Separation and Purification Technology 82 (2011) 1–9

Contents lists available at SciVerse ScienceDirect

Separation and Purification Technology

journal homepage: www.elsevier.com/locate/seppur

Design of neural network for manipulating gas refinery sweetening regenerator column outputs

Mahdi Koolivand Salooki [a,*], Reza Abedini [b], Hooman Adib [c], Hadis Koolivand [b]

[a] Department of Petroleum, National Iranian South Oil Company, Ahwaz, Iran
[b] Department of Chemical Engineering, Faculty of Chemical Engineering, Tarbiat Modares University, P.O. Box 14115-143, Tehran, Iran
[c] Department of Petroleum Engineering, Petroleum University of Technology, Ahwaz, Iran

ARTICLE INFO

Article history:
Received 25 May 2011
Received in revised form 10 July 2011
Accepted 13 July 2011
Available online 22 July 2011

Keywords:
Amine
Gas sweetening plant
Regeneration column
Artificial neural network

ABSTRACT

In this study, a new approach for the prediction collection outputs of regenerator column in gas sweetening plant is suggested. The experimental input data, including inlet temperatures of reflux, difference between inlet and outlet condenser temperatures, amount of H_2O and inlet amine temperatures and outlet down temperature of tower and amount of reflux as outputs have been used to create artificial neural network (ANN) models. The testing results from the model are in good agreement with the experimental data. The new proposed method was evaluated by a case study in HASHEMI NEJAD gas refinery in KHORASAN of Iran. Design of topology and parameters of the neural networks as decision variables was done by trial and error, high performance efficiency networks was obtained to predict the output parameters of regenerator column.

1. Introduction

Many natural and industrial gases contain H_2S. The presence of H_2S usually prohibits the direct use of these gases because of its toxic properties [1]. H_2S removal, often called gas sweetening, is necessary to avoid corrosion of combustion engines and SO_x generation in the flue gases [2,3]. There are many treating processes available for removal of H_2S from natural gas. These processes include chemical solvents, physical solvents, adsorption processes hybrid solvents and physical separation (membrane) [4]. Chemical solvents chemically react with the acidic gas for removal of impurities. Chemical solvents invariably are provided in aqueous form and effectively operate on gaseous mixtures containing low concentrations of acidic gas. In the past few years, mixed amine solvents for the removal of acid gases have received increased attention [5]. Alkanolamines, such as, monoethanolamine (MEA), diethanolamine (DEA), di-2-propanolamine (DIPA), and *N*-methyldiethanolamine (MDEA) are commonly used. These amine mixtures have been called a variety of names including formulated amines and MDEA based amines [6]. MDEA has a number of properties which make it desirable for broader application [7]:

- high solution concentration (up 50 to 55 wt.%),
- high acid gas loading,
- low corrosion,
- slow degradation rates,
- lower heats of reaction,
- low vapor pressure and solution losses.

Today, artificial intelligence (AI) systems are widely accepted as a technology offering an alternative way to tackle complex and ill-defined problems. Artificial neural networks (ANNs) are information processing systems, and since their inception, they have been used in several areas of engineering applications [8]. The use of such networks can now be found for number prediction such as modeling the greenhouse effect, simulation N_2O emissions from a temperate grassland ecosystem, and assessment of flotation experiments [9]. Neural networks have a better filtering capacity than empirical models because of the micro feature concept, as each node encodes only a micro feature of the overall input–output pattern. The concept of micro feature implies that each node affects the input–output pattern only slightly. Only when all the nodes are assembled together into a single coordinated network do these micro features map the macroscopic input–output pattern [10]. In this study, we used artificial neural networks (ANN) to model the gas sweetening regeneration column. The network was trained using time course data obtained from HASHEMI NEJAD gas refinery of KHORASAN.

* Corresponding author. Tel.: +98 9183532398.
E-mail addresses: Koolivand.m@nisoc.ir (M. Koolivand Salooki), r.abedini@modares.ac.ir (R. Abedini), hooman.a2008@gmail.com (H. Adib), h4ds.kolivand@gmail.com (H. Koolivand).

1383-5866/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.seppur.2011.07.015

Nomenclature

Symbols

n	number of all data
M	number of neuron in the hidden layer
P	number of neuron in the input layer
N	number of input training data
Z	input transformation of activation function
y	network response from output layer neuron(s)
a	coefficient of sigmoidal activation function
O	experimental value
T	predicted value
W	weight
x	input Parameter

Greeks

φ	activation function (basis function)
α	linear parameter
β	non-linear parameter
η	learning rate
α^*	best-fit of linear parameter
β^*	best-fit of non-linear parameter
θ	threshold (bias)
σ^2	variance of experimental data

Abbreviations

ANN	artificial neural networks
AI	artificial intelligence
MEA	monoethanolamine
DEA	diethanolamine
DIPA	di-2-propanolamine
MDEA	*N*-methyldiethanolamine
MLP	multilayer perceptron
RBF	radial basis function
ADALINE	adaptive linear neuron
ANFIS	adaptive network based fuzzy inference system
MSE	mean square error
NMSE	normalized mean-squared error
MAE	mean absolute error
RMSE	root mean square error
R^2	coefficient of determination

2. Gas sweetening plant

2.1. Natural gas processing

2.1.1. General

Natural gas can be hard to find since it is usually trapped in porous rocks deep underground. After natural gas comes out of the ground, it goes to a processing plant where it is cleaned of impurities and separated its various components. Approximately 90% of natural gas is composed of methane, but it is also contains other gases such as propane and butane [3]. Hydrogen sulfide is a common reduced sulfur compound found in several industrial waste gases. It is easily recognizable by its offensive rotten eggs odor [10]. The removal of the acid gases from hydrocarbon gas mixtures is carried out mainly by means of a chemical reaction rather than only physical absorption [4]. Today, amine processes are used widely for gas sweetening. Other methods, such as carbonate processes, solid bed absorbents, and physical absorption, are employed in the other sweetening plants.

