ANFIS Approach for Identification of Debutanizer Column

Hamed Sahraie[1] (Corresponding Author), Ali Ghaffari[2], Majid Amidpour[2]

[1.] National Iranian Southfield Oil Co
*Email: Hamed.Sahraie@yahoo.com
[2.] K.N. Toosi University of Technology, Tehran, Iran
Ghaffari@kntu.ac.ir , Amidpour@kntu.ac.ir

Abstract: In this paper the Adaptive Neuro-Fuzzy Inference System (ANFIS) is used to identify and model a real debutanizer column in one of the Iranian refineries. The outputs of dynamic model in addition to recent inputs depend on previous outputs and inputs. Selected inputs and outputs are those that will be used as manipulated and controlled variables. The type and number of membership functions obtain from error and trial approach and optimal configuration is chosen by root mean square error (RSME) criterion. According to RMSE between real and simulated outputs, the obtained model is acceptable with the aim of control.
[Hamed Sahraie, Ali Ghaffari, Majid Amidpour. **ANFIS Approach for Identification of Debutanizer Column.** *N Y Sci J* 2013;6(12):1-9]. (ISSN: 1554-0200). http://www.sciencepub.net/newyork. 1

Keywords: Debutanizer column; Adaptive Neuro-Fuzzy Inference System; ANFIS; system identification; modeling

1. Introduction

Distillation of multicomponent mixtures is one of the most common separation operations in the chemical industry and refineries. Distillation is a multivariable constrained, coupled, nonlinear, nonstationary process with complex dynamic structure. Distillation columns consume high level of energy that an accurate and tight control of those can decrease a large amount of energy consumption in refineries.[1]

In addition to more classical identification methods such as NARMAX modeling [2], [3] a new set of methods has been developed recently which apply artificial neural networks to the tasks of identification and control of dynamic systems. These works are supported by two of the most important capabilities of neural nets: their ability to learn [4], [5], (based on the optimization of an appropriate error function) and their good performance for the approximation of nonlinear functions [6], [7].

At present, most of the works on system identification using neural nets are based on multilayer feedforward neural networks with backpropagation learning or more efficient variations of this algorithm. These methods have been applied to real processes and they have shown an adequate behavior .It is important to remark that most of them use static discrete-time models that capture the dynamics of the real process through the use of tapped-delay lines in the model inputs and outputs[8],[9]. A number of drawbacks associated with this type of models may appear in the identification of complex dynamic systems, such as difficulties in selecting the appropriate number of required delays and, in some cases, poor identification performance when implemented on-line after training off-line, due to training deficiencies.

Dynamic networks are generally more powerful than static networks (although somewhat more difficult to train). Because dynamic networks have memory, they can be trained to learn sequential or time-varying patterns. This has applications in such disparate areas as prediction in financial markets [10], channel equalization in communication systems [11], phase detection in power systems [12], sorting [13], fault detection [14], speech recognition [15], and even the prediction of protein structure in genetics [16]. You can find a discussion of many more dynamic network applications in [17].

Engell et al. [18] used a semi-batch reactive distillation process. A comparison was carried out between conventional control structures and model-based predictive control by using a neural net plant model. Brizuela et al. [19] used a nonlinear model of the process for prediction of future outputs that using a feed forward neural network (FNN). Wen et al. [20] obtain some new results on system identification with dynamic neural networks. They concluded that the gradient descent algorithm for weight adjustment is stable in an $L\infty$ sense and robust to any bounded uncertainties. Li Shurong et al. [21] used dynamic neural network to learn the input-output behaviors of a binary distillation column by combining the mechanistic property. The convergence of the algorithm was discussed by using the Lyapunov method. Based on the identified model, a nonlinear adaptive controller was designed, which can preserve the stability and robustness of the closed loop system. Calderon et al. [22] worked with the Dynamical Recurrent Neural Network as a tool for system identification and trained the network using a timedependent backpropagation learning algorithm and showed that for modeling a nonlinear dynamical system, their neural device had good performance for interpolation and extrapolation, and was very robust in the presence of noise.

2. Distillation column dynamic modeling

In this section the required equations to obtain mathematical modeling is described. The column for which the model is presented separates a single multicomponent liquid feed into two liquid products in a tray-type distillation column. The column is equipped with a reboiler and a total condenser.

