Songklanakarin J. Sci. Technol.
33 (5), 531-538, Sep. - Oct. 2011

SJST SONGKLANAKARIN JOURNAL OF SCIENCE AND TECHNOLOGY
http://www.sjst.psu.ac.th

Original Article

Predicting the supercritical carbon dioxide extraction of *oregano bract* essential oil

Abdolreza Moghadassi [1,*], Sayed Mohsen Hosseini [1], Fahime Parvizian [1], Ibrahim Al-Hajri [2], and Mehdi Talebbeigi [3]

[1] *Department of Chemical Engineering, Faculty of Engineering, Arak University, Arak 38156-8-8349, Iran.*

[2] *Department of Chemical Engineering, College of Technological Studies, Shuwaikh, Kuwait.*

[3] *Research and Development Center, Shazand Oil Refinery, Shazand, Markazi, Iran.*

Received 17 December 2010; Accepted 28 September 2011

Abstract

The extraction of essential oils using compressed carbon dioxide is a modern technique offering significant advantages over more conventional methods, especially in particular applications. The prediction of extraction efficiency is a powerful tool for designing and optimizing the process. The current work proposed a new method based on the artificial neural network (ANN) for the estimation of the extraction efficiency of the essential oil *oregano bract*. In addition, the work used the back-propagation learning algorithm, incorporating different training methods. The required data were collected; pre-treating was used for ANN training. The accuracy and trend stability of the trained networks were verified according to their ability to predict unseen data. The Levenberg-Marquardt algorithm has been found to be the most suitable algorithm, with the appropriate number of neurons (i.e., ten neurons) in the hidden layer and a minimum average absolute relative error (i.e., 0.019164). In addition, some excellent predictions with maximum error of 0.039313 were observed. The results demonstrated the ANN's capability to predict the measured data. The ANN model performance was also compared to a suitable mathematical model, thereby confirming the superiority of the ANN model.

Keywords: Artificial Neural Network, Prediction, Supercritical extraction, Efficiency, *Oregano bract* essential oil.

1. Introduction

The extraction of essential oils using compressed carbon dioxide is a modern technique providing significant advantages over more conventional methods, such as steam distillation and liquid solvent extraction. The ability to predict the extraction process path, based on knowledge of the process parameters, is a powerful tool for both designing and optimizing a carbon dioxide industrial plant (Gaspar *et al.*, 2003). To achieve this goal, mathematical models have been developed by integrating differential mass balances within the solid and fluid phases. These models require sufficient information related to thermodynamics constraints, equilibrium relationships, kinetic constraints, and the mass transfer mechanism for any given temperature, pressure, and condensed phase composition (Izadifar et al., 2006). A method is accepted based on the experimental data as well as its ability to provide additional practical methods in modeling efforts. The models demonstrate dynamic relationships between input and output variables while bypassing the underlying complexity inherent in the system. Statistical models based on regression analysis are examples of such black box model-

* Corresponding author.
Email address: A_Moghadassi@yahoo.com

ing. Most of these common approaches rely on linear system identification models.

Yet the major processes in chemical engineering are nonlinear processes. Consequently, previous approaches have failed to respond appropriately due to process non-linearity. Another promising alternative modeling technique, artificial neural network (ANN), was recently developed using numerous applications in chemical engineering (Zahedi *et al.*, 2009). ANN is an empirical tool; its behavior is analogous to the behavior of biological neural structures. The model is based on experimental results proposed to predict the required data in order to avoid the need for additional experiments (Moghadassi *et al.*, 2009a). ANN has the ability to identify underlying highly complex relationships using input-output data. In fact, such networks define several empirical relations for each data section (Ganguly, 2003). ANN's advantages over classical methods include speed, simplicity, and capacity to learn. More recently, ANNs have been used to address various thermodynamical problems. In the context of thermodynamical data predictions, the use of ANNs signals the need for suggested networks to estimate compressibility factors in vapor and liquid phases (Chouai *et al.*, 2002), to predict the activity coefficient (Petersen *et al.*, 1994; Urata *et al.*, 2002), to provide VLE (Vapor-Liquid-Equilibrium) data for various systems (Habiballah *et al.*, 1994; Guimaraes *et al.*, 1995; Sharma *et al.*, 1995; Ganguly, 2002; Mohanty, 2006; Moghadassi *et al.*, 2009b), and define asphaltene precipitation (Zahedi *et al.*, 2009) as well as black pepper essential oil extraction (Izadifar *et al.*, 2006), among other needs.