2.1.2. Process description

There are many chemical processes for natural gas sweetening. At present, the amine process (also known as the Girdler process), is the most widely used method for H_2S removal. The process is available as follow [11]:

$$2RNH_2 + H_2S \rightarrow (RNH_3)_2S,$$

where R is mono, di, or tri-ethanol and N, H and S denotes nitrogen, hydrogen and sulfur, respectively.

The amine treating unit removes CO_2 and H_2S from sour gas and hydrocarbon streams in the amine contactor. The amine (MDEA) is regenerated in the amine regenerator, and recycled to the amine contactor. The rich amine surge drum allows separation of hydrocarbon from the amine solution. Condensed hydrocarbons flow over a weir and are pumped to the drain. The rich amine from the surge drum is pumped to the lean/rich amine exchanger. The stripping of H_2S and CO_2 in the amine regenerator regenerates the rich amine solution. The amine regenerator reboiler supplies the necessary heat to strip H_2S and CO_2 from the rich amine, using steam as the heating medium [11]. Fig. 1 shows diagram of gas sweetening regeneration column.

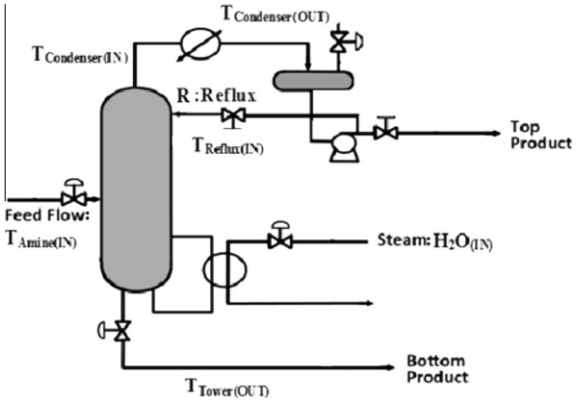

Fig. 1. Diagram of regenerator column.

3. Theory and calculations

3.1. Artificial neural networks

Artificial neural network (ANN) is an empirical modeling tool, which is analogous to the behavior of biological neural structures [12,13]. ANN has also been developed as a generalization of mathematical models of human cognition and neural biology. The commonest type of artificial neural network consists of three layers of units: a layer of "input" units is connected to a layer of "hidden" units, which is connected to a layer of "output" units (see Fig. 2).

Fig. 2 illustrates a typical full-connected network configuration. Such an ANN consists of a series of layers with a number of nodes. As one of the most widely implemented neural network topologies, in this paper, the multilayer perceptron (MLP) is employed [14,15]. When the MLP model is applied to forecast the performance of the regenerator column, it can reveal the highly non-linear relationship between the input parameters and output parameters, by searching an optimal weight in its weighting space. The optimal weights of MLP model store the information, which can best represent such highly non-linear relationships. Mathematically, searching the optimal weight or training the MLP model aims to

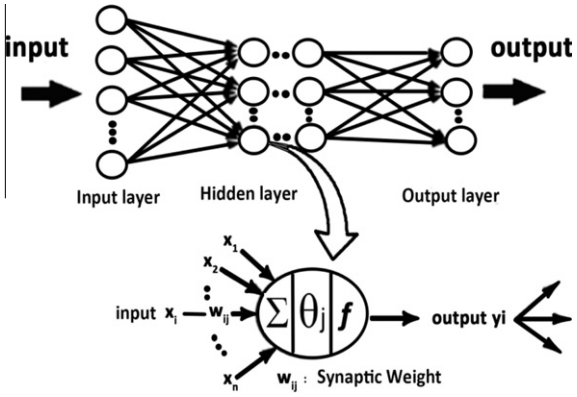

Fig. 2. Architecture of the ANN model [8].

minimize a cost function with respect to the training data set [16,17].

There are multitudes of different types of ANNs. Some of the popular include the multilayer perceptron (MLP), which is more popular and generally trained with the back-propagation of error algorithm, radial basis function (RBF), adaptive linear neuron (ADALINE) and ANFIS (adaptive network based fuzzy inference system). Some ANNs are classified as feed-forward, while others are recurrent, depending on how data is processed through the network. Another way of classifying ANN types is by their method of learning, as some ANNs employ supervised training, while others are referred to as unsupervised or self organizing [12,17]. As illustrated earlier by [17], all feed-forward neural networks (such as MLP and RBF) can be represented by the following equation:

$$\hat{y}(x,\alpha,\beta) = \sum_{j=1}^{M} \alpha_j \varphi_j(x,\beta_j), \tag{1}$$

where $\varphi_j(\cdot)$ can be chosen as any arbitrary non-linear function. The model is always linear with respect to α_js but may be non-linear with respect to the β_js Fig. 3 represents a feed-forward neural network with a single hidden layer for a Multiple Input Single Output (MISO) system.

For the case of multiple independent variables, the non-linear parameters β_js are used to transform the input vector into a scalar argument for the basis function $\varphi_j(\cdot)$ once the input transformation is specified; it remains to choose the functional form of each basis function with respect to its scalar argument. With a specified input transformation of $z_j(x, \beta)$ and a specified functional form for $\varphi_j(\cdot)$, the objective is to find a set of optimal (best-fit) linear α^* and non-linear β^* parameters which minimize a suitably defined merit function for a given set of observations.

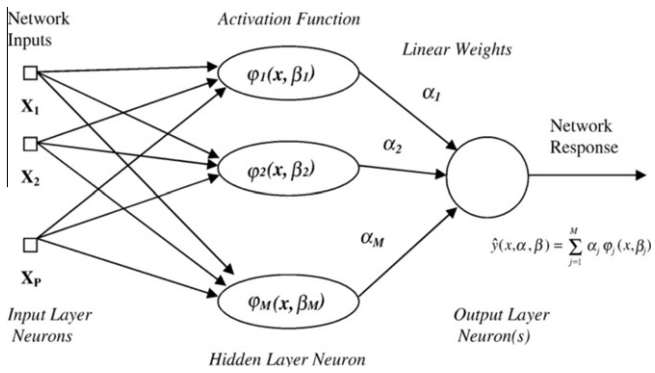

Fig. 3. Architecture of the three-layered feed-forward neural network with a single hidden layer for a MISO system [12].