In most models to simplify the equations usually some assumptions are considered. In mathematical modeling usually the model assumes that vapor holdups are negligible and that the effluent streams are in thermodynamic equilibrium. The column pressure is assumed to remain constant throughout the dynamic tests. The dynamic of the reboiler and the condenser are neglected. Finally, the dynamic changes in internal energy on the trays are assumed to be so rapid that the energy equation reduces to an algebraic equation.

With the foregoing assumptions, the mathematical dynamic model can be expressed by the following set of differential and algebraic equations.

Overall mass balance for each tray:

$$\frac{dM_n}{dt} = L_{n+1} + V_{n-1} - L_n - V_n \tag{1}$$

Component balance for each tray:

$$\frac{dM_n x_{n,j}}{dt} = L_{n+1}x_{n+1,j} + V_{n-1}y_{n-1,j} - L_n x_{n,j} - V_n y_{n,j} \tag{2}$$

Energy balance for tray n:

$$L_{n+1}h_{n+1} + V_{n-1}H_{n-1} - L_n h_n - V_n H_n = 0 \tag{3}$$

Tray hydraulics:

If the Francis Weir formula is used, the relationship is:

$$L_n = 3.33\, lh_n^{3/2} \tag{4}$$

Where *l* is the length of weir in feet, h_n is the height of liquid over weir in feet, and *Ln* is the liquid leaving stage n, ft³/sec.

Phase equilibrium:

$$y_{n,j} = K_{n,j}x_{n,j} \tag{5}$$

Murphree vapor-phase efficiency:

$$y^A_{n,j} = y^A_{n-1,j} + \eta ij(y_{n,j} - y^A_{n-1,j}) \tag{6}$$

where the superscript A denotes actual concentration.[23]

3. Adaptive Neuro-Fuzzy Inference System (ANFIS)

Usual approaches to system modeling rely heavily on mathematical tools which emphasizes a precise description of the physical quantities involved. By contrast, modeling approaches based on neural networks and fuzzy logic are becoming a viable alternative where the earlier conventional techniques fail to achieve satisfactory results. Neuro-fuzzy modeling is concerned with the extraction of models from numerical data representing the behavioral dynamics of a system.

This modeling approach has a two-fold purpose:

- It provides a model that can be used to predict the behavior of the underlying system in range of operation.
- This model may be used for controller design.

The basic idea behind the adaptive neuro-fuzzy learning techniques is very simple. These techniques provide a method for the fuzzy modeling procedure to learn information about data set, in order to compute the membership function parameters that best allow the associated fuzzy inference system to track the given input-output data. ANFIS constructs an input-output mapping based on both human knowledge (in the form of fuzzy if-then rules) and simulated input/output data pairs. It serves as a basis for building the set of fuzzy if-then rules with appropriate membership functions to generate the input output pairs.

The parameters associated with the membership functions are open to change through the learning process. The computation of these parameters (or their adjustment) is facilitated by a gradient vector, which provides a measure of how well the ANFIS is modeling the input output data for a given parameter set. Once the gradient vector is obtained, backpropagation or hybrid learning algorithm can be applied in order to adjust the parameters.

3.1. ANFIS architecture

ANFIS architecture consists of five layers with the output of the nodes in each respective layer is represented by $O_{i,l}$ where i is the ith node of layer l [24]. A simple architecture of ANFIS is shown in Fig 1.

Layer 1: Generate the membership grades

$$O_{1,i} = \mu_{A_i}(x), \qquad \text{For i=1, 2,}$$

$$O_{1,i} = \mu_{B_{i-2}}(x) \qquad \text{For i=3, 4,} \tag{7}$$

where x (or y) is the input to the node and Ai (or Bi_2) is the fuzzy set associated with this node such as the generalized bell function

$$\mu_A(x) = \frac{1}{1 + \left|\frac{x - c_i}{a_i}\right|^{2b}} \tag{8}$$

Where {ai, bi, ci} is the parameter set referred to as premise parameters.