A literature review found few studies involving the modeling of the supercritical extraction of essential oils by ANN, but no studies dealing with the prediction of the extraction of the essential oil oregano bract using the ANN model. Neural networks have been widely and successfully applied in various technological disciplines. The ability to learn the behavior of the data generated by a system gives neural networks its versatility (Zahedi *et al.*, 2009). The remaining part of the current study will briefly describe ANNs and then develop multi-layer perceptron networks to estimate supercritical extraction of oregano bract essential oil data. In addition, the best ANN predictor will also be described. Finally, the results from ANN will be compared with a suitable mathematical model.

2. Artificial Neural Network

In order to identify the relationship between the input and output data derived from experimental work, a more powerful method than what has been traditionally used is necessary. The ANN is an especially efficient algorithm to approximate any function using a finite number of discontinuities by defining the relationships between input and output vectors (Hagan *et al.*, 1996; Bozorgmehry *et al.*, 2005; Zahedi *et al.*, 2009). The ANN is considered to be a black box consisting of a series of complex equations for the calculation of outputs based on a given series of input values. These algorithms can learn from the experiments and are also fault tolerant in the sense that they are able to handle noisy and incomplete data. The ANNs effectively deal with non-linear problems; once trained, they can quickly perform predictions and generalizations (Sozen *et al.*, 2004). Such networks have been used to solve complex problems that are difficult-if not impossible-to solve using conventional approaches. More specifically, the networks minimize the difference between the predicted and observed (actual) outputs (Chouai *et al.*, 2002). ANNs are biological inspirations based on various brain functionalities. They are composed of many simple elements (i.e., neurons) that are interconnected by links and act like axons to define the empirical relationship between the inputs and outputs of a given system in which the inputs are independent variables while the outputs include dependent variables. Therefore, it is essential that the user have a good understanding of the system's fundamental science to provide the appropriate input and consequently support the identified relationship (Zahedi *et al.*, 2009).

Figure 1 depicts the multi-layered arrangement of a typical interconnected neural network, which consists of an input layer, an output layer, and one hidden layer-each of which play different roles. Each connecting line has a corresponding weight. ANNs are trained by adjusting these input weights (connection weights) so that the calculated outputs may be approximated according to the desired values. The output from a given neuron is calculated by applying a transfer function to a weighted summation of its input to result in an output, which in turn serves as input to other neurons, as indicated in Equation 1 (Gharbi, 1997):

$$\alpha_{jk} = F_k\left(\sum_{i=1}^{N_{k-1}} w_{ijk}\alpha_{i(k-1)} + \beta_{jk}\right) \quad (1)$$

where α_{jk} is neuron j's output from k's layer and β_{jk} is the bias weight for neuron j in layer k. The model-fitting parameters, w_{ijk}, are known as the connection weights. The non-linear activation transfer functions F_k may emerge in many different forms; the classical forms are threshold, sigmoid,

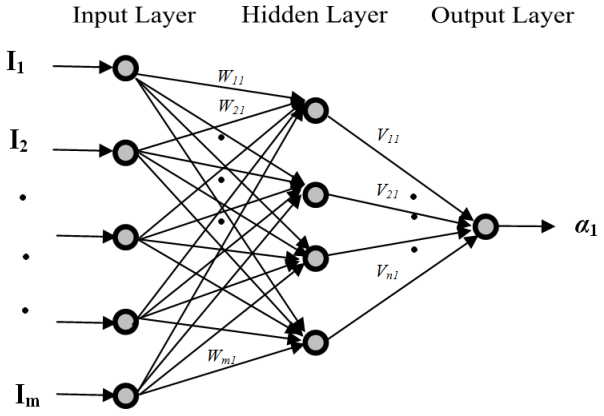


Figure 1. Schematic diagram of a typical multi-layer neural network model.

