We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Chapter

Identification of Heat Exchanger by Neural Network Autoregressive with Exogenous Input Model

Tatang Mulyana

Abstract

This chapter presents the performance of neural network autoregressive with exogenous input (NNARX) model structure and evaluates the training data that provide robust model on fresh data set. The neural network type used is backpropagation neural network also known as multilayer perceptron (MLP). The system under test is a heat exchanger QAD Model BDT921. The real input-output data that collect from the heat exchanger will be used to compare with the model structure. The model was estimated by means of prediction error method with Levenberg-Marquardt algorithm for training neural networks. It is expected that the training data that covers the full operating condition will be the optimum training data. For each data, the model is randomly selected and the selection is based on ARX structure. It was validated by residual analysis and model fit, and validation results are presented and concluded. The simulation results show that the neural network system identification is able to identify good model of the heat exchanger.

Keywords: identification, heat exchanger, neural network, autoregressive with exogenous input, model

1. Introduction

A heat exchanger model QAD BDT921 that is installed in the control laboratory is being used as a model plant to achieve the digital control system design since it is analog in nature. The type of this equipment is a shell and tube heat exchanger, and the application of this model is to generate the heat using steam water. It is the most important equipment in manufacturing and industrial plant in order to maintain and control temperature. In this chapter, more concentration is on the process control and the application of the various methods which are used to develop a mathematical model for the plant. In order to verify the experimental data, a mathematical model is needed to describe the process. Based on the physical and chemical principles, the mathematical models can be derived. However, it is not always possible to develop a model theoretically because the experimental procedures produced by the manufacturer do not provide enough information on how to obtain a mathematical model of the plant. Thus, another useful way is by using system identification to build process models and estimate unknown model

parameters. The objective of system identification is to determine system equations from a given input and output time history. There is a few mathematical modeling that can be implemented to the heat exchanger. In this project, the identification of a nonlinear system has been focused on the neural network autoregressive with exogenous input (NNARX). It is one of the neural network system identification classes. It functions using certain syntax from MATLAB command window. Sampling time can be changed due to the system requirement. The heat exchanger is closed loop with controller using proportional integral derivative (PID) control system, while the main objective of this project is to obtain the real input-output data in open-loop system for system identification purposes. Any dynamic response of the process will depend on this controller setting. To open-loop the closed-loop system, the controller must be off by setting the parameter on the temperature controller TIC11. So, the process of heat transfer occurs between hot water from the T12 through the boiler and the cold water inlet (product liquid) at room temperature around 25.1°C. In order to obtain the modeling system of the heat exchanger, the models need mathematical models by using the system identification. Other forms of identification are available and also used in industry such as modeling systems by their physical properties, but it takes a lot of time compared to using system identification since it is only concerned with input and output signals from the system. Neural network processes information in a similar way the human brain does. The network is composed of many elements (neurons) that work in parallel to solve certain problem. The disadvantage is its unpredictable operation because the network figures out how to solve the problem by itself. In the way to complete this chapter, there are a few objectives that have to be achieved. The objectives are:

1. To collect data of heat exchanger QAD BDT921 through experiment design

2. To design NNARX based on multilayer perceptron neural network (MPNN)

3. To simulate NNARX using M-File MATLAB

4. To analyze the simulation resulted based on MSE and LMA

5. To identify mathematical modeling for heat exchanger using network autoregressive with exogenous input (NNARX)

The scopes of this chapter are:

1. Data input is cold water inlet (product liquid) at room temperature around 25.1°C, and the data output is product (water) temperature from range between 22.5 and 55.7°C.

2. NNARX designed are multilayer perceptron (MLP) as structure model, Akaike Final Prediction Error (AFPE) as model training for ARX, Levenberg-Marquardt (LMA) as model training for neural network, and mean square error (MSE) and residual analysis (auto- and cross-correlation) as model validation.

3. To running M-File MATLAB that will be simulation resulted that are data input output on workspace MATLAB, graph data input output, graph ze (model fit) and graph zv (model fit).

   4. The analysis based on discrete-time IDOPLY model, loss function, FPE, and MSE.

The expected result of this project is to obtain the data of heat exchanger QAD BDT 921 through experimental design and be able to design the NNARX based on multilayer perceptron neural network (MPNN). It will be simulated using M-File MATLAB. The simulation resulted will be analyzed based on discrete-time IDOPLY model, loss function, FPE, and MSE. Some papers make a study on heat exchangers for different purposes by using various methods, among others, discrete-time model [1, 2], modeling for intelligent control design [3], ARX model [4], digital control design [5–7], PID control design [8–10], parametric and nonparametric identification [11–13], data analysis [14], and NNARX model [15].

2. Heat exchanger

These are diagram of a typical shell and tube heat exchanger, showing the components (A) tubes, (B) tube sheets, (C) shell and shell-side nozzles, and (D) baffles identified in **Figure 1**.

The tubes are the basic component of the shell and tube exchanger, providing the heat transfer surface between one fluid flowing inside the tube and the other fluid flowing across the outside of the tubes. It is most commonly made of copper or steel alloys. The shell is simply the container for the shell-side fluid, and the nozzles are the inlet and exit ports. It normally has a circular cross-section. The heat exchanger is built of a single shelf pass and sixteen tube pass. Inside the shell there are a total of 23 baffles that serve to "mix" to shell-side fluid, and at any cross-section, its temperature tends to be uniformed. The product liquid will enter the shell and the hot water will enter the tube. Flow rate of the product liquid and hot water is the dependent factor of the heat transfer rate. The hot water is heated in the boiler at a constant temperature to supply the heat for heating up the liquid in a heat exchanger. Essentially, the plant consists of two main control loops, the first controls the water levels in the boiler drum and the second controls the liquid temperature. These two control loops are designed as a closed-loop control system. But in this project, the main objective is to obtain the open-loop system. Since the plant is in closed-loop system, the temperature controller (TIC11) needs to be off in order to obtain the open-loop system. So, the process of heat transfer occurs between hot water from the T12 through the boiler and the cold water inlet (product liquid) at room temperature around 25.1°C. As the most common heat transfer industrial equipment is the shell and tube heat exchanger, this equipment is used in QAD Model BDT921 to study temperature control of a heat transfer process. **Figure 2** shows the schematic of heat exchanger QAD BDT921.
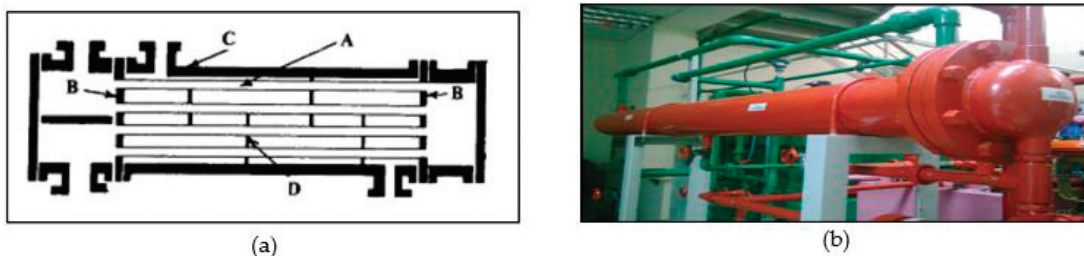

Figure 1.
(a) Diagram of a typical shell and tube heat exchanger. (b) The heat exchanger QAD BDT921.
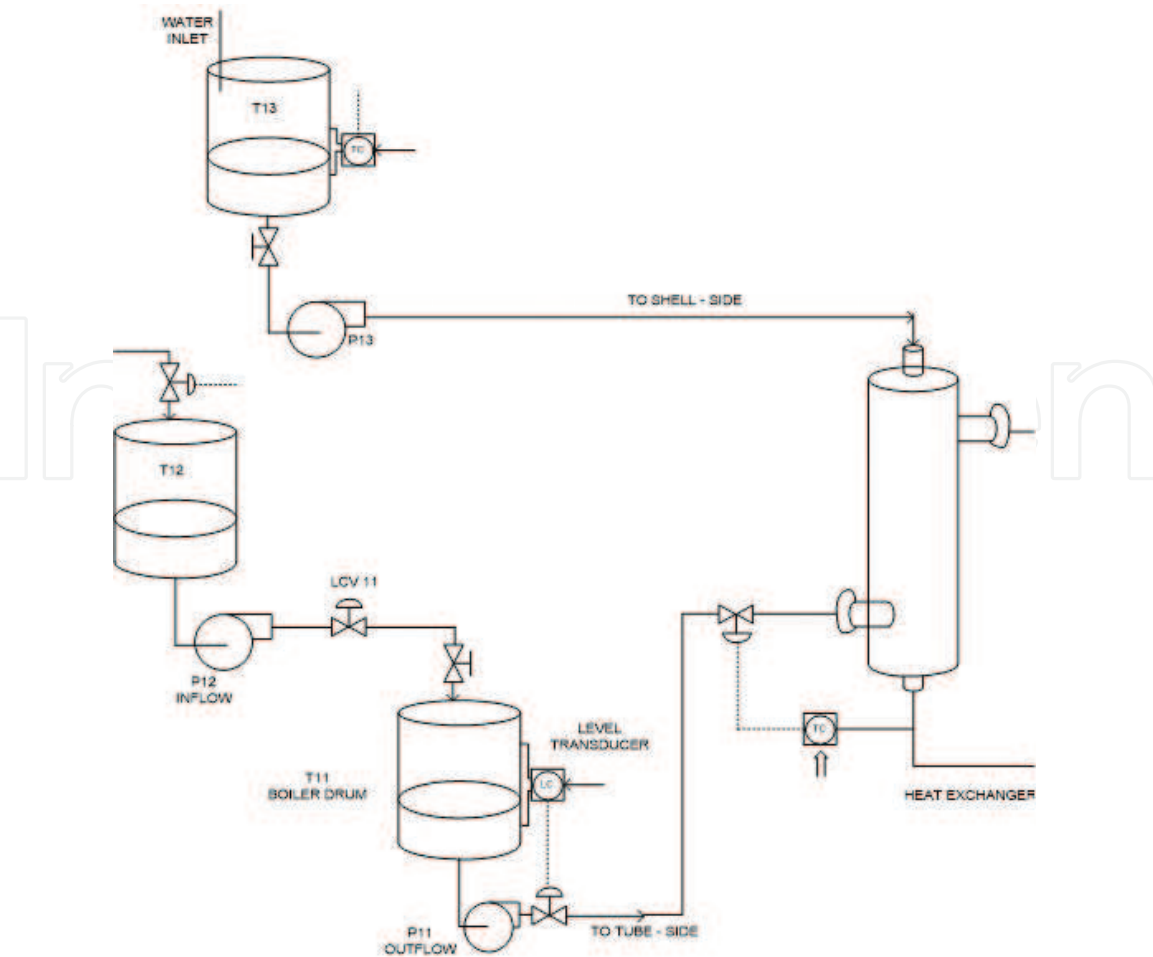
Figure 2.
Schematic diagram of heat exchanger QAD BDT921.