Layer 2: Generate the firing strengths by multiplying the incoming signals and outputs the t-norm operator result,e.g.

$$O_{2,i} = w_i = \mu_{A_i}(x)\mu_{B_i}(y),...,i = 1,2 \tag{9}$$

Layer 3: Normalize the firing strengths

$$O_{3,i} = \overline{w}_i = \frac{w_i}{w_1 + w_2},...,i = 1,2. \tag{10}$$

Layer 4: Calculate rule outputs based on the consequent parameters {pi, qi, ri}

$$O_{4,i} = \overline{w}_i f_i = \overline{w}_i(p_i x + q_i y + r_i), \tag{11}$$

Layer 5: Computes the overall output as the summation of incoming signals

$$\text{overall output} = O_{5,i} = \sum_i \overline{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{12}$$

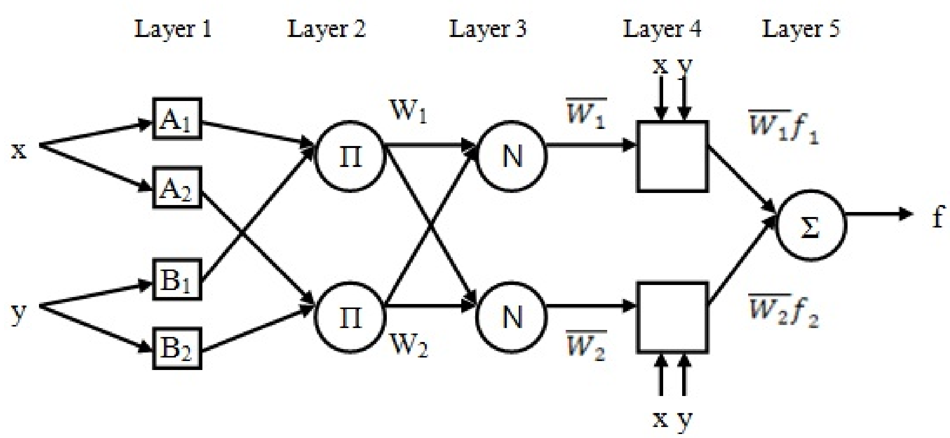

Fig. 1 ANFIS architecture for two rules

3.2. Hybrid learning algorithm

- Forward pass

In the forward pass of the hybrid learning algorithm, node outputs go forward until layer 4 and the consequent are identified by the least-squares method. When the values of the premise parameters are fixed, the overall output can be expressed as a linear combination of the consequent parameters

$$f = \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2$$
$$= \overline{w}_1 f_1 + \overline{w}_2 f_2 = (\overline{w}_1 x) p_1 + (\overline{w}_1 y) q_1 + (\overline{w}_1) r_1 + (\overline{w}_2 x) p_2 + (\overline{w}_2 y) q_2 + (\overline{w}_2) r_2 \quad (13)$$

which is linear in the consequent parameters p1, q1, r1, p2, q2 and r2

$$f = [\overline{w}_1 x \quad \overline{w}_1 y \quad \overline{w}_1 \quad \overline{w}_2 x \quad \overline{w}_2 y \quad \overline{w}_2] \begin{bmatrix} p_1 \\ q_1 \\ r_1 \\ p_2 \\ q_2 \\ r_2 \end{bmatrix} = XW \quad (14)$$

If X matrix is invertible then

$$W = X^{-1} f \quad (15)$$

Otherwise a pseudo-inverse is used to solve for W.

$$W = (X^T X)^{-1} X^T f \quad (16)$$

- Backward pass

In the backward pass, the error signals propagate backward and the premise parameters are updated by gradient descent.

$$a_{ij}(t+1) = a_{ij}(t) - \eta \frac{\partial E}{\partial a_{ij}} \quad (17)$$

where η is the learning rate for aij. The chain rule is used to calculate the partial derivatives used to update the membership function parameters.

$$\frac{\partial E}{\partial a_{ij}} = \frac{\partial E}{\partial f} \frac{\partial f}{\partial f_i} \frac{\partial f_i}{\partial w_i} \frac{\partial w_i}{\partial \mu_{ij}} \frac{\partial \mu_{ij}}{\partial a_{ij}} \quad (18)$$

