

### Neuro-Fuzzy Controller Design for Level Controls

S. Intajag\*, V. Tipsuwanporn\*, N. Koetsam-ang\*\*, and K. Witheephanich\*\*\*

\*Faculty of Engineering, King Mongkut’s Institute of Technology Ladkrabang, Bangkok 10520, Thailand

(Tel: 66-2326-7347: Ext. 102; E-mail: [ktvittay@kmitl.ac.th](mailto:ktvittay@kmitl.ac.th))

\*\*Faculty of Engineering, Kasem Bundit University, Bangkok, Thailand.

\*\*\*Faculty of Engineering, Srinakharinwirot University, Nakornnayok 26120, Thailand.

**Abstract:** In this paper, a level controller is designed with the neuro-fuzzy model based on Takagi-Sugeno fuzzy system. The fuzzy system is employed as the controller, which can be tuned by the neural network mechanism based on a gradient descent technique. The tuning mechanism will provide an optimal process input by forcing the process error to zero. The proposed controller provides the online tunable mode to adjust the consequent membership function parameters. The controller is implemented with M-file and graphic user interface (GUI) of Matlab program. The program uses MPIBM3 interface card to connect with the industrial processes. In the experimentation, the proposed method is tested to vary of the process parameters, set points and load disturbance. Processes of one tank and two tanks are used to evaluate the efficiency of our controller. The results of the both processes are compared with two PID systems that are 3G25A-PIDO1-E and E5AK of OMRON. From the comparison results, our controller performance can be archived in the case of more robustness than the two PID systems.

**Keywords:** Level control, Neuro-Fuzzy, Fuzzy Controller, Controller design.

#### 1. INTRODUCTION

Usually, a fuzzy controller design is nothing more than a heuristic technique for the synthesis of controllers, which limits usefulness of the fuzzy control. In this paper, an artificial neural network mechanism is used to design the fuzzy controller, which both the neural network and fuzzy logic can bridge the gap together. The fuzzy logic has the capability to handle imprecise information through linguistic expressions [1] and the neural networks have the capability to learn for searching the optimal solutions. A combining methodology between neural networks with modules based on fuzzy logics yields a structure that can be called a “neuro-fuzzy” controller or, broadly speaking, a fuzzy neural network.

Although there has been a substantial of research in using the artificial neural networks to tune the fuzzy controllers [2, 3], applications of real-time process control have not been much investigated. This paper proposes the use of tuning method, which is provided by the artificial neural networks based on a gradient descent technique [4]. In this case, the consequent membership functions of fuzzy controllers is tuned by an on-line learning mechanism, the fuzzy system can effectively deal with the equivalent uncertainties that may appear in the subsystems due to plant uncertainty, function approximation error, or external disturbance.

The paper is organized as follows. First, architecture of the proposed neuro-fuzzy controller is presented, which the learning mechanism and its decision-making are discussed. The following section describes applications of the neuro-fuzzy controller for the level controls with two cascaded tanks. The experimental results are investigated on the plants of the level controls and are also provided to compare with two PID systems that are 3G25A-PIDO1-E and E5AK. Finally, the conclusion of the experiments is discussed.

#### 2. CONTROLLER DESIGN

Figure 1 shows our system, which is employed to control the level processes. The block of Personal Computer in Figure 1 is the neuro-fuzzy controller [5,6], which has two inputs (e: error, and Δe: change of error). The outputs of neuro-fuzzy controller provide manipulate values (MV) for using to control the processes. The manipulate values are converted from digital to analog (D/A) signals in the range 1-5 Volts and following by converting volt to current (V/I) by V/I converter,

which provides the current signal 4-20 mA. The process output signals are converted from analog to digital (A/D) signals by interface card MPIBM3 in the range 1-5 Volts for feedback the process values (PV).

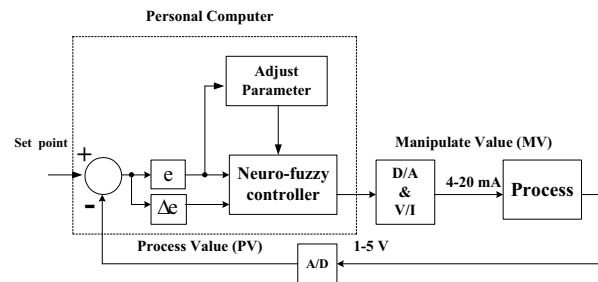


Fig. 1 Diagram of the proposed scheme.