3. Experimental design

The system identification includes the steps such as execute the experiment of heat exchanger and collect input-output data from the process to be identified. The block diagram of open-loop control is shown in **Figure 3**. The data collected by an open-loop controller has the characteristic of not using feedback to determine if its input has achieved the desire goal. The system does not observe the output of the process it is controlling, and it computes its input into system using only the current state and its model of the system. When the data have been collected from the identification experiment, it is often necessary to do some processing on the data set prior to using it for identification. **Figure 4** shows the step of experimental setup in
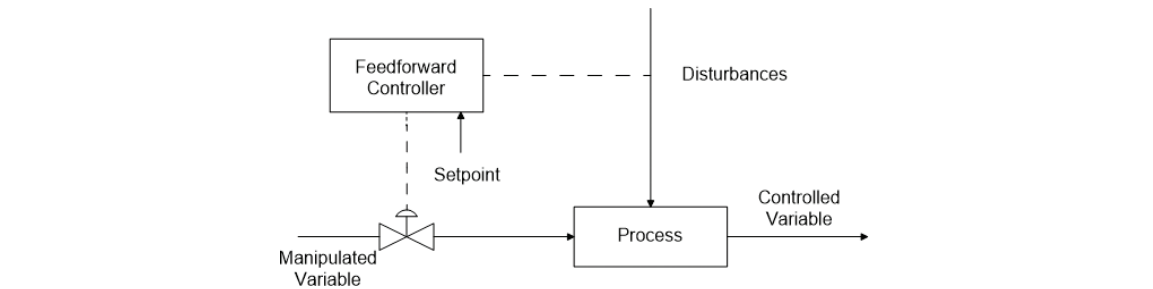

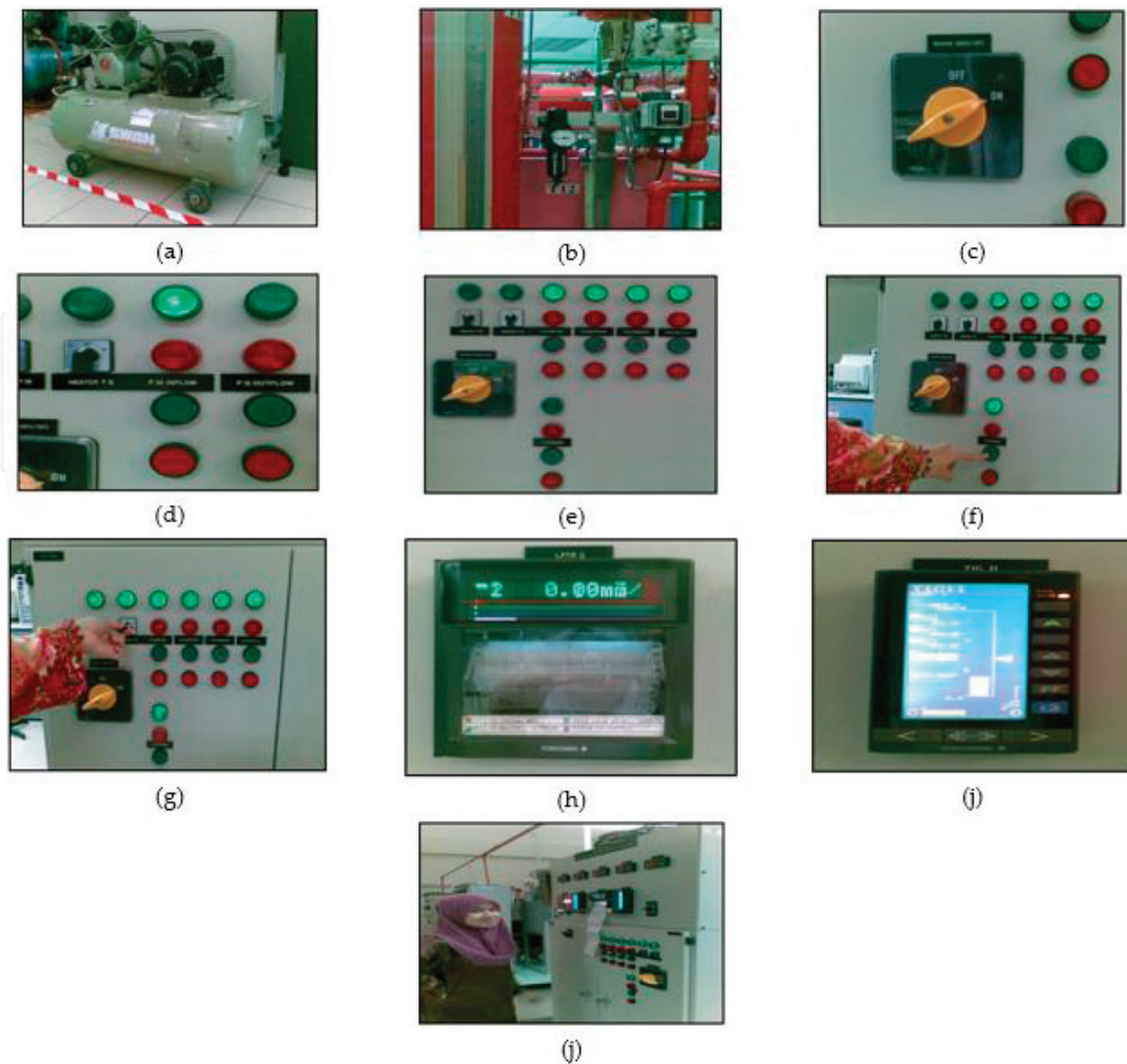**Figure 3.**
Feedforward control loop.

Figure 4.
(a) Start on compressor. (b) Supply AIS. (c) On main switch (415 V/3P). (d) On P12 inflow and P11 outflow. (e) P13 and P14 product. (f) On P15 mixing. (g) On heater T12 and T11. (h) Check recorder LFTR1. (i) Temperature controller TIC. (j) Output graph.

obtained open-loop system, and the detail of this experiment, in flowchart form, is shown in **Figure 5**.