The partial derivatives are derived as follows:

$$E = \frac{1}{2}(f - f^t)^2 \quad \text{hence} \quad \frac{\partial E}{\partial f} = (f - f^t) = e \quad (19)$$

$$f = \sum_{i=1}^{n} f_i \quad \text{hence} \quad \frac{\partial f}{\partial f_i} = 1 \quad (20)$$

$$f_i = \frac{w_i}{\sum_{i=1}^{n} w_i} (p_i x + q_i y + r_i) \quad (21)$$

hence

$$\frac{\partial f_i}{\partial w_i} = \frac{(p_i x + q_i y + r_i) - f}{\sum_{i=1}^{n} w_i} \quad (22)$$

$$w_i = \prod_{j=1}^{m} \mu_{A_{ji}} \quad \text{hence} \quad \frac{\partial w_i}{\partial \mu_{ij}} = \frac{w_i}{\mu_{ij}} \quad (23)$$

The last partial derivative depends on the type of membership functions used. The parameters of the other membership functions are updated in the same fashion.
The gradient is then obtained as

$$\frac{\partial E}{\partial a_{ij}} = e\,\frac{(p_i x + q_i y + r_i) - f}{\sum_{i=1}^{n} w_i}\,\frac{w_i}{\mu_{A_{ij}}}\,\frac{\partial \mu_{A_{ij}}}{\partial a_{ij}} \tag{24}$$

$$\frac{\partial E}{\partial b_{ij}} = e\,\frac{(p_i x + q_i y + r_i) - f}{\sum_{i=1}^{n} w_i}\,\frac{w_i}{\mu_{B_{ij}}}\,\frac{\partial \mu_{B_{ij}}}{\partial b_{ij}} \tag{25}$$

4. Description of the plant

The column is located in one of the Iranian refineries and it is part of naphtha splitter plant.
In the debutanizer column C3 (propane) and C4 (butane) are removed as heavier composition as C5 (pentane).

The debutanizer column is required to:

* make certain about adequate fractionation in the debutanizer;
* minimize the C5 (stabilized gasoline) content in the debutanizer top product (L.P.G splitter feed), while respecting the limit enforced by law;
* minimize the C4 (butane) content in the debutanizer bottoms (Naphtha splitter feed).

A detailed scheme of the debutanizer column is shown in Fig. 2. A number of sensors are installed on the plant to monitor product quality. The subsets of sensors relevant to the described application are listed in Table 1, together with the corresponding description.

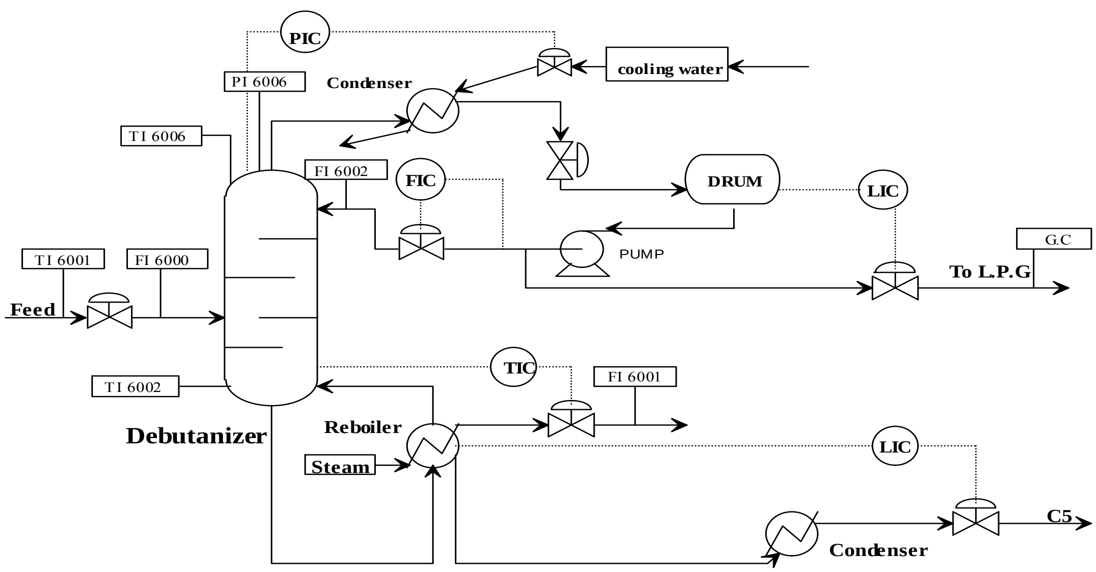

Fig. 2 Schematic diagram of debutanizer column

Table 1. Sensors relevant to the describe application and corresponding characteristics.