An architecture of the proposed neuro-fuzzy network [7] is illustrated in Figure 2. The network architecture has five layers, which the first layer provides to fuzzify an input vector  $x$ . The second layer is used for inference the fuzzy rule-base and the 3<sup>rd</sup> – 5<sup>th</sup> layers are employed for defuzzification process to formulate the output  $y$ . From figure 1, the input vector  $x$  consists of the error and change of error at the time  $t$ , which is given as the following:

$$\begin{aligned} e(t) &= SP - PV(t) \\ \Delta e(t) &= e(t) - e(t-1) \end{aligned} \tag{1}$$

where SP denotes a set point value and  $PV(t)$  is a process value at time  $t$ .

Gradient descent method [4] is adapted for tuning the membership functions of the neuro-fuzzy network in Fig. 2. The fuzzy inference system under consideration has two inputs  $x_1$  and  $x_2$  and one output  $y$ . The numbers of rule-base depends on the numbers of linguistic label in each input; suppose that, the rule base contains two fuzzy if-then rules of Takagi and Sugeno’s type [6]:

Rule 1: If  $x_1$  is  $A_{11}$  and  $x_2$  is  $A_{21}$ , then  $y = \sum_{i=1}^R y_i$ ,

Rule 2: If  $x_1$  is  $A_{12}$  and  $x_2$  is  $A_{22}$ , then  $y = \sum_{i=1}^R y_i$ .

From the rules, the output node use the same function both

Rule 1 and Rule 2; however, the weight from inputs will be difference.

In figure 2, all nodes of the first layer are adaptive nodes, which the outputs of each node  $i$  ( $O_i^1$ ) for this layer are the degree of membership of the input vector ( $\mu_{A_{i,1}}(\mathbf{x})$ ), which obtains from the fuzzy membership function; consequently, each node can be represented by:

$$O_i^1 = \mu_{A_{i,1}}(\mathbf{x}), \quad i = 1, 2, \dots, n$$

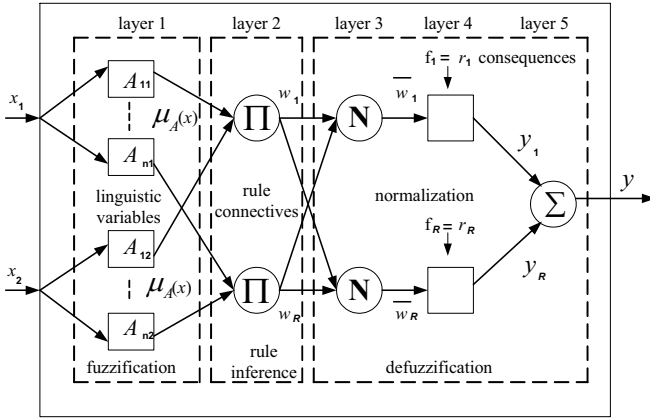


Fig. 2 The architecture of neuro-fuzzy.

where  $n$  denotes the number of term sets of the input vector  $\mathbf{x}$ .  $A_{i,1}$  and  $A_{i,2}$  are the fuzzy sets, in the proposed method, they provide by Gaussian fuzzy as given by:

$$\mu_{A_{k,j}}(x_k) = e^{-0.5 \left( \frac{x_k - c_{ik}}{\sigma_{ik}} \right)^2} \quad (2)$$

In this case,  $k=1,2$ , that represents index of  $\mathbf{e}$  and  $\Delta \mathbf{e}$  in Eq. (1). Where  $c_{ik}$  and  $\sigma_{ik}$  denote a position and spread of the Gaussian parameters, as seen in Fig. 4, which the learning mechanism will formulate these parameters to give the better performance of the controller.

Nodes in 2<sup>nd</sup> and 3<sup>rd</sup> layers of Fig. 2 are fixed that they are labeled with  $\Pi$  and  $N$ , respectively. The output nodes of the 2<sup>nd</sup> layer,  $O_j^2$ , are defined by:

$$O_j^2 = w_j = [\mu_{A_{i,1}}(x_1)]^T \mu_{A_{i,2}}(x_2),$$

where  $j=1, 2, 3, \dots, R$ , which  $R$  is the numbers of rule base. Suppose that, if the  $n = 3$  then  $R = n \times n = 9$ .