The heat exchanger is in closed-loop system, but the main objective of this project is to record the data in open-loop system. To open-loop the closed-loop system, the controller must be in unity state by turning off the temperature sensor. It is off by setting the parameter Pb to 1, Ti to infinity, and Td to 0. The heat exchanger is a shell and tube type used for heat transfer. Preheated tank is used in the plant to supply sufficient hot water. Hot water is heated up in the preheated tank (T12) by a heater starting from zero which needs to be warmed up first. The hot water in the preheated tank T12 is pumped to boiler T11 by water pump P12. The temperature for cold water inlet (product liquid) in tank T13 is at room temperature which is around 25.1°C. It is pumped into the heat exchanger at different flow rates depending on the operator's selection. Operation may select on for water pumps P13 and P14 or either one depending on the flow rate required. The flow indicator FT13 indicates the flow rate of (cold water inlet) product liquid. The product liquid at room temperature is pumped into heat exchanger. The process of heat transfer occurs between hot water from the T12 through the boiler and the cold water inlet (product liquid). The amount of heat transferred in the heat exchanger varies with flow rates of those two liquids. In heat transfer process, the energy
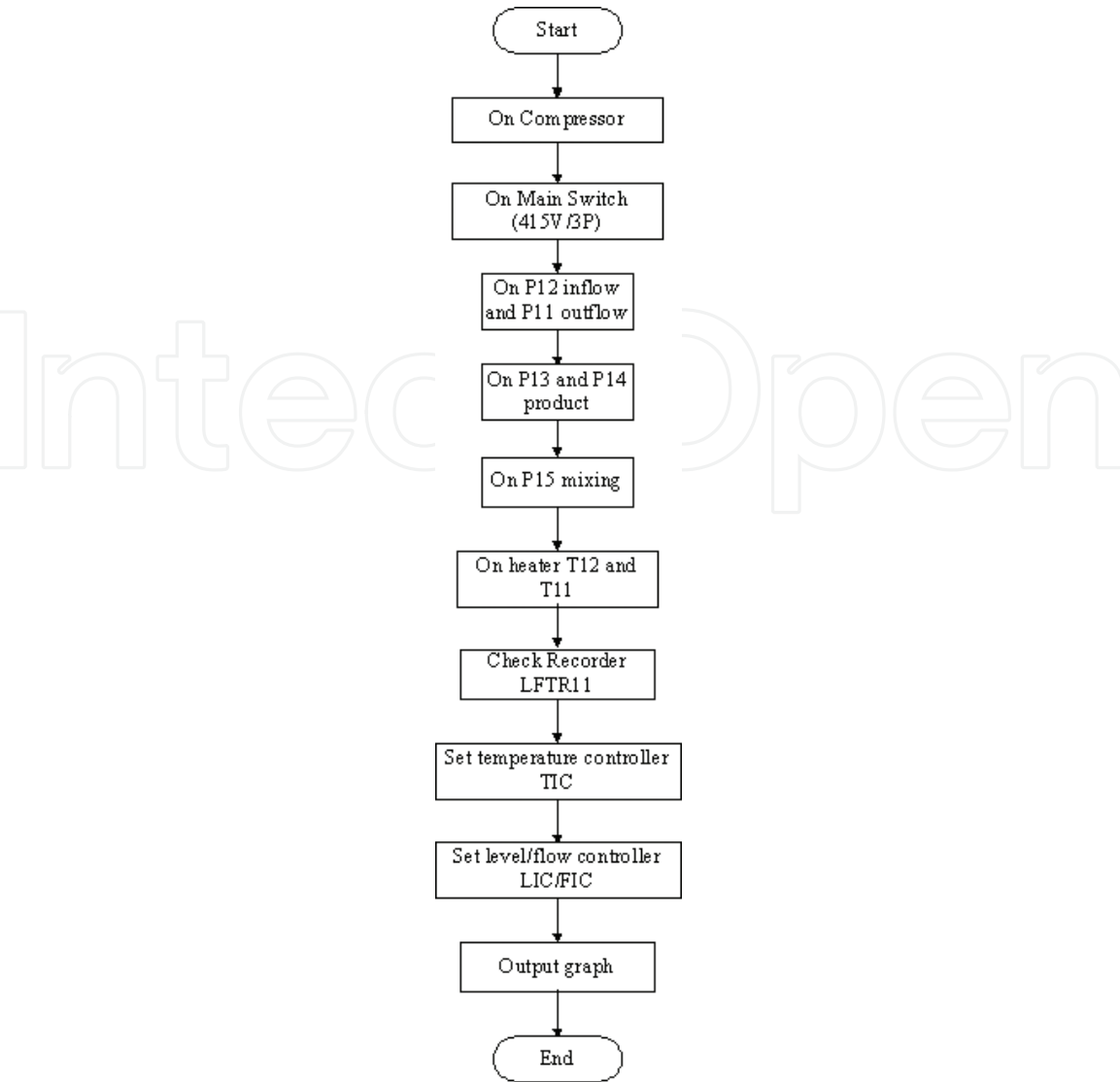
5

Figure 5.
Flowchart of experimental setup.

stored in one medium as heat capacity is transferred to another medium. The heat exchanger used in the plant is a device where heat is transferred from one fluid, across a tube to another fluid. When two fluids at different temperatures enter the heat exchanger, the temperature of the cold fluid will be increased. Water is used as the medium in the process control training system.

4. NNARX model identification

The most common neural architecture is the multilayer perceptron (MLP). An MLP is a feedforward network also known as a backpropagation neural network built up of perceptron-type neurons, arranged in layers. An MLP has an input layer, one or more hidden layers, and output layer. The feedforward neural network begins with an input layer that may be connected directly to the output layer or a hidden layer. When it is connected to a hidden layer, it can also be connected to another layer whether directly to the output layer or hidden layer. The number is not limited as long as there is at least one output or hidden layer provided. Commonly, there is one hidden layer used in most neural network, neural network that have more than two hidden layers been very rare. **Figure 6** illustrates a typical neural network with a single hidden layer.

In order to develop this system identification, the basic concept of the model structure and estimation method in this project will be discussed. The procedure of modeling and system identification is shown in **Figure** 7.

System identification is the experimental approach to process modeling. It is the process of deriving a mathematical model of a system based on observed input-output data. The procedure to determine a model of a dynamical system from observed input-output data involves three stages which are collecting the real input-output data, choosing a set of candidate model structure and a criterion to select a particular model in the set, and basing on the information in the data. Besides that, in solving this system identification process, a model structure is defined. There are a lot of model structure such as ARX (autoregressive with exogenous input) model, ARMAX (autoregressive moving average exogenous input) model, OE (output error) model, and BJ (Box-Jenkins) model. In this project, the candidate model structure that will be used is NNARX (neural network autoregressive with exogenous input model structure) which is the nonlinear model structures based on neural network. The network structure is defined as a multilayer perceptron.

A linear dynamic system can be represented by Eq. (1):
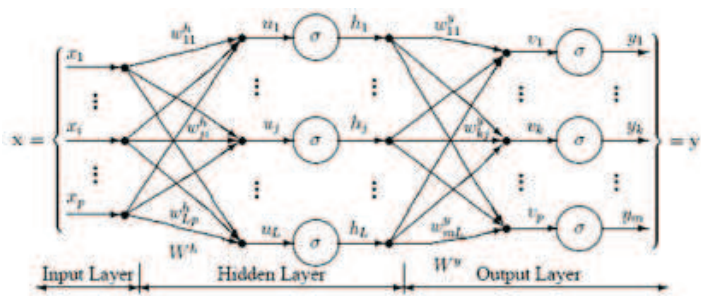
$$y(t) = q^{-nk} G(q) u(t) + H(q) \tag{1}$$

Figure 6.
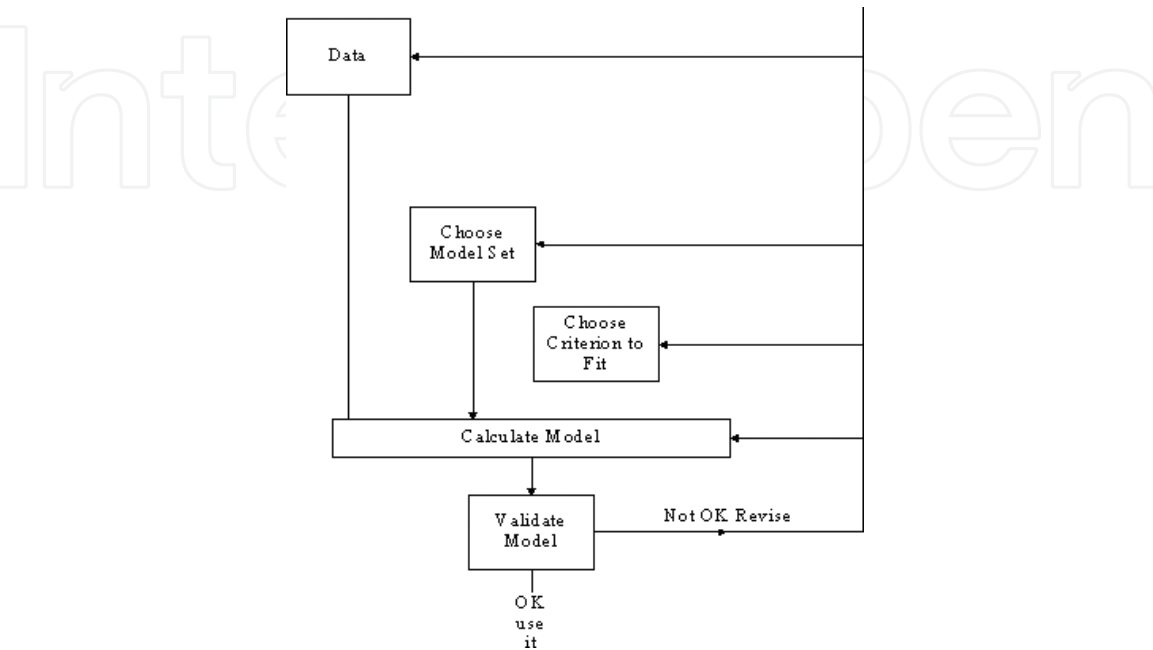*A typical feedforward neural network (single hidden layer).*

Figure 7.
The procedure of modeling and system identification.

where $q^{-nk}G(q)u(t)$ term refers to noise-free output and $H(q)e(t)$ refers to disturbance term. q is an argument of $G(q)$ and $H(q)$ is the negative shift operator, which is equivalent to q^{-1} represent by q^{-nk} and can be demonstrated by $q^{-1}x(t) = x(t-1)$. nk is the time delay in sampling instant between the process input and the output.