Tag	**Description**	**units**
TI 6001	Feed temperature	°C
FI 6000	Feed flow	Kbbl / day
TI 6002	Bottom temperature	°C
TI 6006	Top temperature	°C
PI 6006	Top pressure	bar
FI 6002	Reflux flow	m^3 / hr
FI 6001	Steam flow	m^3 / hr
G.C	Gas Chromatograph	mole fraction

6. Results and Discussions

As shown in Fig.2 there are five control loops in debutanizer column, so, there are five controlled variables and five manipulated variables. And another variable that is changing during the column is operation is the feed flow rate. The model is structured on the base of inputs, that are manipulated variables in control loops (reflux flow, steam flow entering to reboiler, cooling water flow entering to condenser, top and bottom product flows) plus feed flow variable ,and outputs which are controlled variables (top and bottom temperatures, pressure of column, liquid level of drum and reboiler).

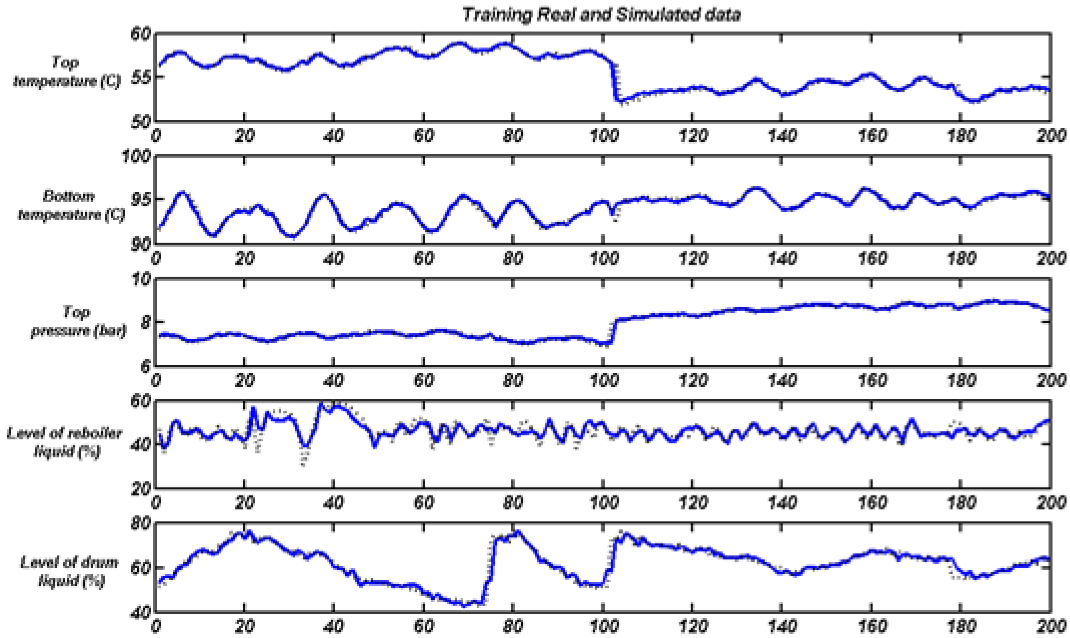

Fig.3 Real and Simulated outputs of training Data

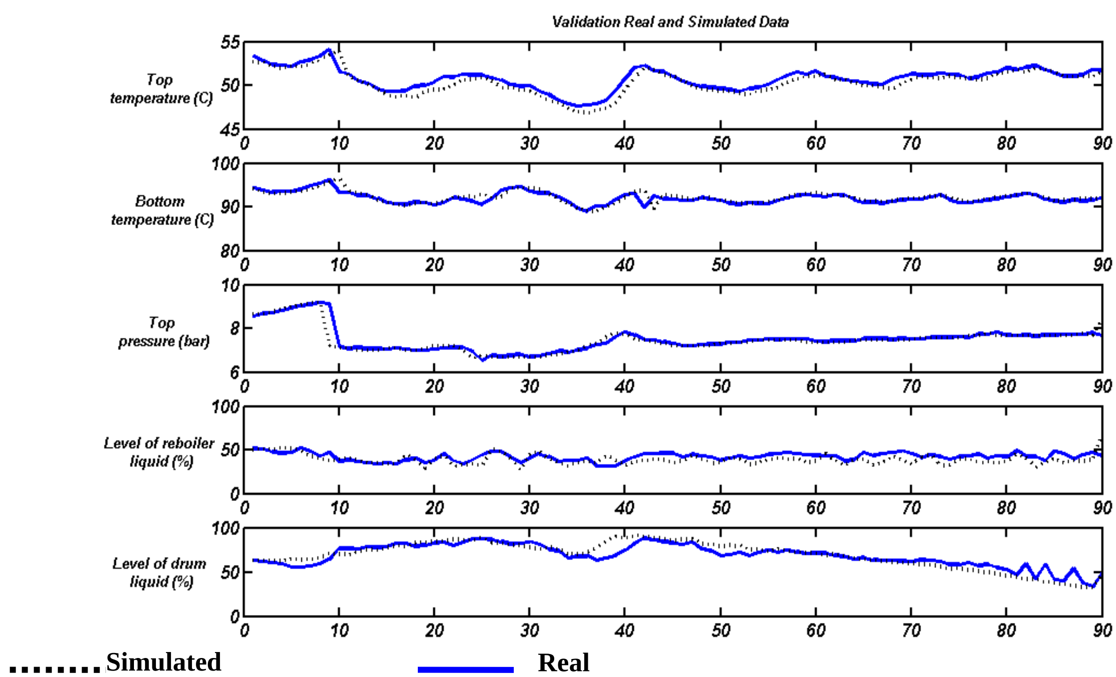

Fig.4 Real and Simulated outputs of validation Data

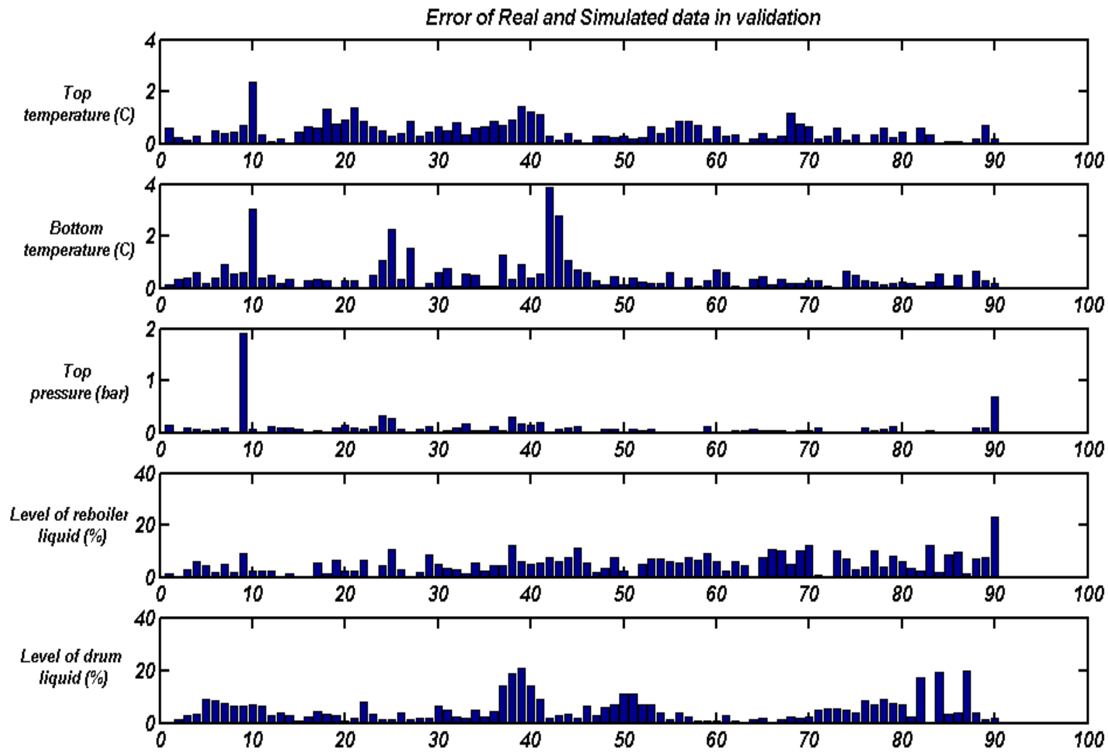

Fig.5 Absolute error between real and simulated outputs of validation data

In this paper the Adaptive Neuro-Fuzzy Inference System (ANFIS) is considered as a new approach for modeling. In comparison of ANN and ANFIS, the performance of ANFIS is better than ANN and also, ANFIS has other advantages than ANN. ANN is more time consuming process than ANFIS. In ANN, number of layers, number of each layer's node and type of transfer functions should be obtained from trial and error method