The output nodes of the 3<sup>rd</sup> layer,  $O_j^3$ , are performed by normalization, which are given by:

$$O_j^3 = \bar{w}_j = \frac{w_j}{\sum_{j=1}^R w_j}.$$

All the nodes in the 4<sup>th</sup> layer are adaptive nodes. The output nodes of this layer,  $O_j^4$ , are simply the products of the

normalized firing strength and a parameter  $r_j$  from the output membership function as seen in Fig. 5.

$$O_j^4 = \bar{w}_j r_j \quad (3)$$

The 5<sup>th</sup> layer has only one node labeled by  $\Sigma$  to indicate that it performs the sum function as given by:

$$O_j^5 = \mathbf{y} = \sum_{j=1}^R y_j.$$

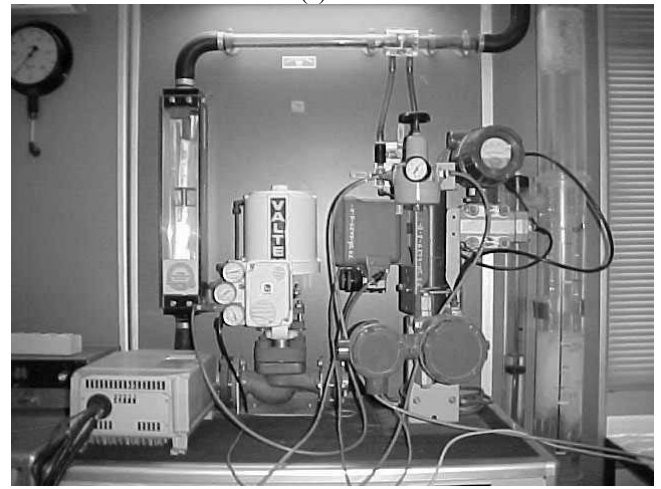
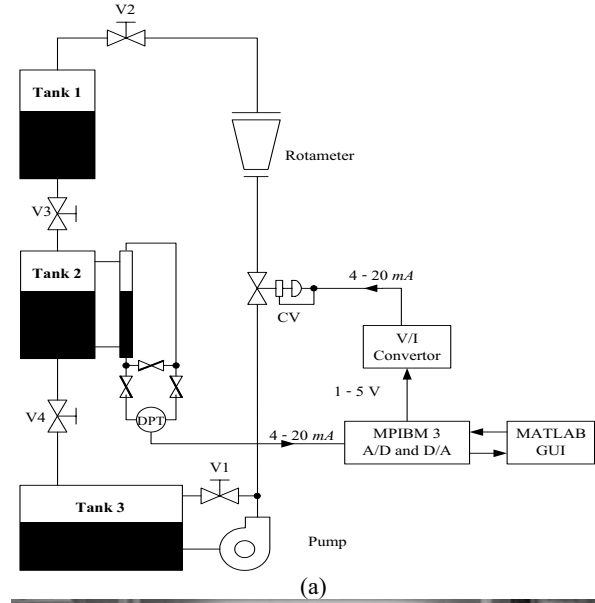


Fig. 3. (a) Diagram of level control system with two tank and (b) the experimental equipments.

In the neuro-fuzzy model, the parameters,  $c_{ik}$ ,  $\sigma_{ik}$  and  $r_j$  are defined by:

$$c_{ik}(t+1) = c_{ik}(t) - \eta_a (y_d - y(t)) \mu_{ik}(x_k) \left( \frac{x_k - c_{ik}}{\sigma_{ik}^2} \right) \frac{r_i - y_i}{\sum_{j=1}^R w_j},$$

$$\sigma_{ik}(t+1) = \sigma_{ik}(t) - \eta_b (y_d - y(t)) \mu_{ik}(x_k) \left( \frac{(x_k - c_{ik})^2}{\sigma_{ik}^3} \right) \frac{r_i - y_i}{\sum_{j=1}^R w_j},$$

$$\text{and } r_j(t+1) = r_j(t) - \eta_c (y_d - y(t)) \bar{w}_j.$$

Where  $y_d$  denotes the desired outputs, and  $\eta_a, \eta_b, \eta_c$  are the learning rates. From the designed network, the learning rates are defined as:  $\eta_a = 0.5, \eta_b = 0.5$  and  $\eta_c = 0.01$ . These parameters are derived from the gradient-based descent method [8].

### 3. LEARNING METHOD

The proposed method is tested to vary the process parameters, set points and load disturbance (DB) on the level controls as seen in Fig. 3.