The ARX model structure is given by Eq. (2):

$$y(t) = q^{-nk}\frac{B(q)}{A(q)}u(t) + \frac{1}{A(q)}e(t) \qquad (2)$$

where the polynomials A(q) and B(q) are given by Eqs. (3) and (4).

$$A(q) = 1 + a_1 q^{-1} + \ldots + a_{na}q^{-na} \qquad (3)$$

$$B(q) = b_1 q^{-1} + \cdots + b_{nb}q^{-nb} \qquad (4)$$

In estimating the nonlinear counterpart of the ARX structure, the neural network can be utilized. According to the research done by M. Norgaard, O. Ravn, N. K. Poulsen, and L. K. Hansen, the multilayer perceptron (MLP) is one of the most popular neural network structures especially in the identification of a nonlinear system. The neural network version of ARX model structure is denoted as ARX (NNARX). Assuming the input delay $nk = 1$, the general NNARX model structure is shown in **Figure 8**.

The input-output relationship of NNARX model structure can be represented by Eq. (5):

$$y(t) = g[\varphi(t) + e(t)] \qquad (5)$$

The one-step-ahead prediction (1-SAP) of the NNARX model structure is given by Eq. (6):

$$\acute{y}(t|\theta) = g[\varphi(t), \theta] \qquad (6)$$

where $\varphi(t)$ is the regression vector, θ is the parameter vector, g is the function realized by the neural network, $e(t)$ is the noise, $y(t)$ is the system output, and $\acute{y}(t|\theta)$ is the predicted output based on the parameter vector θ.

In this work, the network training algorithm is done by using the prediction error method. The prediction error is given by Eq. (7):

$$\varepsilon(t,\theta) = y(t) - \acute{y}(t|\theta) \qquad (7)$$

where $y(t)$ is the observed output, $\acute{y}(t|\theta)$ is the predicted output, and θ is the estimated parameter.
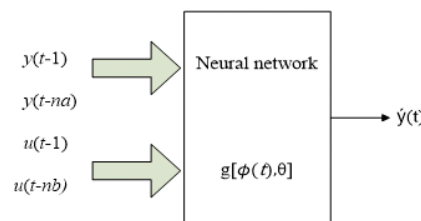

Figure 8.
The general NNARX model structure.

The measurement of the prediction error is often represented by a function known as the loss function. Its general representation can be written by Eq. (8):

$$V_N = \frac{1}{2N}\sum_{t=1}^{N}\varepsilon^2(t,\theta) \tag{8}$$

where Z^N refers to the training data set. In this project, the network training algorithms used Levenberg-Marquardt training and allow up to 100 epochs to train the network.

The value for this Akaike Final Prediction Error using the Eq. (9).

$$\text{FPE} = \frac{1+n/N}{1+n/N} \cdot V_N \tag{9}$$

where n is the total number of estimated parameters and N is the length of the data record. This criterion reflecting the prediction error variance says that in collection of different models, the one with the smallest value of FPE should be chosen. Once a model structure has been identified, it is important to validate the model using a data set. Model validation is needed to verify that the identified model fulfills the modeling requirement according to the subjective and objective of good model approximation.

The model validation algorithm used in this identification is mean square error method and residual analysis. The average squared error is shown by Eq. (10):

$$\text{MSE} = \frac{1}{N}\sum_{t=1}^{N}\varepsilon^2(t) \tag{10}$$

This is a measure in a single positive number of how well the model output fits the measured data.

It is good to check the autocorrelation of the residual analysis by Eq. (11) as it is assumed to be a white noise sequence:

$$R_\varepsilon^N = \frac{1}{N}\sum_{t=1}^{Nn}\varepsilon(t)\,\varepsilon(1-\tau) \tag{11}$$

If these numbers are not small for $\tau \neq 0$, then part of ε could have been predicted from past data, and so this is a sign of deficiency in the model. Similarly the residuals should not be correlated with the input, so it is also good to check the cross-correlation of the residuals and the input by Eq. (12):

$$R_{\varepsilon u}^N = \frac{1}{N}\sum_{t=1}^{Nn}\varepsilon(t)\,u(t-\tau) \tag{12}$$

5. Result and analysis

In this part, the result for data collection and data prefiltering or data preprocessing will be displayed. The 1000 sample data will be collected by sampling the output graph of the heat exchanger QAD BDT921. **Figure 9** shows an example part of data on the graph.

In this chapter, the important part is data collection. This is because, in this project, the model will be tested based on the real data obtained from the heat exchanger process. In this experiment, the input and output data are temperature. About 1000 sample data were collected from the graph. **Table 1** shows example of 20 data for input and output.
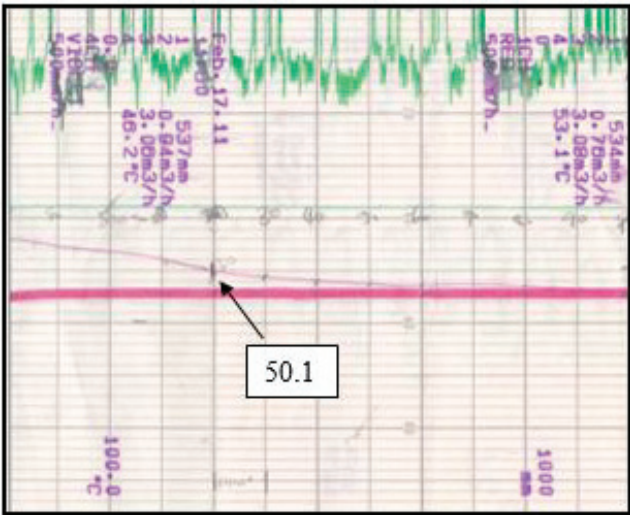
Figure 9.
Example of data on the graph.

Table 1.
The 20 data for input and output of the heat exchanger QAD BDT921.

After data collection has been done, the next step is data prefiltering or data preprocessing. The input and output data obtained from the experiment are plotted using MATLAB software. **Figure 10** shows the results for input and output data (**Figure 11**).

Several types of parameters are used to create system identification with M-file. The parameters are listed as time sampling, orders, and layer size for neural network. Those parameters [Ts] [na nb nk] [layer size] have been analyzed with M-file and also the effect of this parameters. The values (0.072, 0.72) for the number of units in time sampling with the same orders [1 1 1] were tested whether the model fit improves for the data estimation and validation. The effect of time sampling on model fit analyzed with M-file is shown in **Figures 12** and **13**.

One of the important parameters is orders. For orders, the value of [na nb nk] will be set randomly. In this case, the examples of selected orders are [1 1 1] and [1 2 1] with the same layer sizes [2 1] will be tested whether the model fit improves for the estimation and validation data. **Figures 14** and **15** show the effect of orders [1 1 1] on model fit for the data estimation and validation.
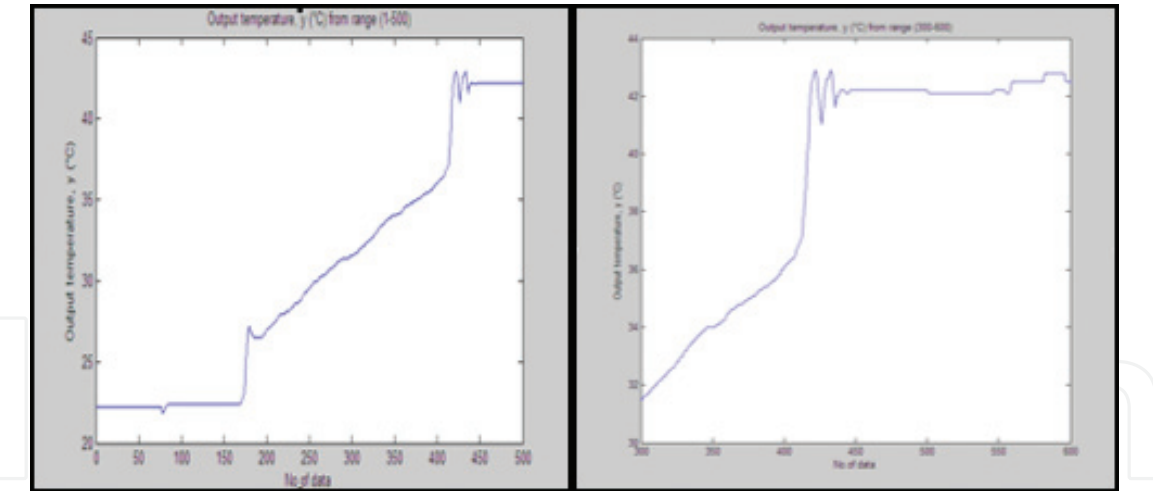
Figure 10.
Output temperature from range (1–500) and (300–600).
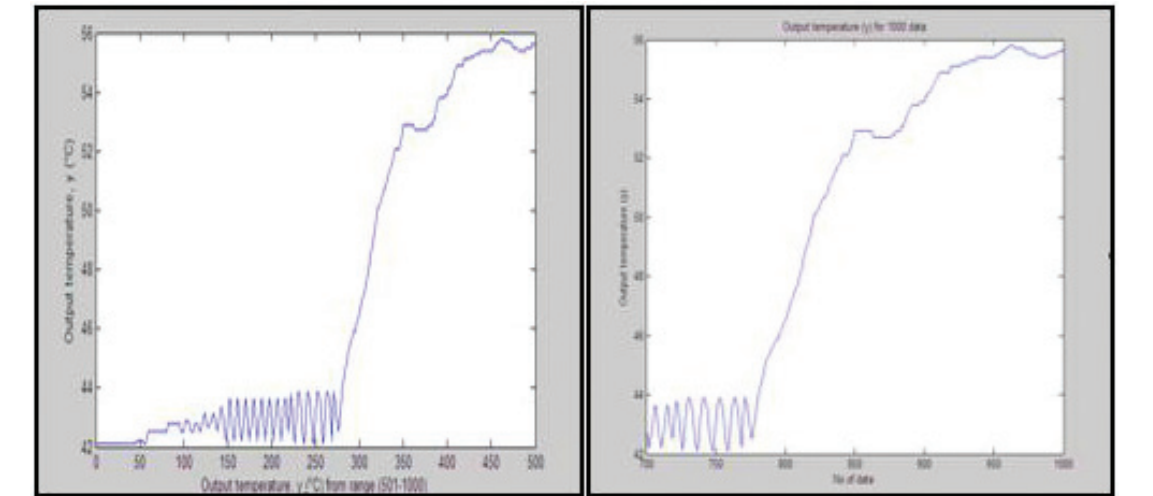
Figure 11.
Output temperature from range (501–1000) and (700–1000).