that they are time consuming process but, in ANFIS just number of membership function and type of them must be selected that suitable result is usually obtained after four or five trial. However an ANN model is constructed and the RMSE of validation data have been shown in Table 4.

Table 4. RMSE of validating data for ANFIS and ANN model

	Top temperature (°C)	Bottom temperature (°C)	Top pressure (bar)	Level of reboiler liquid (%)	Level of drum liquid (%)
RMSE (ANFIS model)	0.618	0.785	0.232	6.407	6.710
RMSE (ANN model)	0.956	0.854	0.239	7.852	6.965

7. Conclusion

As known the distillation column is severe nonlinear system and to achieve the mathematical model many nonlinear and linear differential and algebraic equations should be solved that this is difficult and time consumer work. So, other methods as artificial neural networks and fuzzy logic or combination of both of them are employed. Adaptive neuro-fuzzy inference system (ANFIS) can use previous inputs and outputs to obtain new outputs. ANN model, the ANFIS model is faster and more accurate.

Nomenclature

ηij	Murphree stage efficiency
h_F	total molar enthalpy of feed (kJ/kmol)
h_i	total molar enthalpy of liquid mixture (kJ/kmol)
H_i	total molar enthalpy of vapor (kJ/kmol)
h_{lij}	component liquid enthalpy (kJ/kmol)
H_{vij}	component vapor enthalpy (kJ/kmol)
$K_{n,j}$	equilibrium constant
L_i	total liquid flow rate leaving the tray (kmols/h)
L_{ij}	component liquid flow rate leaving the *i*th tray (kmols/h)
M_i	liquid molar holdup on *i*th tray (kmols)
NC	number of components
v_I	scalar product of *i*th weight vector and input vector