From the level plant in Fig. 3(a), the input signal to the MPIBM3 card is 4-20 mA. This signal is used to control the plants by converting into the membership functions of the input variables as seen in Fig. 4, which are employed to formulate an output value through the fuzzy singleton in Fig. 5. This output value is encoded to the output signals, 4-20 mA, by the V/I converter.

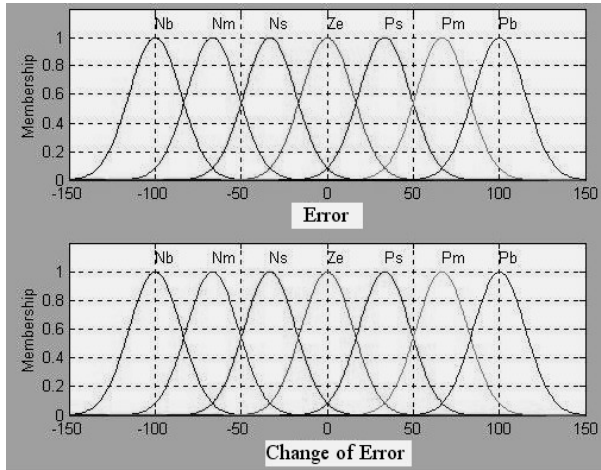


Fig. 4 Membership functions of the input variables.

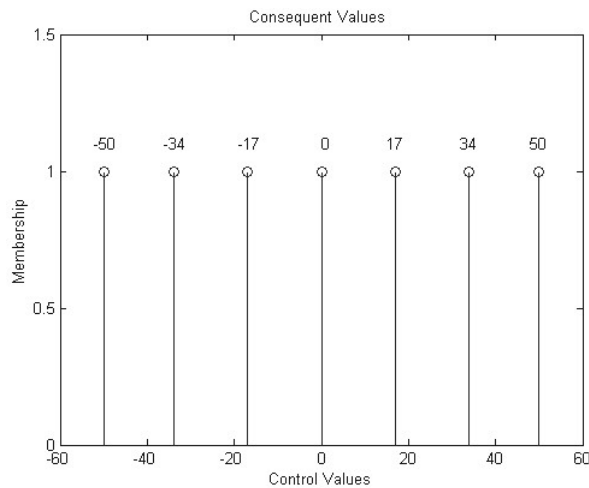


Fig. 5 Fuzzy membership functions of output variables for level control.

In Fig. 4 and 5, the fuzzy variables, which consist of Error, Change of Error and Control Values, are formulated in the form of error percentage for input variables and in the form of

valve-opening in percentage for output variables. In this case, the signals of MPIBM3 card is encoded and decoded in the form of percentage. The encoding of input signals as denoted in  $PV(t)$  of Eq. (1) is given by:

$$PV(t) = 100 \times \frac{s(t) - \text{Min}(t)}{\text{Max}(t) - \text{Min}(t)} \quad (4)$$

where  $s(t)$  denotes the 12-bit value of MPIBM3 at time  $t$ ,  $\text{Min}(t)$  and  $\text{Max}(t)$  are the minimum and maximum values of  $s(t)$ . From Fig. 5, the singleton values are provided to control the valve position, which corresponds to the fuzzy rule inference in Table 1. These values are decoded to control valve or manipulate value,  $MV(t)$ , by:

$$MV(t) = 100 \times \frac{y(t) - \text{Min}_{y(t)}}{\text{Max}_{y(t)} - \text{Min}_{y(t)}} \quad (5)$$

where  $y(t)$  denotes the output values of fuzzy controller as seen in Fig. 2,  $\text{Min}_{y(t)}$  and  $\text{Max}_{y(t)}$  are minimum and maximum values of  $y(t)$ , respectively.  $MV(t)$  in Eq. (5) is converted to 1-5 Volts by MPIBM3 card, and transmitting into V/I converter to transform 1-5 Volts to 4-20 mA for controlling the valve position.

The membership functions in Fig. 4 and 5 are employed as initialization values for controlling the plants. If the controller provides the error values more than desired value the network turn to the learning mode until the error less than the desired value the neuro-fuzzy controller will use only feed-forward part.