Data Estimation   Data Validation

Figure 12.
1-SAP by the model against the data estimation and validation for Ts = 0.072s and [0.072][1 1 1][2 1].

In this case, the neurons' values (2, 4, 8) for the number of units in the hidden layer will be tested to see if the model fit improves for the estimation and validation data. The model used one past value of the input and one past value of the output

Figure 13.
1-SAP by the model against the data estimation and validation for Ts = 0.72s and [0.072][1 1 1][2 1].
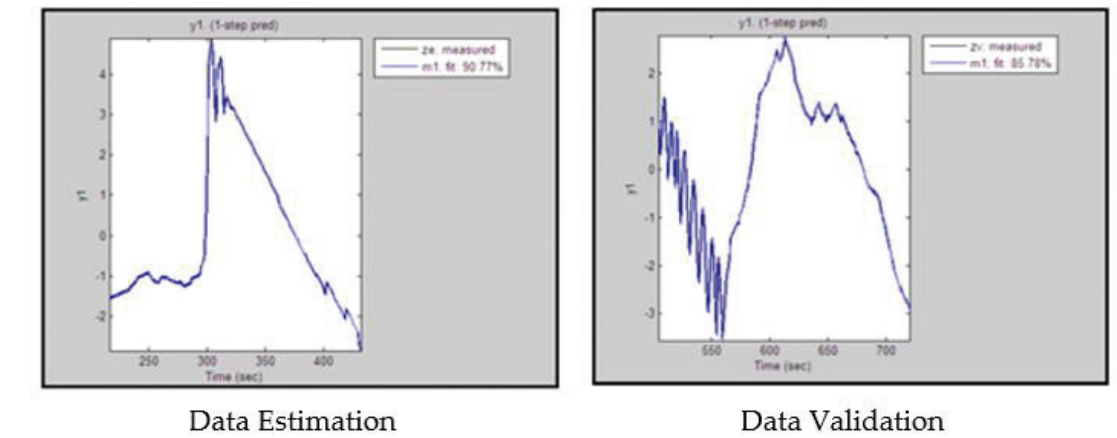
Figure 14.
1-SAP by the model against the data estimation and validation for [0.072][1 1 1][2 1] at [1 1 1].
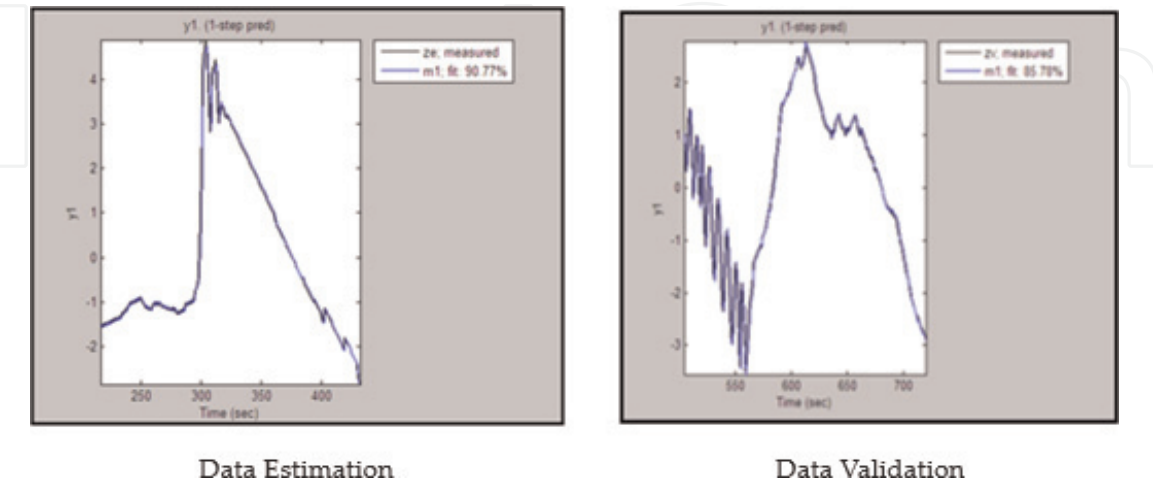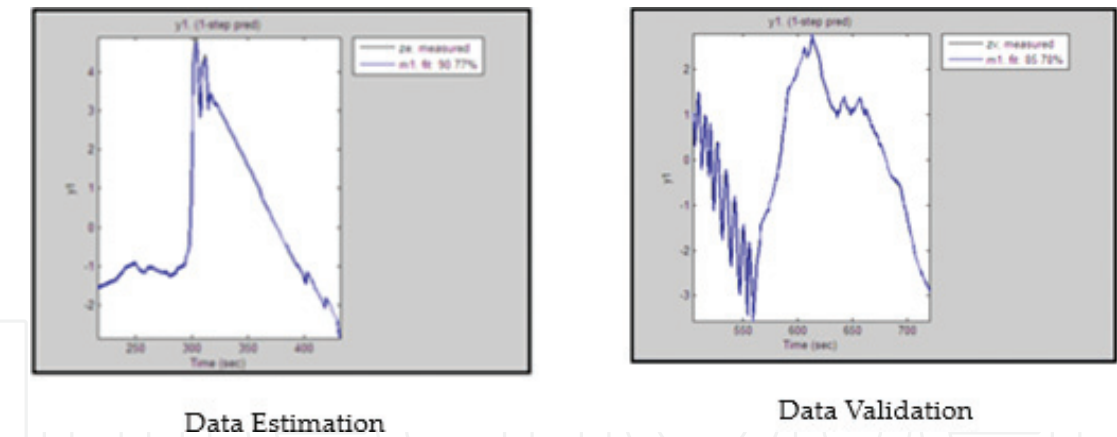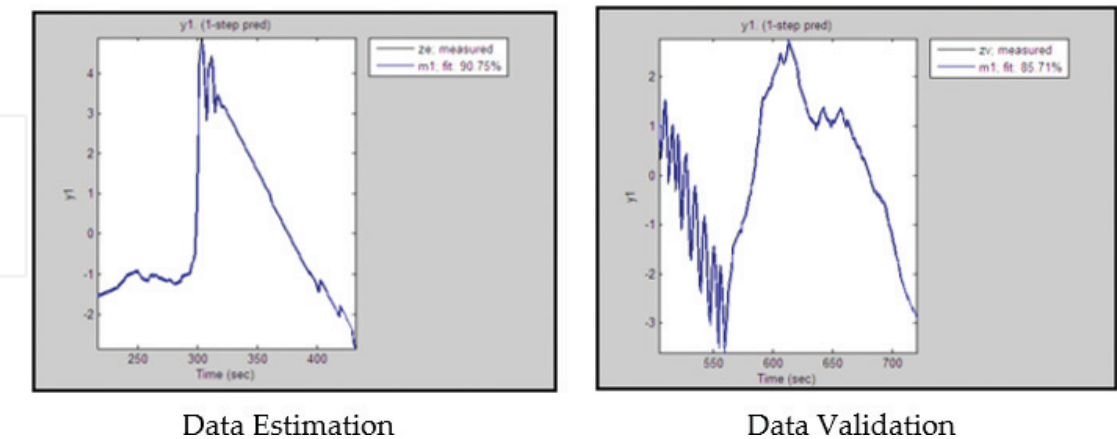

Figure 15.
1-SAP by the model against the data estimation and validation for [0.072][1 1 1][2 1] at [1 2 1].