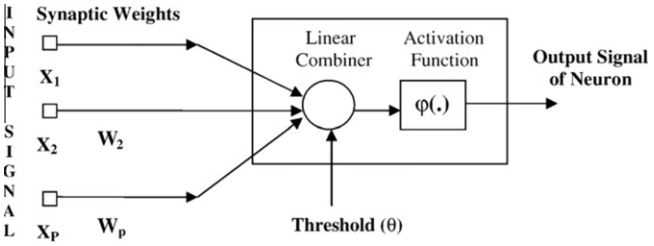

Fig. 4. Schematic representation of a projection based neuron [14].

Wang et al. [18] showed that a feed-forward neural network with two hidden layers employing sigmoidal basis functions is capable of approximating many non-linear functions. The sigmoidal basis function is defined as below:

$$\text{Input transformation} : z = \sum_{j=1}^{P} \beta_j x_j + \beta_0 \tag{2}$$

$$\text{Equation} : \varphi(z) = \frac{1}{1+\exp(-\alpha z)} \tag{3}$$

In MLP neural networks, the basic element is the artificial neuron shown in Fig. 4 which performs a simple mathematical operation on its inputs. The input of the neuron consists of the variables x_1,x_2,...x_p and a threshold (or bias) term. Each of the input values is multiplied by a weight, W_i after which the results are added with the bias term. On the result, a known activation function φ performs a pre-specified (non-linear) mathematical operation.

MLP networks may consist of many neurons ordered in layers. The neurons in the hidden layers do the actual processing, while the neurons in the input and output layer merely distribute and collect the signals. Although, many hidden layers can be used, however, one hidden layer networks are more popular for practical applications due to their simple structures [19,20].

The MLP network is trained by adapting the synaptic weights using a back-propagation technique or any other optimization procedure. During training phase, the network output is compared with a desired output. The error between these two signals is used

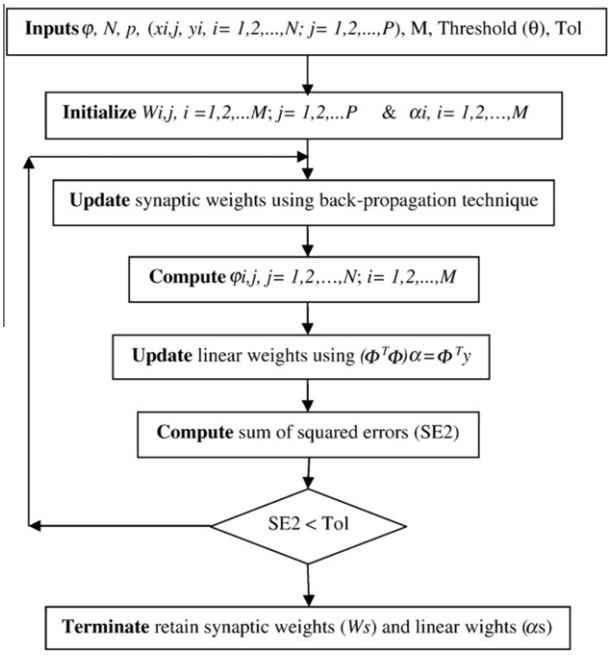

Fig. 5. Learning procedure for training MLP networks [21].

to adapt the weights. This rate of adaptation may be controlled by a learning rate (η). A high learning rate will make the network adapt its weights quickly, but will make it potentially unstable [21]. Setting the learning rate to zero will make the network keep its weights constant.

Additional linear weights (αs), as shown in Fig. 3) were used in this work to accelerate the network convergence. The optimal values of these linear parameters were updated after each iteration of back propagation method using the following equation:

$$(\Phi^T\Phi)\alpha = \Phi^T y, \tag{4}$$

where Φ_i, $j = \varphi(z_i,j)$,$i = 1,...,N$ and $j = 1,...,M$ and y is the $N \times 1$ vector of measured values. The parameters N and M represent number of training data and number of neurons, respectively. Fig. 5 illustrates the training flow chart of such MLP network [22].

Mean square error (MSE), normalized mean-squared error (NMSE), mean absolute error (MAE), root mean square error (RMSE) and coefficient of determination (R^2) for each output were calculated by using Eqs. (5–9) for training and testing data.

$$\mathrm{MSE} = \frac{\sum_{i=1}^{N}(O_i - T_i)^2}{N} \tag{5}$$

$$\mathrm{NMSE} = \frac{1}{\sigma^2}\frac{1}{N}(O_i - T_i)^2 \tag{6}$$

$$\mathrm{MAE} = \frac{1}{\sigma^2}\frac{1}{N}\sum_{i=1}^{N}|O_i - T_i| \tag{7}$$

$$\mathrm{RMSE} = \sqrt{MSE} \tag{8}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(O_i - T_i)^2}{\sum_{i=1}^{N}(O_i - T_m)^2} \tag{9-a}$$

$$T_m = \frac{\sum_{i=1}^{N} O_i}{N}, \tag{9-b}$$

where O_i is the ith experimental value, T_i is the ith predicted value, N is the number of data, σ^2 is the variance of experimental data.

4. Regeneration column modeling

4.1. Gas refinery regeneration column

In this work the prototype refinery is located in the SARAKHS gas field, Iranian onshore. The total number of data's acquired at the time of this study was added up to 145 data. All the above data's have been collected during 1 year. Therefore, the operational data of almost all the days were available. The entire regeneration column's data obtained from this gas refinery were inlet temperatures of reflux, difference between inlet and outlet condenser temperatures, amount of H_2O and inlet amine temperatures, outlet down temperatures of tower and amount of reflux. Table 1 shows the ranges of the used database for the model.