NT	total number of trays in distillation column
R	reflux rate (kmols/h)
U	side stream in liquid phase
Vi	total vapor flow rate from the tray (kmols/h)
v_{ij}	component vapor flow rate from the tray (kmols/h)
W	side stream in vapor phase
x	liquid composition of more volatile component (mole fraction)
x_{ij}	liquid composition if *j*th component on *i*th tray (mole fraction)
y	vapor composition of more volatile component (mole fraction)
y^*	equilibrium vapor composition of more volatile component (mole fraction)
y_{ij}	vapor composition of *j*th component on *i*th tray (mole fraction)
y_{ij}^*	equilibrium vapor composition of *j*th component on *i*th tray (mole fraction)
yn	input vector to neuron layer
ANFIS	Adaptive Neuro-Fuzzy Inference System
RMSE	Root Mean Square Error
ANN	Artificial Neural Network

Acknowledgment

Authors thank National Iranian Oil Company (NIOC) and National Iranian Southfield Oil Company (NISOC) for their help and financial support.

References

1 Y. Chetouani. (1992).Using Artificial Neural networks for the modeling Of a distillation column. International Journal of Computer Science & Applications ,Vol. 4 Issue 3, pp 119-133.

2 M.J. Korenberg and L.D. Paarmann.(1991). Orthogonal approaches to time-series analysis and system identification. IEEE Signal Proc. Mag., no.7, pp. 29-43.

3 V.J. Mathews.(1991). Adaptive polynomial filters,'' IEEE Signal Proc. Mag., no. 7 , pp. 10-26.

4 K.S. Narendra and K. Parthasarathy. (1990). Identification and control of dynamical systems using neural networks. IEEE Trans. Neural Networks, vol. 1, no. 1, pp. 4-26.