[1 1 1]. **Figure 16** shows the one-step-ahead prediction results of optimized NNARX model for estimation data at orders [1 1 1] with 2, 4, and 8 hidden units.

Figure 17 shows the one-step-ahead prediction results of optimized NNARX model for validation data at orders [1 1 1] with 2, 4, and 8 hidden units.

Table 2 shows the effect time sampling (Ts) on model fit for NNARX model based on **Figures 12** and **13**. At 0.72 second, the model fit for estimation data (ze) is 90.96% and for validation data (zv) is 87.64%, while at 7.2 second the model fit is

Figure 16.
Estimation plot for model with 2, 4, and 8 hidden units for [0.072] [1 1 1].

Figure 17.
Validation plot for model with 2, 4, and 8 hidden units for [0.072] [1 1 1].

Ts (second)	Order [na nb nk]	Layer	Model fit (%)	
			ze	zv
0.072	[1 1 1]	[2 1]	90.96	87.64
0.72	[1 1 1]	[2 1]	90.96	87.64

Table 2.
Effect of time sampling on model fit.

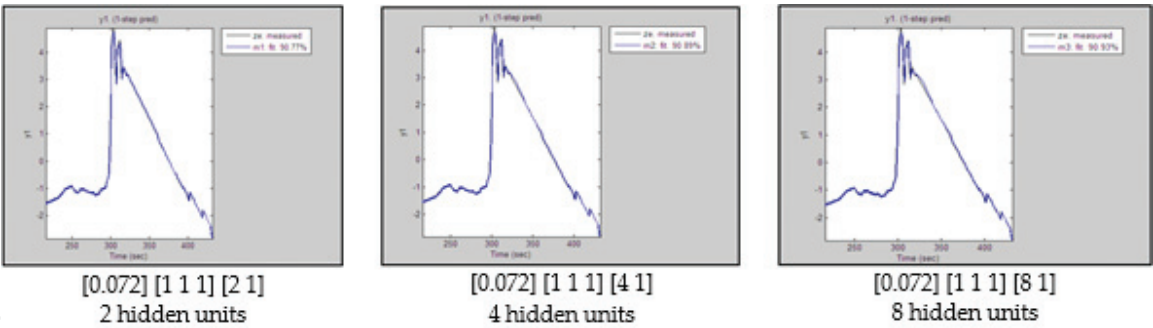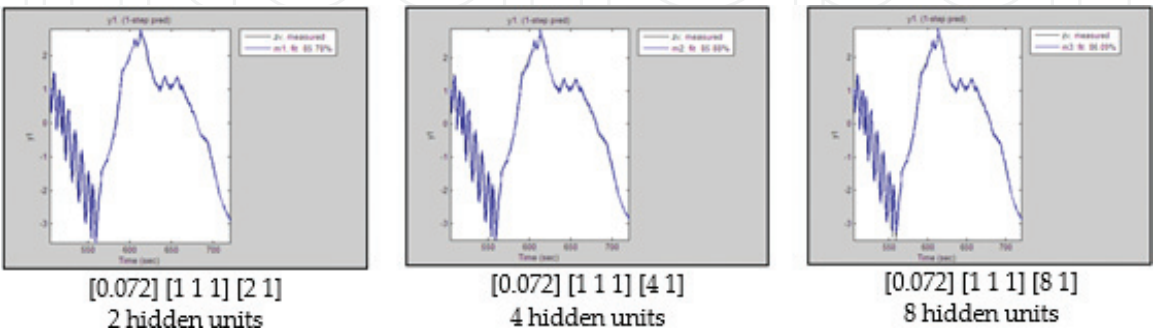also the same for estimation data (ze) and validation data (zv). These results clearly indicate that sampling time did not effect on the model fit. It is the number of samples per second taken from a continuous signal to make it discrete, and holding time is the time between two samples.

Based on **Figures 14** and **15**, the best orders turn out to be na = 1, nb = 2, and nk = 2. The higher order [1 2 2] ARX model is able to reproduce the validation data best, but the differences between the orders are really minor. **Table 3** shows the comparisons between ze and zv at sampling time, Ts = 7.2 second, and layer size [2 1].

The effect of the number of units in the hidden layer on model fit for NNARX model is shown in **Table 4**, where the fit for the validation data with different numbers of units in the hidden layer more accurate. Validation plot for model with two neurons in the hidden layer has a model fit of about 85.78%, while for model with four neurons the model fit is 85.88%, and for model with eight neurons the model fit is 86.06%. Out of the three models found, the one with eight hidden units provides the best fit for the validation data. In this case, the model fit for each hidden units did not improve for estimation data.

The MATLAB result for least square estimates of ARX model at orders [1 1 1] and [3 2 2] is shown in **Figure 18**. It could be seen in **Figure 18** that the results for discrete-time IDPOLY are $A(q) = 1 - 0.9969q^{-1}$ and $B(q) = 1.717\text{e}011q^{-1}$. The loss

Ts (second)	Order [na nb nk]	Layer	Model fit (%)	
			ze	zv
0.072	[1 1 1]	[2 1]	90.77	85.78
0.72	[1 2 2]	[2 1]	90.76	85.81

Table 3.
Effect of orders on model fit.

Orders [nn]			Model fit %					
			Hidden layer [2 1]		Hidden layer [4 1]		Hidden layer [8 1]	
na	nb	nk	ze	zv	ze	zv	ze	zv
1	1	1	90.77	85.78	90.81	85.88	90.77	86.06

Table 4.
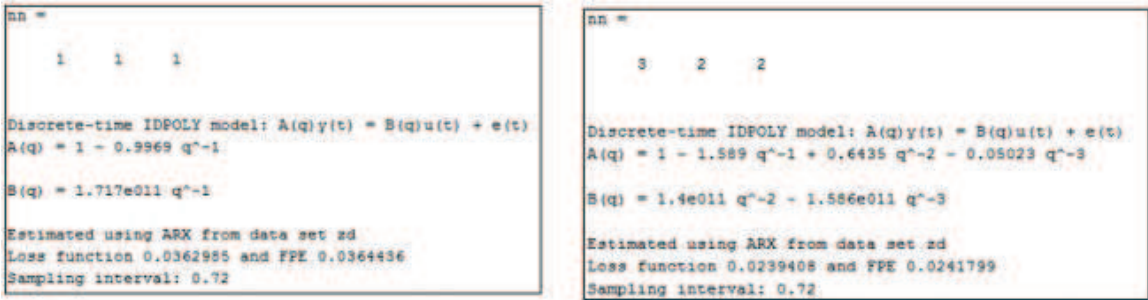*Effect of the number of units in the hidden layer on model fit.*

Figure 18.
Least square estimates of ARX model with orders [1 1 1] and [3 2 2].

function for this model is equal to 0.0362985, and the final prediction error is equal to 0.0364436. The value for this Akaike Final Prediction Error has been calculated by MATLAB using the equation of FPE = $V^*(1 + d/N)/(1 − d/N)$ where V is the loss function of the model, d is the number of estimated parameters, and N is the number of estimation data. The best selected model is at orders [3 2 2] because it has smallest value loss function and FPE. **Figure 18** shows that the loss function at these orders is 0.0239408 while for the FPE is 0.0241799, and the result for the discrete-time IDPOLY model is $A(q) = 1 − 1.589q^{-1} + 0.6435q^{-2} − 0.05023q^{-3}$ and $B(q) = 1.4e011q^{-2} − 1.586e011q^{-3}$. Although this gives the best result, the suitable model will be decided until the validation process will be done.

Table 5 shows the results for estimated parameter using ARX model from data (zd) at several orders consisting of IDPOLY model, loss function, and FPE. From the table, it can be seen that when the model order selection is low, then the model will have high loss function and FPE. The best estimated parameter is obtained from the smallest value of loss function and FPE. The best selected model is also at orders [3 2 2] and layer size [8 1] because the estimated results using NNARX with nonlinearity estimator (neuralnet) for loss function have a smallest value at these orders. **Table 5** shows the estimated results using ARX from data set zd for IDPOLY model, loss function, and FPE for various orders.

Results for nonlinear regressors are estimated using NNARX with nonlinearity estimator (neuralnet) presented in **Table 6**.

The mean square error was used to evaluate the performance of each neural network. **Table 7** shows the mean square error (MSE) performance for training.