The database to be introduced to the neural network was broken down randomly into three groups: training, cross-validation, and verification. The network was trained using the training set data. The actual output of the training set data was used to develop the weights in the network. At the established intervals, the test set was used to evaluate the predictive ability of the network. The cross-validation set also insured that the network would not memorize (over fitting) the data which means a tendency for all the new data to be regarded as identical to the training data. Training continued as long as the computed error between the actual and predicted outputs for the test set was decreased. Typically, 80% of the data is used for training and validation purposes. The other 20% of the data is categorized as verification. The verification set is used to evaluate the accuracy of the newly trained network by providing the network a set of data it has never seen There is possibility of using the current network weights or using the best network weights saved during a genetically training trial run. Note if a cross-validation set is used during the training, the best network weights are the ones that give the minimum cross-validation error. Otherwise, the best network weights are the ones that give the minimum training error. During the testing, the learning is turned off and the chosen data set is fed through the network. The network output is collected and a report is then generated showing the testing results.

4.2. Data analysis

The data was analyzed to establish relevant relationships between the input parameters and output parameter with the aim of evaluating output dirty amine in an absorption column of gas sweetening plant.

Data security is composed of three major factors: secrecy, availability, and integrity [23]. Data secrecy focuses on preventing unauthorized disclosure; data availability deals with methods of preventing denial of authorized access; and data integrity refers to preventing unauthorized modification of data. Data integrity is an important aspect of storage security and reliability, which is prerequisite for most computers, based applications.

In this paper, for detection of data integrity violations during storing or transmission of data's and modeling of the process the original training data's are displayed. For better visualization and proper trend analysis, Fig. 6 shows the three-dimensional plots of training data for output reflux versus feed parameters.

5. Results and discussion

A feed-forward ANN was designed using back propagation training algorithms. Three layers were optimized according to the number of input and output variables and complexity of the problem: input, hidden and output layers. The number of neurons of 18 and 20 for the two hidden layers of each network were

Table 1
The range of used data for this work.

Measures parameters		Min	Max
$T_{reflux(in)}$	Inlet reflux temperature (°C)	36.25	59.75
$T_{condenser(in-out)}$	Inlet, outlet condenser temperature difference (°C)	44.5	64.58
$H_2O_{(in)}$	Inlet H_2O (m^3/h)	7.87	13.43
$T_{amine(in)}$	Inlet amine temperature (°C)	99.67	104.33
$T_{tower(out)}$	Outlet down temperature of tower (°C)	116.5	121
R	Volume flow rate of reflux (m^3/h)	21.25	25.42

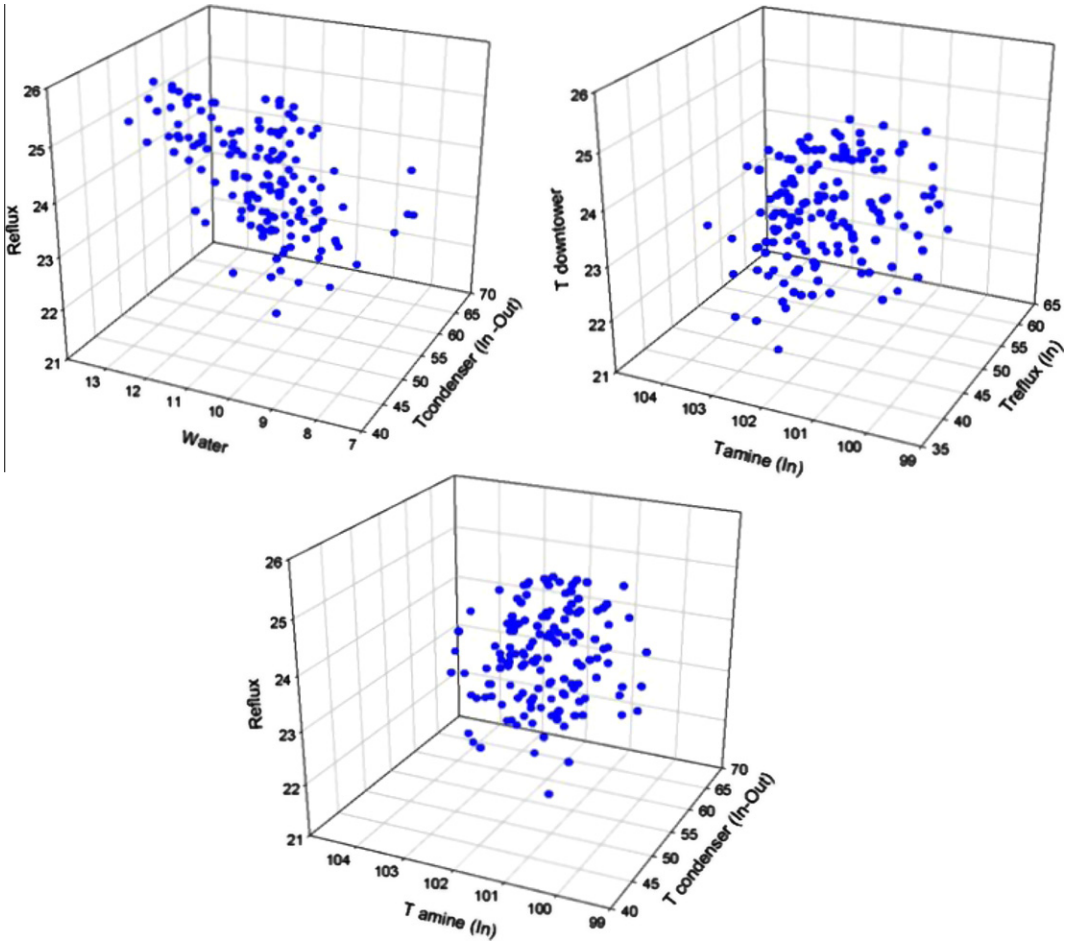

Fig. 6. Desired Parameters value versus input parameters of used training data.

obtained. For the best results from the algorithms, the input data were normalized to the range [−1,+1] by the 'minmax' Matlab function before the training. Sigmoid 'tansig' for input-hidden, hidden-hidden layers and lineal 'purelin' for output layers were the applied transfer functions. The 'trainrp' was the best training function found; this function updates weight and bias values according the 'resilient back propagation algorithm.

