian Journal of Chemical Technology, 2006. **13**: p. 634-639.

38.   Duran, O., N. Rodriguez, and L.A. Consalter, *Neural networks for cost estimation of shell and tube heat exchangers.* Expert Systems with Applications, 2009. **36**(4): p. 7435-7440.

39.   Fadare, D., A. Fatona, and M. Sc, *Artificial Neural Network Modeling of Heat Transfer in a Staggered Cross-flow Tube-type Heat Exchanger.* Pac. J. Sci. Technol., 2008. **9**.

40.   Kashani, M.N., et al., *Dynamic crude oil fouling prediction in industrial preheaters using optimized ANN based moving window technique.* Chemical Engineering Research and Design, 2012. **90**: p. 938–949.

41.   Ławryńczuk, M., *Modelling and nonlinear predictive control of a yeast fermentation biochemical reactor using neural networks.* Chemical Engineering Journal, 2008. **145**(2): p. 290-307.

42.   Vasickaninová, A., et al., *Neural network predictive control of a heat exchanger.* Applied Thermal Engineering, 2011. **31**(13): p. 2094-2100.

43.   Pazhooh, F., et al., *Multivariable adaptive neural network predictive control in the presence of measurement time-delay; application in control of Vinyl Acetate monomer process.* Journal of Process Control, 2018. **66**.

44.   Kimaev, G. and L.A. Ricardez-Sandoval, *Nonlinear model predictive control of a multiscale thin film deposition process using artificial neural networks.* Chemical Engineering Science, 2019. **207**: p. 1230-1245.

45.   Shin, Y., R. Smith, and S. Hwang, *Development of model predictive control system using an artificial neural network: A case study with a distillation column.* Journal of Cleaner Production, 2020. **277**: p. 124124.

46.   Jamil, M., et al., *Neural network predictive control of vibrations in tall structure: An experimental controlled vision.* Computers & Electrical Engineering, 2021. **89**: p. 106940.

47.   Wang, Y., et al., *A review of experimental measurement and prediction models of crude oil fouling rate in crude refinery preheat trains.* Asia-Pacific Journal of Chemical Engineering, 2015. **10**: p. 607–625.

48.   Profillidis, V.A. and G.N. Botzoris, *Chapter 8 - Artificial intelligence—Neural network methods.* Modeling of Transport Demand, 2019: p. 353-382.

49.   Liu, Q., et al., *An Optimal NARX Neural Network Identification Model for a Magnetorheological Damper With Force-Distortion Behavior.* Frontiers in Materials, 2020. **7**(10).

50.   Yan, D., et al., *Bayesian regularisation neural network based on artificial intelligence optimisation.* International Journal of Production Research, 2016. **55**: p. 1-22.

51.   Nepal, B. and O.P. Yadav, *Bayesian belief network-based framework for sourcing risk analysis during supplier selection.* International Journal of Production Research, 2015. **53**: p. 1-22.

52.   Møller, M.F., *A scaled conjugate gradient algorithm for fast supervised learning.* Neural Networks, 1993. **6**(4): p. 525-533.

53.   Rivera, D.E., M. Morari, and S. Skogestad, *Internal model control: PID controller design.* Industrial & Engineering Chemistry Process Design and Development, 1986. **25**(1): p. 252-265.

54.   Chien, I.-L. and P.S. Fruehauf, *Consider IMC tuning to improve controller performance.* Chemical Engineering Progress, 1990. **86**: p. 33.

55.   Skogestad, S., *Simple analytic rules for model reduction and PID controller*

tuning. Journal of Process Control, 2003. **13**(4): p. 291-309.

56.   Sinnott, R., *Chemical Engineering Design*. 4th ed. Coulson and Richardson's Chemical Engineering. 2005: Elsevier, Butterworth-Heinemann, Oxford.

57.   Cybenko, G., *Approximation by superpositions of asigmoidal function.* Mathematics of Control, Signals, and Systems, 1989. **2**: p. 303-314.

58.   Smith, C.A. and B.B. Corripio, *Principles and practices of automatic process control*. 2005: Wiley.

APPENDIX

APPENDIX A

DETERMINING THE OPTIMAL NUMBER OF DELAYS

Table A1. Determination of the best topology of the NARX-ANN model by the different number of delays for a clean heat exchanger.