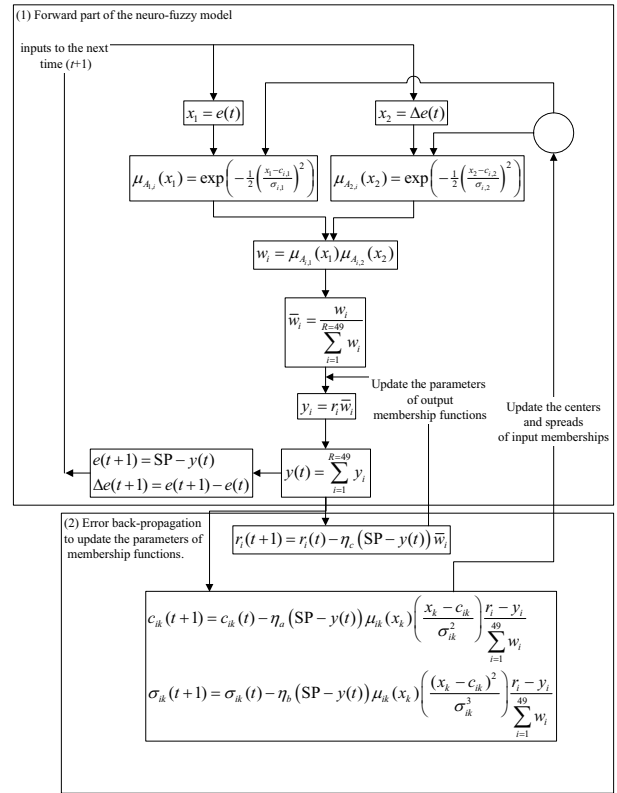


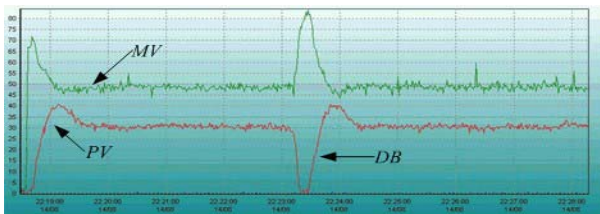
Fig. 6 The learning mechanism diagram of the neuro-fuzzy model.

Table 1 Fuzzy rule inference of the level control

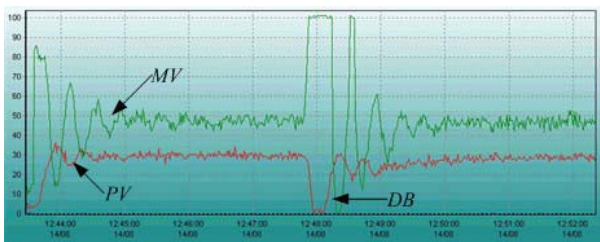
$\Delta e$	Nb	Nm	Ns	Ze	Ps	Pm	Pb
Nb	-50	-50	-50	-50	-34	-17	0
Nm	-50	-50	-50	-34	-17	0	17
Ns	-50	-50	-34	-17	0	17	34
Ze	-50	-34	-17	0	17	34	50
Ps	-34	-17	0	17	34	50	50
Pm	-17	0	17	34	50	50	50
Pb	0	17	34	50	50	50	50

Table 2 Fuzzy rule-base after tuning with the set point 70%.

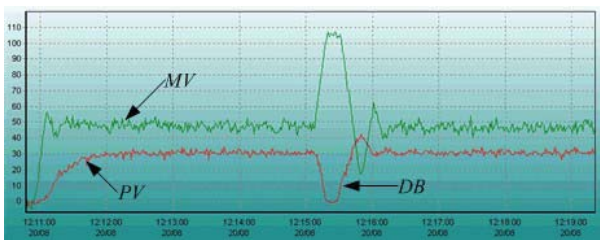
$\Delta e$	Nb	Nm	Ns	Ze	Ps	Pm	Pb
Nb	-50	-50	-50	-50	-34	-17	0
Nm	-50	-50	-50	-34	-17	0	17
Ns	-50	-50	-34	<b>-18.28</b>	<b>-1.18</b>	<b>16.99</b>	34
Ze	-50	-34	-17	<b>5.66</b>	<b>22.17</b>	34	50
Ps	-34	<b>-16.99</b>	<b>0.02</b>	<b>47.17</b>	<b>61.48</b>	<b>50.02</b>	50
Pm	-17	0	<b>17.15</b>	<b>49.33</b>	<b>61.60</b>	<b>50.49</b>	<b>50.07</b>
Pb	0	17	<b>34.04</b>	<b>52.70</b>	<b>51.83</b>	<b>50.07</b>	<b>50.01</b>



(a) The results of neuro-fuzzy controller.



(b) The results of 3G2A5-PID01-E controller.



(c) The results of E5AK-PID01-E controller.

Fig. 7 The process response results at control level 30%.