Inlet temperatures of reflux, difference between inlet and outlet temperatures of condenser, amount of H_2O and inlet amine temperatures are inputs to the network and outlet down temperatures of tower and amount of reflux are the outputs of the network. For these two outputs, make two distinctive networks. For the first output, down temperature of tower, the optimal number of neurons of one hidden layers network using trial and error method is shown in the Table 2. The training results for the ANN on the cross-validation data showed the lowest MSE when the number of hidden neurons was 20. As we know the basic method of training a neural network is trial and error, if the accuracy of the network declines another network with different specification is tried. In the first network a 4-10-10-1 architecture provide the best model for temperature prediction in terms of MSE, since by trial and error we got a two-layer network which has the least cross validation error of 0.0429392 and an acceptable correlation coefficient of 0.94498, between the predicted and operational data points. So, it means that in 4-10-10-1 architecture there are 4, 10, and 1 neurons in input layer, hidden and output layers, respectively. The results of the first output trial and error are shown in the Table 2.

Table 2
ANN designing with different neurons in the hidden layer of first output (down temperature of tower).

Number of ANN 2 layers	Number of PEs in hidden layer 1	Number of PEs in hidden layer 2	Cross-validation error (MSE)
1	7	7	0.706425
2	8	8	0.138578
3	**10**	**10**	**0.0429392**
4	11	11	0.137071
5	13	13	0.198581
6	14	14	0.0965167
7	15	15	0.290382
8	17	17	0.100615
9	19	19	0.0947325

For the second output, amount of reflux, the optimal number of neurons of one hidden layer network using trial and error method is shown in the Table 3. In the second network, by trial and error and changing the different parameters, a 4-9-9-1 architecture provides the best model for reflux prediction in terms of MSE. Since this network has got the least cross validation error of 0.0356498 and the best correlation coefficient of 0.92312, it can be considered the best network in our trial and error test. It means that the lowest MSE was obtained when the numbers of the hidden neurons were 18. In 4-9-9-1 architecture there are 4, 9, and 1 neurons in input layer, hidden and output layers, respectively.

Performance efficiency of the network was evaluated using the measured and ANN estimated value. Table 4 reports the ANN

Table 3
ANN designing with different neurons in the hidden layer of second output.

Number of networks	Number of PEs in the first hidden layer	Number of PEs in the second hidden layer	Cross-validation error (MSE)
1	6	6	0.680435
2	8	8	0.505392
3	10	10	0.487391
4	12	12	0. 436498
5	**15**	**15**	**0.406707**
6	17	17	0.453533
7	20	20	0.555357
8	25	25	0.681704
9	29	29	0.777579

Table 4
ANN performance values.

Performance	Down temperature of the tower	Amount of reflux
MSE	0.0429392	0.0356498
RMSE	0.2072177	0.1888111
MAE	0.2061	0.186314
NMSE	0.654214	0.435612
r^2	0.9498	0.9231

performances in terms of mean squared error (MSE), normalized mean squared error (NMSE), mean absolute error (MAE), and the correlation coefficient (r^2) between the real targets and neural network outputs. In brief, the ANN predictions are optimum if r^2, MAE, NMSE and MSE are found to be close to 1, 0, 0, and 0, respectively.

In statistics, the mean square error or MSE of an estimator is one of many ways to quantify the difference between an estimator and the true value of the quantity being estimated. The MSE of an estimator $\hat{\theta}$ with respect to the estimated parameter θ is defined as Eq. (10):

$$\mathrm{MSE}\left(\hat{\theta}\right) = E\left[\left(\hat{\theta} - \theta\right)^2\right] \tag{10}$$

The MSE is equal to the sum of the variance and the squared bias of the estimator (Eq. (11)).

$$\mathrm{MSE}\left(\hat{\theta}\right) = \mathrm{Var}\left(\hat{\theta}\right) + \left(\mathrm{Bias}\left(\hat{\theta}, \theta\right)\right)^2. \tag{11}$$

When MSE approaches to zero it means that the error of our network decreases. Two parameters namely correlation coefficient (r^2) and root mean square error (RMSE) values were used for the performance evaluation of the models and comparison of the results for prediction data points. The higher value of correlation coefficient (Eq. (9-a)) and a smaller value of RMSE (Eq. (8)) mean a better performance of the model. According to Eq. (6) standard measure of fit for the neural network is given by the normalized mean-squared error (NMSE), that is the mean-squared error divided by σ^2, the empirical variance of the simulation values calculated over all the N training patterns [24]. Where O_i is the desired output for the training data or cross-validation data i, T_i is the network output for the training data or cross-validation data i, and n is the number of data in the training data set or the cross validation data set. MAE means mean absolute error is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The mean absolute error (MAE) is given by Eq. (7). As the name suggests, the mean absolute error is an average of the absolute errors $e_i = O_i - T_i$, where O_i is the prediction and T_i the true value. As this error approaches to zero, the network is our satisfaction.

In the present study, MSE was used only for the estimation of the network training performance, whereas r^2, MAE, NMSE were used to measure the prediction performance of ANN on the validation data set.

In Figs. 7 and 8 the desired values (down temperatures of tower and volume rate of reflux) versus ANN predictions to verification data points are shown. These figures reveal that an acceptable agreement (correlation coefficient are 0.94498 for down temperatures of tower and 0.92312 for reflux) between the predicted and operational data can be achieved. Ultimately the optimal evolved network had 20 nodes in two hidden layers for the first network (network of down temperature) for 900 epochs, max_fail = 900, mem_reduc = 1, mu = 0.001, mu_dec = 0.2, mu_Inc. = 10 and mu_max = 1E10. For the second one (network of reflux), the optimal evolved network had 18 nodes in just one hidden layer, number of epochs = 1000, max-fail = 1000, mem_reduc = 1, min_grad = 1e-10, mu = 0.001, mu_dec = 0.1, mu_Inc. = 8 and mu_max = 1E10.