Number of delays (n_u, n_y)	Train		Test	
	MSE	R^2	MSE	R^2
0,1	0.00125000	0.99959	0.00062474	0.99987
1,1	0.00006690	0.99999	0.00013667	0.99997
2,1	0.00000951	0.99999	0.00001556	0.99999
3,1	**0.00000451**	**0.99999**	**0.00000756**	**0.99999**
4,1	0.00000523	0.99999	0.00000344	0.99999
1,2	0.00052700	0.99987	0.00034885	0.99992
2,2	0.00004410	0.99999	0.00008212	0.99998
3,2	0.00003460	0.99999	0.00054070	0.99999
4,2	0.00000471	0.99999	0.00000288	0.99999
1,3	0.00143000	0.99966	0.00102220	0.99976
2,3	0.00026100	0.99994	0.00021988	0.99995
3,3	0.00003430	0.99999	0.00005760	0.99999
4,3	0.00001050	0.99999	0.00001011	0.99999
1,4	0.00249000	0.99942	0.00157730	0.99965
2,4	0.00066700	0.99985	0.00049566	0.99989
3,4	0.00013900	0.99997	0.00012676	0.99997
4,4	0.00002120	0.99999	0.00002674	0.99999

Table A2. Determination of the best topology of the NARX-ANN model by the different number of delays for operating a month of the heat exchanger.

Number of delays	Train		Test	
(n_u, n_y)	MSE	R^2	MSE	R^2
0,1	0.00120000	0.99956	0.00036224	0.99990
1,1	0.00007800	0.99998	0.00008529	0.99998
2,1	**0.00000726**	**0.99999**	**0.00000680**	**0.99999**
3,1	0.00001820	0.99999	0.00001657	0.99999
4,1	0.00000802	0.99987	0.00000649	0.99992
1,2	0.00044200	0.99987	0.00032540	0.99992
2,2	0.00004180	0.99999	0.00005457	0.99999
3,2	0.00001030	0.99999	0.00000833	0.99999
4,2	0.00000758	0.99999	0.00000635	0.99999
1,3	0.00120000	0.99966	0.00072669	0.99981
2,3	0.00023200	0.99994	0.00016920	0.99996
3,3	0.00003000	0.99999	0.00003284	0.99999
4,3	0.00000968	0.99999	0.00000937	0.99999
1,4	0.00195000	0.99944	0.00136560	0.99963
2,4	0.00058800	0.99984	0.00043139	0.99989
3,4	0.00013400	0.99996	0.00011538	0.99997
4,4	0.00002310	0.99999	0.00002141	0.99999

Table A3. Determination of the best topology of the NARX-ANN model by the different number of delays for operating 2 months of the heat exchanger.

Number of delays	Train		Test	
(n_u, n_y)	MSE	R^2	MSE	R^2
0,1	0.00134000	0.99948	0.00053187	0.99985
1,1	0.00004400	0.99999	0.00009975	0.99997
2,1	**0.00000778**	**0.99999**	**0.00000649**	**0.99997**
3,1	0.00001210	0.99999	0.00000960	0.99999
4,1	0.00000888	0.99999	0.00000719	0.99999
1,2	0.00053000	0.99985	0.00048054	0.99986
2,2	0.00003280	0.99999	0.00006393	0.99998
3,2	0.00000992	0.99999	0.00000903	0.99999
4,2	0.00000784	0.99999	0.00000758	0.99999
1,3	0.00135000	0.99963	0.00086545	0.99977
2,3	0.00025700	0.99993	0.00020167	0.99995
3,3	0.00003090	0.99999	0.00004760	0.99999
4,3	0.00001340	0.99999	0.00001559	0.99999
1,4	0.00237000	0.99936	0.00129390	0.99966
2,4	0.00063400	0.99983	0.00042812	0.99989
3,4	0.00021200	0.99994	0.00021589	0.99994
4,4	0.00002160	0.99999	0.00002100	0.99999

Table A4. Determination the best topology of the NARX-ANN model by the different number of delays for operating 3 months of the heat exchanger.