Gaussian, and linear functions, among others. For more details of various activation functions, see Bulsari (1995).

The training process requires a proper set of data-namely, input (I_i) and target output (t_i). During training, the network's weights and biases are iteratively adjusted to minimize the network error function (Demuth *et al.*, 2002). The typical error function is the Average of Absolute Relative Errors (AARE), as denoted in Equation 2.

$$AARE = \frac{1}{N}\sum_{i=1}^{N} ABS\left(\frac{t_i - \alpha_i}{t_i}\right) \quad (2)$$

Many types of neural networks exist, differing according to their network topologies and/or learning algorithm. One of the most commonly applied ANN is the feed-forward multi-layer ANN that uses a back-propagation learning algorithm. In the current paper, the back propagation learning algorithm is applied to predict the supercritical carbon dioxide extraction of oregano bract essential oil. Back propagation is a multi-layered feed-forward network with hidden layers between the input and output (Osman *et al.*, 2002). The simplest implementation of back propagation learning contains the network weights and biases updates in the direction of the negative gradient by changing the momentum and learning rate based on the behavior of the errors (Zahedi *et al.*, 2009), in which the performance function decreases very quickly. An iteration of this algorithm can be written as follows (Gharbi, 1997):

$$x_{k+1} = x_k - l_k g_k \quad (3)$$

where x_k is a vector of current weights and biases, g_k is the current gradient, is the learning rate, and x_{k+1} is a vector of new weights and biases.

Figure 2 depicts the process detail flowchart used to locate the optimal model. Various back propagation algorithms exist, including the Scaled Conjugate Gradient (SCG), Levenberg-Marquardt (LM), and Gradient Descent with variable learning rate Back propagation (GDX), and Resilient back Propagation (RP). The LM is the fastest training algorithm for networks of a moderate size; its memory reduction feature can be used when the training set is large. The SCG is also one of the most important general purpose back propagation training algorithms (Lang, 2000; Demuth *et al.*, 2002). The neural networks learn to recognize the patterns of the data sets during the training process, teaching the patterns of the data set and thereby enabling the analyst to perform more interesting and flexible work in a changing environment (Moghadassi *et al.*, 2009a). Although the neural network may require extensive time to learn a sudden drastic change, it excels at adapting to constantly changing information. However, the programmed systems are constrained to the designed situation and are not valid in other conditions. In the learning process, several variables influence the ANN training: the number of iterations, learning rate, momentum coefficient, number of hidden layers, and number of hidden neurons. Determining the ideal set of these variables and

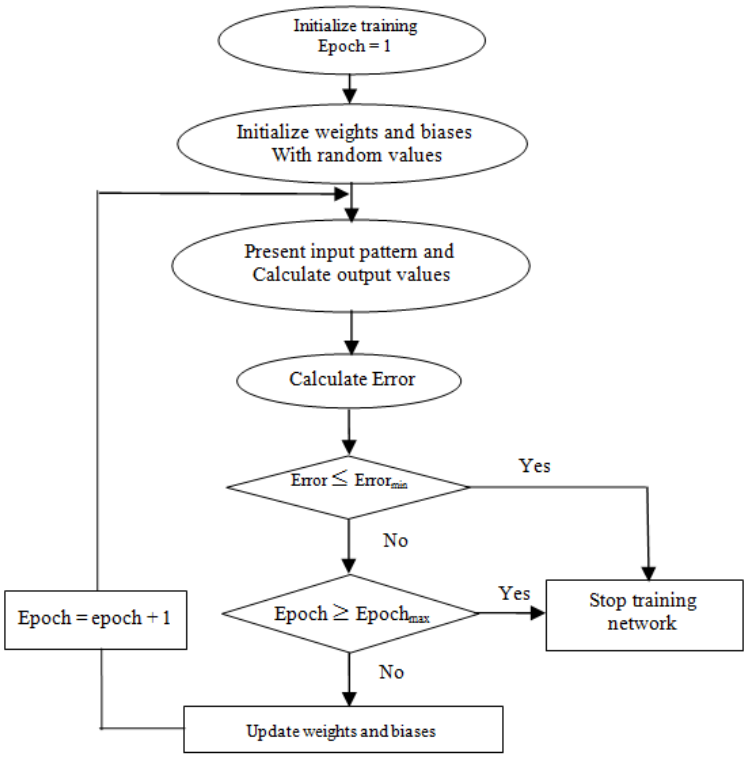


Figure 2. Training process flowchart.

parameters requires varying them all before choosing the best combination (Demuth *et al.*, 2002).