Order [na nb nk]	IDPOLY model	Loss function	FPE
1 1 1	$A(q) = 1 − 0.9969q^{−1}$ $B(q) = 1.717e011q^{−1}$	0.0362985	0.0364436
1 2 1	$A(q) = 1 − 0.9969q^{−1}$ $B(q) = −2.066e011q^{−1} + 2.72e011q^{−2}$	0.0357804	0.0359949
1 2 2	$A(q) = 1 − 0.9969q^{−1}$ $B(q) = 5.001e010q^{−2} − 3.074e010q^{−3}$	0.0354304	0.0356427
2 1 1	$A(q) = 1 − 1.561q^{−1} + 0.5661q^{−2}$ $B(q) = 4.573e010q^{−1}$	0.0242571	0.0244025
2 1 2	$A(q) = 1 − 1.561q^{−1} + 0.5655q^{−2}$ $B(q) = 2.155e010q^{−2}$	0.0242772	0.0244227
2 2 1	$A(q) = 1 − 1.566q^{−1} + 0.5707q^{−2}$ $B(q) = 1.293e011q^{−1} − 7.543e010q^{−2}$	0.0242369	0.0244306
2 2 2	$A(q) = 1 − 1.56q^{−1} + 0.5651q^{−2}$ $B(q) = 1.637e011q^{−2} − 1.814e011q^{−3}$	0.0240013	0.0241931
3 1 1	$A(q) = 1 − 1.589q^{−1} + 0.6556q^{−2} − 0.06161q^{−3}$ $B(q) = 1.359e010q^{−1}$	0.0240461	0.0242383
3 1 2	$A(q) = 1 − 1.59q^{−1} + 0.656q^{−2} − 0.06091q^{−3}$ $B(q) = −3.576e010q^{−2}$	0.0240302	0.0242223
3 2 2	$A(q) = 1 − 1.589q^{−1} + 0.6435q^{−2} − 0.05023q^{−3}$ $B(q) = 1.4e011q^{−2} − 1.586e011q^{−3}$	0.0239408	0.0241799

Table 5.
Results for estimated using ARX from data set zd.

Order [na nb nk]	Regressors	Loss function		
		Layer sizes [2 1]	Layer sizes [4 1]	Layer sizes [8 1]
1 1 1	$y1(t − 1)$ $u1(t − 1)$	0.029315	0.0269315	0.28331
1 2 1	$y1(t − 1)$ $u1(t − 1)$ $u1(t − 2)$	0.029486	0.028948	0.020861
1 2 2	$y1(t − 1)$ $u1(t − 1)$ $u1(t − 3)$	0.029567	0.029033	0.027956
2 1 1	$y1(t − 1)$ $y1(t − 2)$ $u1(t − 1)$	0.0192380	0.010579	0.0077851
2 1 2	$y1(t − 1)$ $y1(t − 2)$ $u1(t − 2)$	0.0192380	0.010579	0.0077851
2 2 1	$y1(t − 1)$ $y1(t − 2)$ $u1(t − 1)$ $u1(t − 2)$	0.0212208	0.011918	0.0049208
2 2 2	$y1(t − 1)$ $y1(t − 2)$ $u1(t − 2)$ $u1(t − 3)$	0.021253	0.011954	0.0048163
3 1 1	$y1(t − 1)$ $y1(t − 2)$	0.016355	0.0065896	0.0046813

Order [na nb nk]	Regressors	Loss function		
		Layer sizes [2 1]	Layer sizes [4 1]	Layer sizes [8 1]
	y1(t − 3) u1(t − 1)			
3 1 2	y1(t − 1) y1(t − 2) y1(t − 3) u1(t − 2)	0.016355	0.0065896	0.0046813
3 2 2	y1(t − 1) y1(t − 2) y1(t − 3) u1(t − 2) u1(t − 3)	0.016355	0.0063295	0.0045067

Table 6.
Results for nonlinear regressors estimated using NNARX with nonlinearity estimator (neuralnet).

Order [na nb nk]	Mean square error (MSE)		
	Layer sizes [2 1]	Layer sizes [4 1]	Layer sizes [8 1]
1 1 1	0.0292	0.0285	0.0292
1 2 1	0.0293	0.0293	0.0266
1 2 2	0.0293	0.0293	0.0285
2 1 1	0.0105	0.0105	0.0101
2 1 2	0.0105	0.0105	0.0101
2 2 1	0.0211	0.0118	0.0063
2 2 2	0.0210	0.0118	0.0049
3 1 1	0.0162	0.007	0.0047
3 1 2	0.0162	0.007	0.0047
3 2 2	0.0162	0.0064	0.0045

Table 7.
MSE performance for model structures with nonlinearity estimator (neural network).
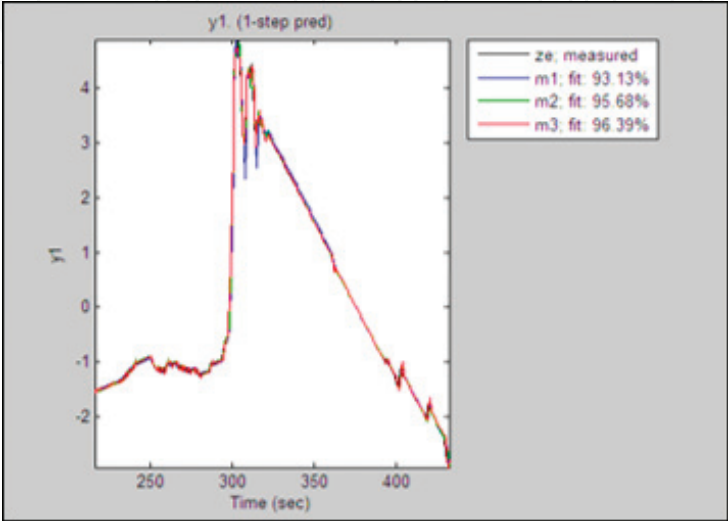

Figure 19.
1-SAP by the model against the estimation data of layer sizes (2, 4, and 8).

From observation, the model has smallest MSE when the model order selection is high. The best model which has the smallest value of MSE of about 0.0045 has order [3 2 2] and at layer size [8 1].

Figure 19 shows the comparison between the system and the model output at order [3 2 2]. The output model trend is similar with the output system. The model fit is resultant of three types of layer sizes such as 2, 4, and 8 in the hidden layer. It can be observed that the best fit for estimation data is at layer (8 1). The model fit is 96.39%.

Based on all the results from the simulation graph, it generally can be said that the model fit is satisfactory in all conditions of training and testing with percentage fit between 90 and 97%. **Table 8** shows the orders consisting of all combinations na, nb, and nk in the range 1–3 and hidden layer effect of the model fit for data estimation. The best identified model has na = 3, nb = 2, and nk = 2 at layer size [8 1] where the m1 is 96.39%. The medium model turns out to be na = 3, nb = 1, and

Time	Orders [nn]			Model fit %		
				Layer size [2 1]	Layer size [4 1]	Layer size [8 1]
	na	nb	nk	m1	m2	m3
Ts = 0.072	1	1	1	90.77	90.81	90.77
	1	2	1	90.75	90.76	91.19
	1	2	2	90.76	90.76	90.87
	2	1	1	95.23	94.46	94.58
	2	1	2	92.53	94.46	94.58
	2	2	1	92.16	94.12	95.71
	2	2	2	92.16	94.12	96.21
	3	1	1	93.13	95.47	96.3
	3	1	2	93.13	95.47	96.3
	3	2	2	93.13	95.68	96.39

Table 8.
Comparisons between several different orders and layer sizes, based on the fit between estimation data and simulated output.
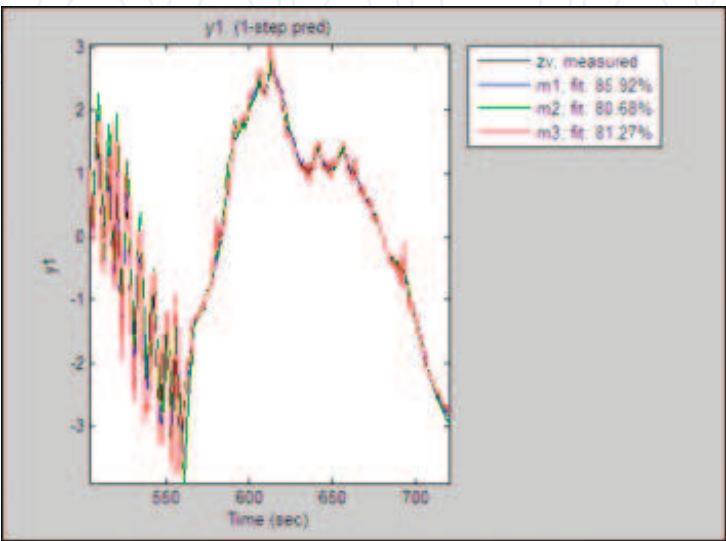

Figure 20.
1-SAP by the model against the validation data of layer sizes (2, 4, and 8).

nk = 2 at layer size [8 1] where the m2 is 96.3%, and the lower model turns out to be na = 1, nb = 2, and nk = 1 at layer size [2 1] where the m3 is 90.75%. **Table 8** shows the comparisons between several different orders and layer sizes, based on the fit between estimation data and simulated output.

The purpose of model validation is to verify that the identified model fulfills the modeling requirements according to subjective and objective criteria of good model approximation. It is important to validate the model using a data set from range (700–1000). **Figure 20** shows the resultant for validation data of three layer sizes at [3 2 2] on model fit.