As indicated in Fig. 7 predicted down temperatures of tower using neural network follows experimental results in test process appropriately. It is obvious that predicted down temperatures of tower using neural network and experimental results are found in a perfect match for the train data and just in some points has a little deviation. The same result for volume rate of reflux was found. As it is shown in Fig. 8 the good agreement between network outputs and experimental is achieved and both data's series cover each other appropriately.

Evidently, all hyper-surfaces generated by MLP networks pass through each and every training data point. The corresponding generalization performances of these networks as shown in Fig. 9 show some small but unrealistic oscillations. These fluctuations are due to the noise content of the training data and can be

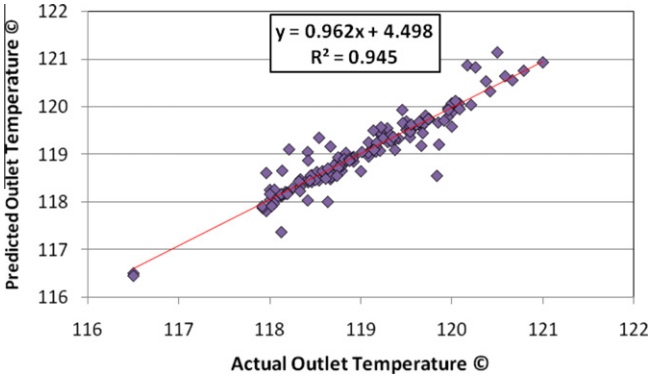

Fig. 7. Down temperatures of tower measurements versus network predictions for the test data.

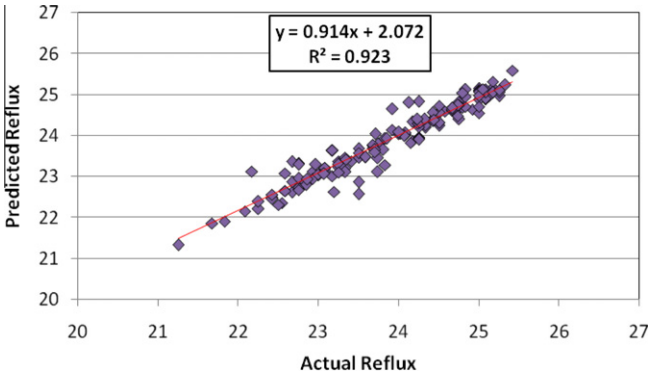

Fig. 8. Reflux value measurements versus network predictions for the test data.

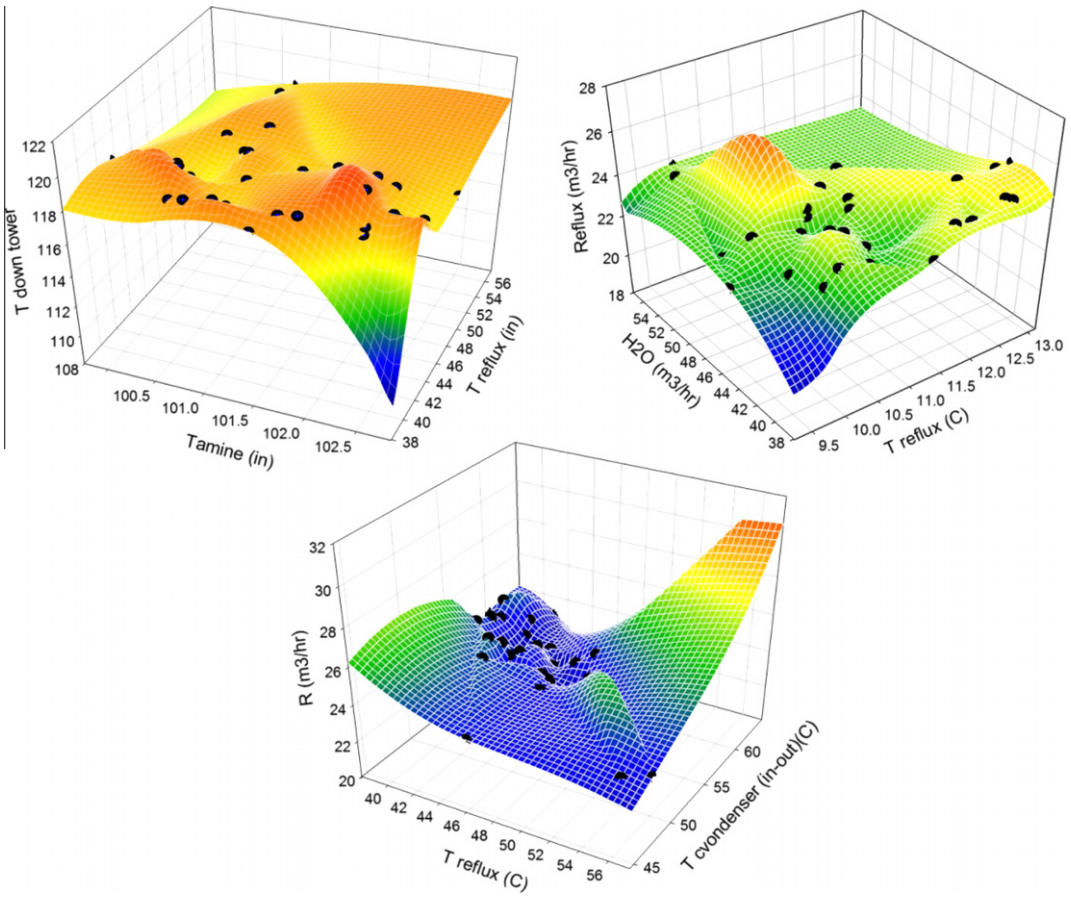

Fig. 9. Generalization of best-fitted surfaces predicted with ANN models.

alleviated if the learning algorithm is equipped with some proper noise filtering facility (as in RBF networks).