5 K.S. Narendra and K. Parthasarathy. (1990). Gradient methods for the optimization of dynamical systems containing neural networks. IEEE Trans, Neural Networks, vol. 2, no. 2, pp. 252-262.

6 M.M. Polycarpou and P.A. Ioannou,(1991). Identification and control of nonlinear systems using neural network models: Design and stability analysis,'' Tech. Rep. 91-09-01, Univ, of Southern California

7 F. Girosi and T. Poggio. (1989). Representation properties of networks: Kolmogorov's theorem is irrelevant,'' Neural Computation, vol, 1 pp. 465-469.

8 W.T. Miller, R.S. Sutton, and P.J. Werbos. (1990). Neural Networks for Control. MIT Press, Cambridge, MA.

9 K.S. Narendra and K. Parthasarathy.(1990). Identification and control of dynamical systems using neural networks. IEEE Trans. Neural Networks, vol. 1, no. 1, pp. 4-26.

10 J.Roman, and A.Jameel.(1996). Backpropagation and recurrent neural networks in financial analysis of multiple stock market returns. Proceedings of the Twenty-Ninth Hawaii International Conference on System Sciences, Vol. 2, pp. 454-460.

11 J.Feng, C.K. Tse, and F.C.M. Lau. (2003). A neural-network-based channel-equalization strategy for chaos-based communication systems. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, Vol. 50, No. 7, pp. 954-957.
12 I.Kamwa, R. Grondin, V.K. Sood, C. Gagnon, Van Thich Nguyen, and J. Mereb. (1996). Recurrent neural networks for phasor detection and adaptive identification in power system control and protection. IEEE Transactions on Instrumentation and Measurement, Vol. 45, No. 2, pp. 657-664.
13 Jayadeva and S.A.Rahman.(2004). A neural network with O(N) neurons for ranking N numbers in O(1/N) time. IEEE Transactions on Circuits and Systems I: Regular Papers, Vol. 51, No. 10, pp. 2044-2051.
14 G.Chengyu, and K. Danai. (1999). Fault diagnosis of the IFAC Benchmark Problem with a model-based recurrent neural network. Proceedings of the 1999 IEEE International Conference on Control Applications, Vol. 2, pp. 1755-1760.
15 A.J.Robinson. (1994). An application of recurrent nets to phone probability estimation," IEEE Transactions on Neural Networks, Vol. 5.
16 P.Gianluca, D.Przybylski, B.Rost, P.Baldi, (2002). Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. Proteins: Structure, Function, and Genetics, Vol. 47, No. 2, pp. 228-235.
17 L.R.Medsker, and L.C. Jain.(2000). Recurrent neural networks: design and applications, Boca Raton, FL: CRC Press.
18 S.Engell, and G.Fernholz,(2003). Control of a reactive separation process. Chemical Engineering and Processing, 42:201-210.
19 E.Brizuela,M.Uria,R.Lamanna. (1996). Predictive control of a multi-component distillation column basedon neural networks. IEEE Neural Networks for Identification, Control, Robotics, and Signal/Image Processing, 1996. Proceedings.,pp. 270-278.
20 Y.Wen, L.Xiaoou, (2001). Some new results on system identification with dynamic neuralnetworks. IEEE Transactions on Neural Networks ,Vol 12, Issue 2, pp:412 – 417.
21 L. Shurong, L. Feng, (2000). Dynamic neural network based nonlinear adaptive control for adistillation column , Proceedings of the 3rd World Congress on Intelligent Control and Automation,.Vol 5.
22 G.Calderon, J.P.Draye, D Pavisic, R.Teran and G.Libert.(1996).Nonlinear Dynamic System Identification with Dynamic Recurrent Neural Networks. International workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing (NICROSP '96)
23 P.B. Deshpande.(1985). Distillation Dynamics and Control, Instrument Society of America, McGraw Hill Publishing Co. Ltd.
24 M.A. Denai, F.Palis, and A.Zeghbib. (2007). Modeling and control of non-linear systems using soft computing technique. Applied Soft Computing ,pp 728-738.

11/12/2013