Time	Orders [nn]			Model fit %		
				Layer size [2 1]	Layer size [4 1]	Layer size [8 1]
	na	nb	nk	zv	zv	zv
Ts = 0.072	1	1	1	85.78	85.88	86.06
	1	2	1	85.71	85.96	85.77
	1	2	2	85.81	86.04	86.23
	2	1	1	87.14	76.86	82.18
	2	1	2	87.13	76.86	82.18
	2	2	1	88.55	72.89	59.69
	2	2	2	88.55	72.54	75.75
	3	1	1	85.95	71	74.4
	3	1	2	85.95	71	74.4
	3	2	2	85.92	80.68	81.27

Table 9.
Comparisons between several different orders and layer sizes, based on the fit between validation data and validation output.
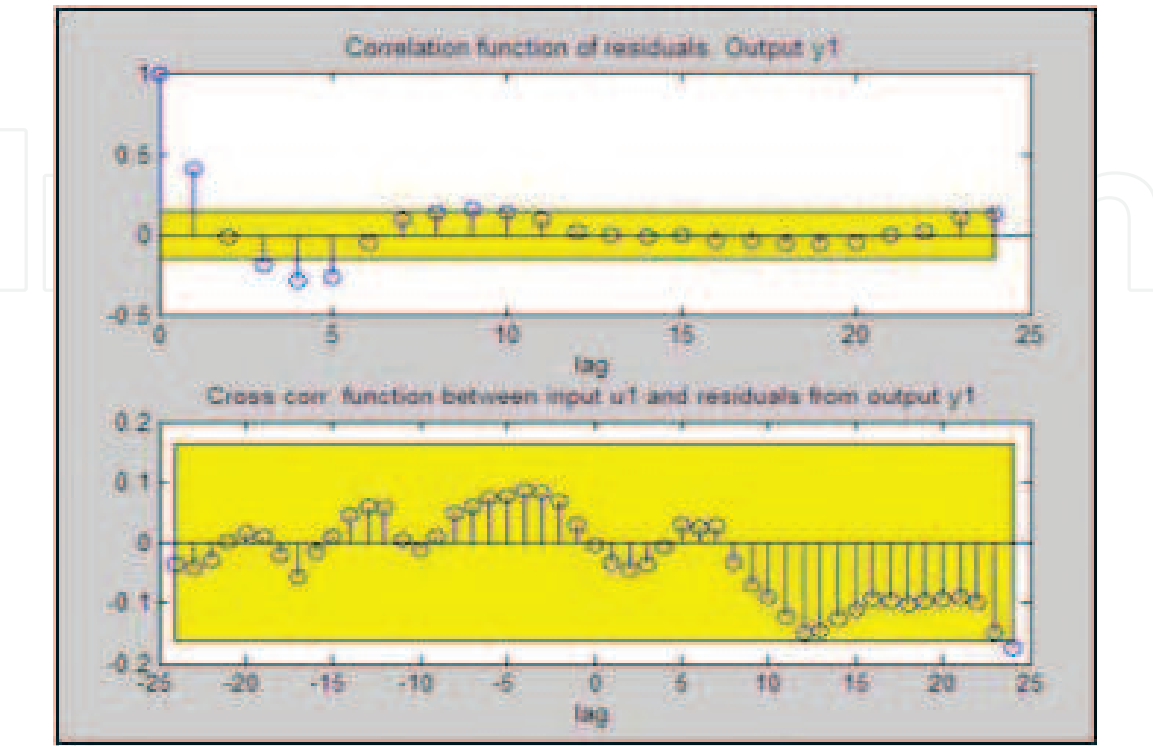

Figure 21.
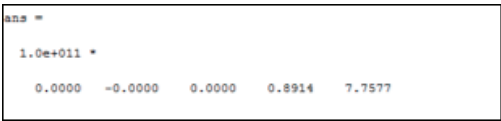*MATLAB result for residual analysis.*

Figure 22.
Weight and bias value at orders [3 2 2].

Based on all the results from the simulation graph, it generally can be said that the model fit is satisfactory in all conditions of training and testing with percentage fit between 60 and 90%. **Table 9** shows the orders consisting of all combinations na, nb, and nk in the range 1–3 and hidden layer effect of the model fit for data validation. The best identified model has na = 2, nb = 2, and nk = 2 at layer size [2 1] where the m1 is 88.55%. The medium model turns out to be na = 3, nb = 2, and nk = 2 at layer size [2 1] where the m1 is 85.95%, and the lower model turns out to be na = 2, nb = 2, and nk = 1 at layer size [8 1] where the m3 is 59.69%. **Table 9** shows the comparisons between several different orders and layer sizes, based on the fit between validation data and simulated output.

This percentage of model fit cannot be used to assume that the model is good. It can be proven when the validation check of this model is done with the residual analysis. The result of residual analysis for this model on validation data is shown in **Figure 21**. For a good model, the cross-correlation between residuals and input does not go significantly outside the confidence region. From the figure, the ARX model has satisfactory correlations. The samples are between the confidence interval.

The weight and bias value at orders [3 2 2] with two neuron in hidden layer [2 1] is shown in **Figure 22**.

6. Conclusion

This chapter has presented the capability of regularized NNARX model structure in representing the system. The nonlinear model of heat exchanger system can be identified by using system identification algorithm. The result shows that the estimated model has a nearly similar trend of output with the output signal of the system based on the percentage of model fit. The major influence to maximize the performance of the model is the selection of proper training data. Overall, with proper training data of regularized NNARX, it is capable in representing the system under test. In conclusion, the result of the best NNARX model of heat exchanger QAD Model BDT921 is na = 3, nb = 2, and nk = 2 with two neurons in hidden layer [2 1]. The model can be written down as:

$$y(t) = 0.8914u(t - 1) + 7.7577u(t - 3)$$

$$Loss\ function = 0.016355$$

$$MSE = 0.0162$$

$$ze = 93.13\%$$

$$zv = 85.92\%$$

Author details

Tatang Mulyana
Telkom University, Bandung, Indonesia

*Address all correspondence to: tatangmulyana@telkomuniversity.ac.id

IntechOpen

References

[1] Tatang M, Mohd Nor MT, Dirman H. A discrete time model of boiler drum and heat exchanger QAD Model BDT 921. In: International Conference on Instrumentation, Control & Automation ICA2009; 20-22 October 2009; Bandung, Indonesia. 2009. pp. 154-159

[2] Tatang M, Mohd Nor MT, Dirman H. A discrete time model of four heat exchanger types HE 158C. In: 7th International Conference on Heat Transfer, Fluid Mechanics and Thermodynamics (HEFAT2010); 19-21 July 2010; Antalya, Turkey. 2010

[3] Dirman H et al. Heat exchanger's shell and tube modelling for intelligent control design. In: International Conference on Computer and Communication Devices (ICCCD 2011), V2-37-41. 2011

[4] Tatang M, Farah Najaa S, Dirman H, Mohamad Nor MT. ARX Model of Four Types Heat Exchanger Identification, MUiCET; 2011

[5] Tatang M, Mohd Nor MT, Dirman H, Afrodi A. Digital Control Design for the Boiler Drum, UTHM FKEE Compilation of Papers; 2009

[6] Tatang M, Mohd Nor MT, Dirman H, Afrodi A. Digital control design for the boiler drum BDT921. In: Proceeding of Conf. on Industrial and Appl. Math.; Indonesia. 2010. pp. 257-260

[7] Tatang M, Mohamad Nor MT, Dirman H, Afrodi A. Digital control design for the boiler drum BDT921. Journal of Mechanics Engineering and Automation. 2011;**1**:392-397

[8] Tatang M, Mohd Nor MT, Dirman H, Hairul A. Modeling and simulation of temperature control PID single loop of a heat exchanger process control training system QAD model BDT921. In:

International Conference Electrical Energy and Industrial Electronic Systems (EEIES); 7-8 December 2009; Park Royal Penang, Malaysia. Re: 73-85976. 2009

[9] Tatang M, Hairul A. Modeling and Simulation of Temperature Control PID Single Loop of a Heat Exchanger Process Control Training System QAD Model BDT921, UTHM FKEE Compilation of Papers. 2009. pp. 227-236

[10] Tatang M, Mohd Nor MT, Dirman H, Muhaimin I. Simulation of various PI controllers setting for boiler drums T11 QAD MODEL BDT 921. In: IndoMS International Conference on Mathematics and its Applications 2009 UGM, 034/Mulyana/IICMA/2009; 12-13 October, 2009; Yogyakarta, Indonesia

[11] Tatang M. Parametric and Nonparametric Identification of Shell and Tube Heat Exchanger Mathematical Model [A thesis submitted in Fulfilment of the requirement for the award of the Degree of Philosophy of Doctoral in Electrical Engineering]; Faculty of Electrical and Electronic Engineering Universiti Tun Hussein Onn Malaysia; 2014

[12] Tatang M, Nonparametric A. System identification based on transient analysis with plant process of heat exchanger as study case. International Journal Of Innovation in Mechanical Engineering and Advanced Materials. 2015;**1**(1):19-26

[13] Tatang M, Mohd Nor MT, Dirman H. Parametric Model of Laboratory Heat Exchanger, Keynote Speaker: International Conference on Vocational Education and Electrical Engineering (ICVEE). 2015. pp. 25-37

[14] Tatang M, Judi A, Ekki K. Data analysis using system identification

toolbox of heat exchanger process control training system. In: Fourth International Conference on Information and Communication Technologies (ICoICT). 2016. pp. 466-471

[15] Tatang M. NNARX model structure for the purposes of controller design and optimization of heat exchanger process control training system operation. AIP Conference Proceedings. 2017;**1831**: 020040