Furthermore, the least square merit function of MLP networks usually contain several local minima, therefore, the performances of these networks strongly depend on the initial choice of synaptic weights.

Fig. 9 illustrates the generalization performance of the best fitted surfaces (according to coefficient of determination). As it is demonstrated in Fig. 9 These ANN models fit experimental data appropriately.

The Figs. 10–13 shows that the accuracy of predicted value of the regenerator column outputs is excellent. Fig. 10 provides a visual depiction of the comparison of actual temperatures and network prediction data points versus volume flowrate of water, as it is shown in this figure the predicted and actual down temperature of the tower cover each other appropriately and have a same trend with respect to increasment of H_2O flowrate. Fig. 10 shows the actual down temperatures of tower that were measured in the Khangiran gas refinery and network's prediction in *y*-axis and inlet H_2O in *x*-axis. This figure reveals that an acceptable agreement (correlation coefficient are 0.94498 for down temperatures of tower) between the predicted and operational data can be achieved.

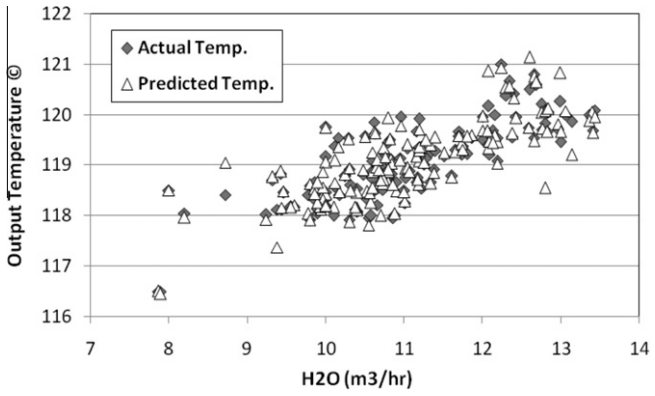

Fig. 10. Comparison of actual temperatures and network prediction data points.

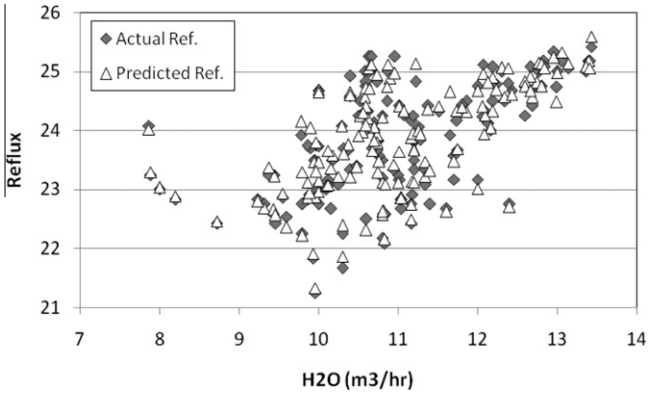

Fig. 11. Comparison of actual reflux and network prediction data points.

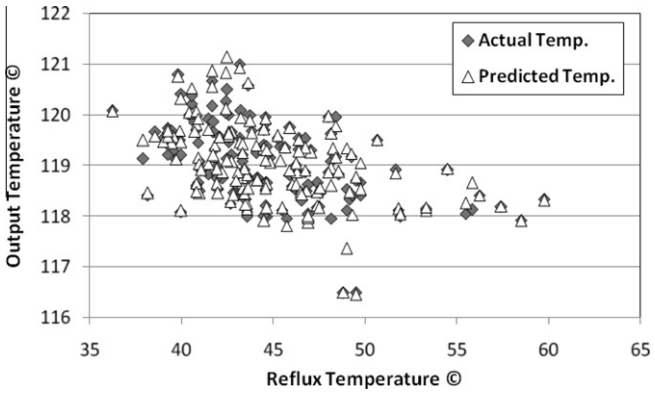

Fig. 12. Comparison of actual down temperatures of tower and network prediction data point.

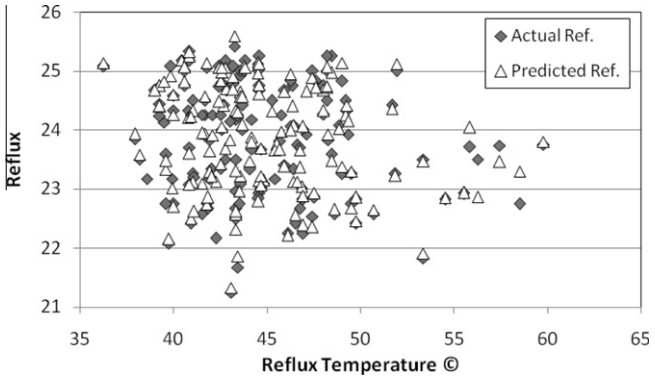

Fig. 13. Comparison of actual reflux and network prediction data points.

Fig. 11 indicates that there are not significant differences between the ANN prediction and the references values of reflux flowrate. As it is clear in this figure comparison of actual reflux and network prediction data points of reflux they have same behavior by increasing water flowrate.

In order to evaluate the capabilities of the ANN model, In Fig. 11 the desired values (reflux) and network prediction versus inlet H_2O are shown, and again this figure showed that an acceptable agreement (correlation coefficient are 0.9231 for amount of reflux) between the predicted and operational data can be achieved.

Fig. 12 shows the actual down temperatures of tower and network's prediction versus reflux temperatures and Fig. 13 shows the actual reflux and its network's prediction versus reflux temperatures. The model was presented with new randomized data that were part of the training data set. The randomized estimated by ANN model are compared with the actual values of actual down temperatures of tower and network prediction data points. Fig. 12 allows easy visual comparison between the actual down temperatures of tower and network prediction data points values. As it is clear these data's are in good agreement with each other and they have a same behavior with reflux temperature increasment. The actual reflux flowrate of regenerator column compared with the ANN model predictions. The results are plotted against the reflux temperature. Comparing the graphs in Fig. 13, it can be seen that ANN model gives very reliable estimates of reflux flowrate of the regenerator column. Also, the values of the predicted reflux flowrate have same behavior with respect to reflux temperature. Again these figures are stamps of approval that the correlation coefficients for actual down temperature of tower and actual amount of reflux versus their network predictions are satisfactory.