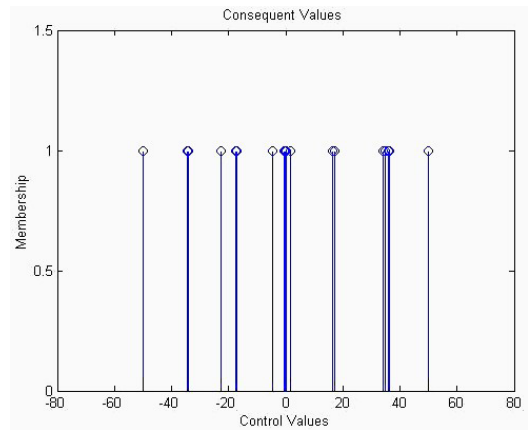
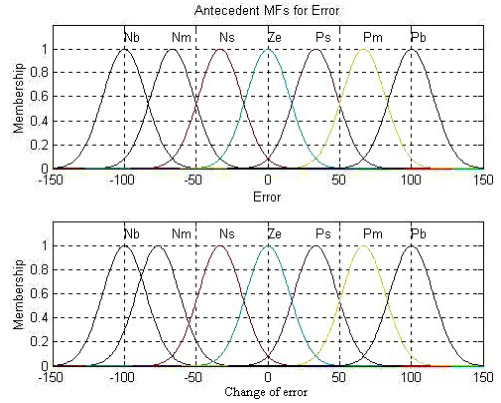
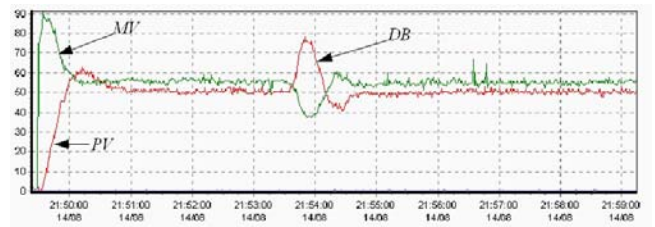
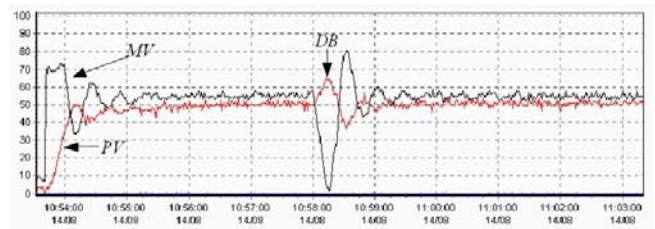


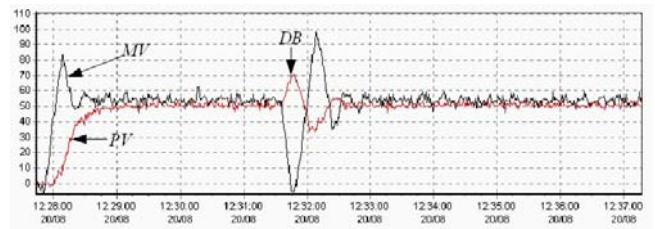
Fig. 8 The membership functions of level control at 30% of the set point.



(a) The results of neuro-fuzzy controller.



(b) The results of 3G2A5-PID01-E controller.



(c) The results of E5AK-PID01-E controller.

Fig. 9 The response results at control level 50%.