Number of delays (n_u, n_y)	Train		Test	
	MSE	R^2	MSE	R^2
0,1	0.00133000	0.99952	0.00051139	0.99986
1,1	0.00005350	0.99998	0.00010377	0.99997
2,1	**0.00000807**	**0.99999**	**0.00000948**	**0.99999**
3,1	0.00001710	0.99999	0.00001383	0.99999
4,1	0.00000875	0.99999	0.00000745	0.99999
1,2	0.00060200	0.99984	0.00039684	0.99991
2,2	0.00003080	0.99999	0.00005770	0.99999
3,2	0.00001410	0.99999	0.00001814	0.99999
4,2	0.00000810	0.99999	0.00000307	0.99999
1,3	0.00160000	0.99959	0.00074841	0.99983
2,3	0.00027400	0.99993	0.00017690	0.99996
3,3	0.00003310	0.99999	0.00003867	0.99999
4,3	0.00001550	0.99999	0.00001261	0.99999
1,4	0.00268000	0.99929	0.00117510	0.99972
2,4	0.00074700	0.99982	0.00045115	0.99989
3,4	0.00017100	0.99996	0.00011589	0.99997
4,4	0.00002890	0.99999	0.00002249	0.99999

APPENDIX B

DETERMINING THE OPTIMAL NUMBER OF HIDDEN NEURONS

Table B1. Determination the best architecture of the NARX-ANN models with a different number of hidden neurons for a clean heat exchanger.

Number of hidden neurons	Train		Test	
	MSE	R^2	MSE	R^2
1	0.10100000	0.97746	0.10042000	0.97729
2	0.00593000	0.99884	0.00544350	0.99895
3	0.00117000	0.99992	0.00105460	0.99996
4	0.00023200	0.99993	0.00012410	0.99997
5	0.00015200	0.99995	0.00005156	0.99999
6	0.00002680	0.99999	0.00002341	0.99999
7	0.00001220	0.99999	0.00001566	0.99999
8	0.00000910	0.99999	0.00001432	0.99999
9	0.00000454	0.99999	0.00000654	0.99999
10	0.00000451	0.99999	0.00000756	0.99999
11	0.00000506	0.99999	0.00000982	0.99999
12	0.00000454	0.99999	0.00000498	0.99999
13	0.00000168	0.99999	0.00000459	0.99999
14	0.00000138	0.99999	0.00000275	0.99999
15	0.00000292	0.99999	0.00000815	0.99999
16	0.00000223	0.99999	0.00000542	0.99999
17	0.00000105	0.99999	0.00000164	0.99999
18	0.00000115	0.99999	0.00000272	0.99999
19	0.00000089	0.99999	0.00000204	0.99999
20	0.00000140	0.99999	0.00000307	0.99999

Table B2. Determination the best architecture of the NARX-ANN models with a different number of hidden neurons for operating a month of the heat exchanger.

Number of hidden neurons	Train		Test	
	MSE	R^2	MSE	R^2
1	0.11400000	0.96480	0.10895000	0.96613
2	0.00572000	0.99866	0.00572100	0.99865
3	0.00080300	0.99993	0.00071641	0.99996
4	0.00020000	0.99993	0.00011966	0.99997
5	0.00011600	0.99996	0.00006782	0.99998
6	0.00004290	0.99999	0.00003229	0.99999
7	0.00002380	0.99999	0.00002093	0.99999
8	0.00000873	0.99999	0.00005322	0.99999
9	0.00000838	0.99999	0.00000733	0.99999
10	0.00000726	0.99999	0.00000680	0.99999
11	0.00000670	0.99999	0.00000943	0.99999
12	0.00000543	0.99999	0.00000478	0.99999
13	0.00000510	0.99999	0.00001461	0.99999
14	0.00000502	0.99999	0.00000384	0.99999
15	0.00000452	0.99999	0.00000514	0.99999
16	0.00000651	0.99999	0.00000839	0.99999
17	0.00000486	0.99999	0.00000517	0.99999
18	0.00000488	0.99999	0.00000520	0.99999
19	0.00000448	0.99999	0.00000852	0.99999
20	0.00000336	0.99999	0.00000336	0.99999

Table B3. Determination the best architecture of the NARX-ANN models with a different number of hidden neurons for operating 2 months of the heat exchanger.