6. Conclusion

This paper demonstrates the applicability and feasibility of artificial neural network (ANN) to predict the output parameters of gas sweetening regeneration column. A well-trained and tested ANN by a lot of data that was measured in Khangiran gas refinery was employed. It must be mentioned that the model was evaluated for the first time using experimental data. As the model has a good consistency with the experimental data and also with ANN, it can be concluded that the ANNs with the structured presented here can be used for the prediction of output parameters of the regeneration column. In general, comparison of the artificial neural network model and experimental data shows that ANN simulation presented in this article can be used for accurate prediction of output parameters of regeneration column.

Acknowledgment

The authors would like to thank NIOC and National Iranian South Oil Company (NISOC) for financial support of this project.

References

[1] A.L. Kohl, R.B. Nielsen, Gas Purification, fifth ed., Gulf Publishing Co., Houston, 1997. pp. 60–79.
[2] R. Abedini, M. Koolivand Salooki, S. Ghasemian, Modeling and simulation of condensed sulfur in catalytic beds of Claus process: rapid estimation, Chem. Eng. Res. Bull. 14 (2010) 110–114.
[3] M. Fortuny, J.A. Baeza, X. Gamisans, C. Casas, J. Lafuente, M.A. Deshusses, D. Gabriel, Biological sweetening of energy gases mimics in biotrickling filters, J. Chemosphere 71 (2008) 10–17.
[4] N. Seqatoleslami, M. Koolivand Salooki, N. Mohamadi, A neural network for the gas sweetening absorption column using genetic algorithm, Pet. Sci. Technol. 29 (2011) 1437–1448.
[5] J.C. Polasek, G.A. Iglesias-Silva, Using Mixed Amine Solutions for Gas Sweetening, Bryan Research and Engineering Inc., Technical Paper, (2006), pp. 1–12.
[6] J.A. Bullin, J.C. Polasek, S.T. Donnelly, The use of MDEA and mixtures of amines for bulk CO_2 removal, Proc. of 69th Gas Processors Association Convention, New Orleans, Louisiana, 1984.
[7] K. Dincer, S. Tasdemir, S. Baskaya, B.Z. Uysal, Modeling of the effects of length to diameter ratio and nozzle number on the performance of counterflow Ranque–Hilsch vortex tubes using artificial neural networks, J. Appl. Therm. Eng. 28 (2008) 2380–2390.
[8] E. Jorjani, S. Chehreh-Chelgani, Sh. Mesroghli, Prediction of microbial desulfurization of coal using artificial neural networks, J. Miner. Eng. 20 (2007) 1285–1292.
[9] S. Satish, Y. Pydi Setty, Modeling of a continuous fluidized bed dryer using artificial neural networks, Int. Comm. Heat. Mass. 32 (2005) 539–547.
[10] R.N. Maddox, J. Diers, A.M. Bhairi, P.A. Thomas-Cooper, E.M. Elizondo, Plant/Oper. Prog. 6 (1987) 112–117.
[11] H.M. Yao, H.B. Vuthaluru, M.O. Tade, D. Djukanovic, Artificial neural network-based prediction of hydrogen content of coal in power station boilers, Fuel 84 (2005) 1535–1542.
[12] S. Ashoori, A. Abedini, R. Abedini, Kh. Qorbani Nasheghi, Comparison of scaling equation with neural network model for prediction of asphaltene precipitation, J. Pet. Sci. Eng. 72 (2010) 186–194.
[13] C. Stergiou, D. Siganos, Neural Networks, Surprise 96 Journal, vol. 14, Imperial College of Science, Technology and Medicine, London, 1996.
[14] R. Abedini, I. Zanganeh, M. Mohagheghian, Simulation and estimation of vapor–liquid equilibrium for asymmetric binary systems (CO_2 – Alcohols) using artificial neural network, J. Phase. Equilibr. Diff. 32 (2011) 105–114.
[15] T. Poggio, F. Girosi, Regularization algorithms for learning that are equivalent to multilayer networks, Science 247 (1990) 978–982.
[16] B. Rahmanian, S.A.A. Mansoori, R. Abedini, M. Pakizeh, Application of experimental design approach and artificial neural network (ANN) for the determination of potential micellarenhanced ultrafiltration process, J. Hazard. Mater. 11 (2010) 135–143.
[17] B.R. Bakshi, U. Utojo, Unification of neural and statistical modeling methods that combine inputs by linear projection, Comp. Chem. Eng. 22 (1998) 1859–1878.
[18] X. Wang, R. Luo, H. Shao, Designing a soft sensor for a distillation column with the fuzzy distributed radial basis function neural network, Proc. IEEE Conf. Decis. Cont. 2 (1996) 1714–1719.
[19] S.P. Niculescu, Artificial neural networks and genetic algorithms in QSAR, J. Mol. Struct. 622 (2003) 71–83.
[20] R. Abedini, M. Esfandyari, A. Nezhadmoghadam, B. Rahmanian, Prediction of undersaturated crude oil viscosity: artificial neural network and fuzzy model approach, Pet. Sci. Technol. (2011).

[21] R. Abedini, A. Abedini, Development of an artificial neural network algorithm for prediction of asphaltene precipitation, Pet. Sci. Technol. 29 (2011) 1565–1577.
[22] F. Torabi, A. Abedini, R. Abedini, The development of an artificial neural network model for prediction of crude oil viscosities, Pet. Sci. Technol. 29 (2011) 804–816.
[23] C.P. Eeger, Security in Computing, second ed., Prentice-Hall, Englewood Cliєs, NJ, 1997. pp. 45–57.
[24] A. Aussem, D. Hill, Neural-network metamodelling for the prediction of Caulerpa taxifolia development in the Mediterranean sea, Neurocomputing 30 (2000) 71–78.