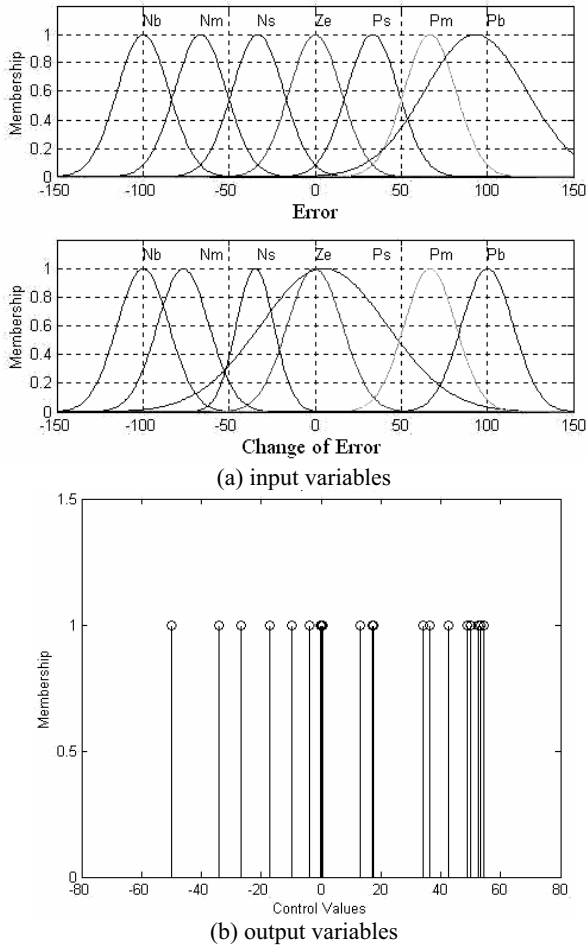


Fig. 10 The membership functions of level control at 50% of the set point.

### 3.1 Determining neuro-fuzzy parameters

Gaussian membership functions in Eq. (2) are provided to define seven term sets of the input vector that the predefined term sets before training the network of the level controls are illustrated in Fig. 4. Table 1 shows the rule-base inference of the level controls, which its values correspond to the output layer of the neuro-fuzzy network being the form of fuzzy singletons.

### 3.2 Tuning mechanism

As described in Section 2, the Gaussian input membership functions with centers  $c_{ik}$  and spreads  $\sigma_{ik}$  produce for the premise and implication and output membership function center  $r_i$  provide for the center-average defuzzification, which these functions are taken in the form:

$$f(x|\theta) = \frac{\sum_{i=1}^R r_i \prod_{k=1}^n \exp\left(-\frac{1}{2}\left(\frac{x_k - c_{ik}}{\sigma_{ik}}\right)^2\right)}{\sum_{i=1}^R \prod_{k=1}^n \exp\left(-\frac{1}{2}\left(\frac{x_k - c_{ik}}{\sigma_{ik}}\right)^2\right)} \quad (6)$$

where  $f(x|\theta)$  is a function used to estimate parameter  $\theta$  from data  $x$ .

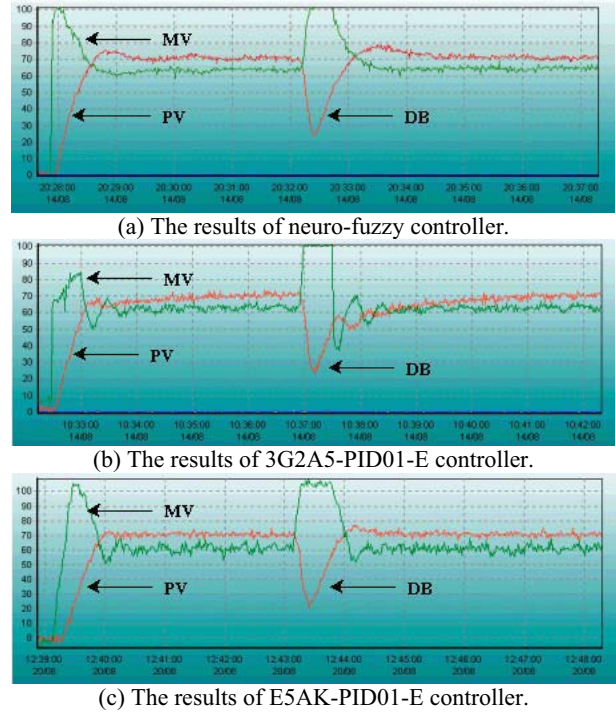


Fig. 11 The process response at set-point value 70%.