Number of hidden neurons	Train		Test	
	MSE	R^2	MSE	R^2
1	0.11300000	0.96478	0.10791000	0.96621
2	0.00664000	0.99874	0.00652320	0.99877
3	0.00191000	0.99992	0.00184860	0.99996
4	0.00022300	0.99993	0.00014327	0.99996
5	0.00012000	0.99996	0.00007069	0.99998
6	0.00003440	0.99999	0.00002703	0.99999
7	0.00002590	0.99999	0.00001882	0.99999
8	0.00002120	0.99999	0.00000786	0.99999
9	0.00001610	0.99999	0.00004163	0.99999
10	0.00000978	0.99999	0.00000649	0.99999
11	0.00000977	0.99999	0.00000793	0.99999
12	0.00000650	0.99999	0.00000481	0.99999
13	0.00000545	0.99999	0.00000466	0.99999
14	0.00000498	0.99999	0.00000489	0.99999
15	0.00000398	0.99999	0.00000433	0.99999
16	0.00000707	0.99999	0.00000765	0.99999
17	0.00000391	0.99999	0.00000417	0.99999
18	0.00000372	0.99999	0.00000400	0.99999
19	0.00000324	0.99999	0.00000442	0.99999
20	0.00000300	0.99999	0.00000309	0.99999

Table B4. Determination the best architecture of the NARX-ANN models with a different number of hidden neurons for operating 3 months of the heat exchanger.

Number of hidden neurons	Train		Test	
	MSE	R^2	MSE	R^2
1	0.11900000	0.96578	0.11113000	0.96774
2	0.00610000	0.99868	0.00598690	0.99870
3	0.00074600	0.99993	0.00067793	0.99996
4	0.00018000	0.99994	0.00010407	0.99997
5	0.00011300	0.99997	0.00006952	0.99998
6	0.00003950	0.99999	0.00003876	0.99999
7	0.00003760	0.99999	0.00003405	0.99999
8	0.00001930	0.99999	0.00002033	0.99999
9	0.00001200	0.99999	0.00001560	0.99999
10	0.00001107	0.99999	0.00000948	0.99999
11	0.00001060	0.99999	0.00001329	0.99999
12	0.00000919	0.99999	0.00001498	0.99999
13	0.00000845	0.99999	0.00000951	0.99999
14	0.00000608	0.99999	0.00000534	0.99999
15	**0.00000548**	**0.99999**	**0.00000788**	**0.99999**
16	0.00000570	0.99999	0.00000933	0.99999
17	0.00000598	0.99999	0.00000698	0.99999
18	0.00000549	0.99999	0.00000836	0.99999
19	0.00000718	0.99999	0.00000858	0.99999
20	0.00000519	0.99999	0.00000831	0.99999

APPENDIX C

DETERMINING THE BEST TRAINING ALGORITHM

Table C1. Determination the best architecture of the NARX-ANN models with a different training algorithm for a clean heat exchanger.

The training algorithm	MSE	R^2	Processing time (s/epoch)
Bayesian regulation backpropagation	**0.00000237**	**0.99999**	**0.12515**
Levenberg-Marquardt backpropagation	0.00000451	0.99999	0.12143
Scaled conjugate gradient backpropagation	0.00010800	0.99997	0.00748

Table C2. Determination the best architecture of the NARX-ANN models with a different training algorithm for operating a month of the heat exchanger.

The training algorithm	MSE	R^2	Processing time (s/epoch)
Bayesian regulation backpropagation	**0.00000392**	**0.99999**	**0.21800**
Levenberg-Marquardt backpropagation	0.00000452	0.99999	0.17413
Scaled conjugate gradient backpropagation	0.00009810	0.99998	0.00870

Table C3. Determination the best architecture of the NARX-ANN models with a different training algorithm for operating 2 months of the heat exchanger.

The training algorithm	MSE	R^2	Processing time (s/epoch)
Bayesian regulation backpropagation	**0.00000307**	**0.99999**	**0.17946**
Levenberg-Marquardt backpropagation	0.00000398	0.99999	0.17949
Scaled conjugate gradient backpropagation	0.00017300	0.99995	0.00858

Table C4. Determination the best architecture of the NARX-ANN models with a different training algorithm for operating 3 months of the heat exchanger.

The training algorithm	MSE	R^2	Processing time (s/epoch)
Bayesian regulation backpropagation	**0.00000380**	**0.99999**	**0.19101**
Levenberg-Marquardt backpropagation	0.00000548	0.99999	0.18000
Scaled conjugate gradient backpropagation	0.00013500	0.99996	0.00920

VITA

NAME   Ratchaneekorn Sroipetch

DATE OF BIRTH   06 December 1995

PLACE OF BIRTH   Chanthaburi

INSTITUTIONS ATTENDED   Silpakorn University

HOME ADDRESS   181/2 Soi 5, Tesabansai 2 Road, Khlung, Chanthaburi, Thailand, 22110