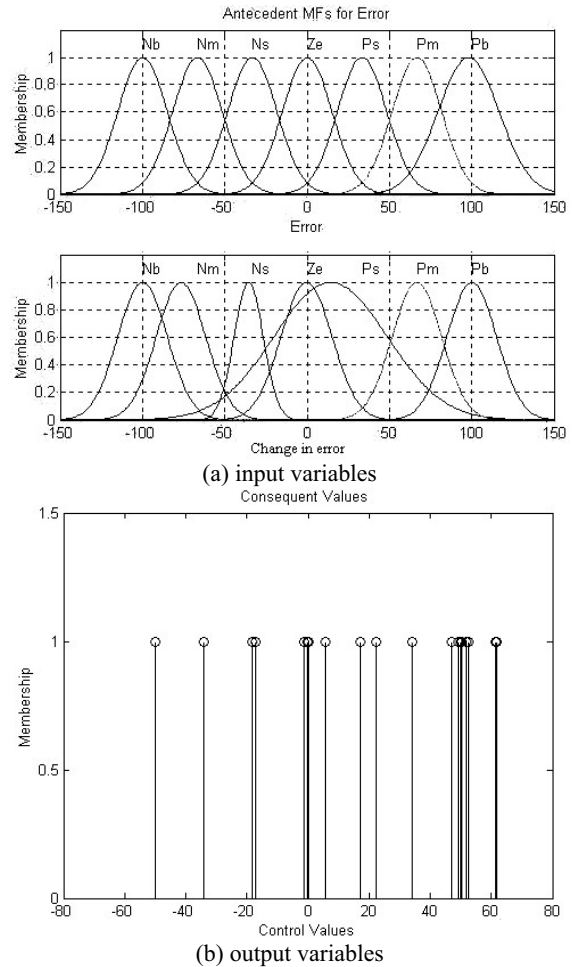


Fig. 12 The membership functions of level control at 70% of the set point.

In gradient descent method [4], we seek to minimize the error term,  $y_d - y(t)$ , by establishing the parameter  $\theta$ , which, in this case, the parameter consists of  $c_{ik}$ ,  $\sigma_{ik}$  and  $r_i$  as defined in Section 2.

In Fig. 6, the learning part and the forward part are illustrated. Usually, the neuro-fuzzy controller will be employed only the forward part to control the plants; however, if the system is disturbed and the error of system more than 10%; the learning part will be turned on until system error less than 10%; the learning part will be stopped. In the learning part, the parameters of both input and output membership functions are updated to provide for calculating the membership values in the next coming epoch ( $t+1$ ).

In our fuzzy controller, the numbers of rule-base ( $R$ ) have 49 rules ( $R=49$ ) as seen in Table 1. When the rule-base in Table 1 is tuned at the set point value equal to 70% ( $SP=70$ ) of the level in second tank, this rule-base is adjusted as shown in Table 2. With the boldface values in Table are the effects from the tuning mechanism. For its controlling results will be illustrated in the next Section.

#### 4. EXPERIMENTAL RESULTS

In the level control, two PID controllers: 3G2A5-PID01-E, which is auto-tuning controller based on Ziegler-Nichols, and E5AK based on fuzzy self-tuning, are used for comparing with our controller. Figure 7 shows the results of the controllers at 30% of the set point valve. The membership functions after trained the network and used for controlling the valve are presented in Figure 8. Figure 9 illustrates the controlling results at 50% of the set point value. Figure 10 is shown the membership functions of the neuro-fuzzy controller that is trained at the set-point value 50%. Figure 11 shows the results of the controllers at 70% of the set point valve that the membership functions after trained the network and used for controlling the valve are presented in Figure 12.

#### 5. CONCLUSIONS

From the experimental result, our controller performance can be archived in the case of more robustness of our controller than the one of two PID controllers, that is, for the transient response after taking the load disturbance,  $DB$ , the rise time and the settling time are best provided by our controller. The advantage of the proposed controller is the self-tuning mechanism providing the optimal parameters to control the processes on the real time; consequently, the neuro-fuzzy controller endows the tolerance with the disturbance.

#### REFERENCES

- [1] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, Vol. 8, pp. 338-353, 1965.
- [2] T. Tani, S. Murakoshi, T. Sato, M. Umano, and K. Tanaka, "Application of neuro-fuzzy hybrid control system to tank level control," *Second IEEE Intl. Conf. on Fuzzy Systems*, Vol.1, pp. 618-623, 1993.
- [3] T. Tani, S. Murakoshi, and M. Umano, "Neuro-fuzzy hybrid control system of tank level in petroleum plant," *IEEE Trans. on Fuzzy Systems*, Vol. 4, No. 3, pp. 360-368, 1996.
- [4] D. E. Rumelhart, L. McClelland, *Exploration in Parallel Distributed Processing; A Handbook of Model Programs; and Exercises*, MIT Press, Massachusetts, USA., 1988.
- [5] C. T. Lin and C.S George Lee, *Neural Fuzzy System*, Prentice-Hall, 1996.
- [6] J. S. R. Jang, C.T Sun and E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Prentice Hall, 1997.
- [7] L. X. Wang, *A course in fuzzy systems and control*, Prentice-Hall, 1997.
- [8] J. W. Hines, *Matlab Supplement to Fuzzy and Neural Approaches in Engineering*, USA: John Wiley & Sons, Ins., 1997.