Neural networks build informative models, unlike more conventional models, which fail to do so. Due to the need to handle extremely complex interactions, neural networks can easily model data too difficult to model in classic ways (e.g., inferential statistics or programming logic). The performance of neural networks is at least as good as classical statistical modeling and even better in most cases (Osman et al., 2002). Models built by neural networks are more reflective of the data structure and are significantly faster. Today's neural networks operate well with modest computer hardware. Although neural networks are computationally intensive, the routines have been optimized to the point that they can now run in a reasonable time on personal computers. As such, they no longer require supercomputers, as they did in the early days of neural network research.

This research paper uses multi-layer perceptron networks to estimate the supercritical extraction of oregano bract essential oil data. The developed networks are trained and evaluated using the experimental data reported by a previously published paper (i.e. Gaspar *et al.*, 2003). One part of experimental data was used to train the networks, while the remaining parts were devoted to the evaluation of the networks' performance. The experimental data and corresponding predicted values by ANNs were compared, and the best ANN model was evaluated by comparing it to an appropriate mathematical model.

3. Modeling

3.1 Neural network development to extract an efficient prediction

Developing a neural network model to accurately predict extraction efficiency requires exposing it to a suitable data set during the training phase. The current work used back-propagation learning with one hidden layer network. The Scaled Conjugate Gradient, Levenberg-Marquardt, Resilient back Propagation, and Gradient Descent with variable learning rate Back propagation were implemented according to simulation purposes. As input layer neurons have no transfer functions, the neurons in the hidden layer perform two tasks: summing the weighted inputs connected to them and passing the result through a non-linear activation function to the output or adjacent neurons of the corresponding hidden layer. The network inputs should be selected carefully if the best results are expected to be achieved. The input variables should reflect the underlying physics of the process being analyzed. In the supercritical extraction process, the extraction time, pressure, temperature, flow rate, and size of particles significantly affect extraction efficiency. Therefore, the network model inputs include extraction time, pressure, temperature, material flow rate, and the size of particles; the amount of extraction efficiency is considered as the model output (see Figure 3). Inputs and outputs are normalized between the ranges of 0 to 1. The Logistic Sigmoid and purelin transfer functions were used to build the ANNs, each of which was trained using two thirds of the data set while the remainder of the data was used to evaluate accuracy and trend stability. The comparison was undertaken using statistical methods. The comparison criterion was the average of absolute relative errors (AARE) between net output and training data. The computer program was developed using MATLAB.

3.2 Mathematical modeling applied to extract an efficient prediction

The prediction of extraction efficiency, indentified in the following steps, was undertaken using Gaspar *et al.* (2003) mathematical model. The Fluid Phase/Simple Single Plate (FP/SSP) model contains a detailed description of the fluid phase mass and the particle mass balances. The model combines the fluid phase mass balances because it involves intraparticle diffusion that controls the extraction rate. The FP/SSP model also considers the likely presence of essential oils in the fluid phase at the start of the dynamic extraction. The system pressurization causes some of the non-retained oils to be leached by the solvent into the fluid phase (Gaspar *et al.*, 2003). In addition, the oil is uniformly distributed within the fluid phase and particles. The modeling equation is as follows:

$$E(t) = \frac{M(t)}{M_\infty} = \frac{\int_0^t C(z=L,t)u_s A dt}{M_\infty} \quad (4)$$

where $M(t)$ and M_∞ are the mass of oil extracted from the bed after time (t) and in infinity, u_s is the superficial velocity of the solvent, A is the cross-sectional area of the bed, and $C(z=L, t)$ is the concentration at the exit of the bed. $E(t)$ is the extraction degree of essential oils.

The concentration profile in the fluid phase, C, is obtained from the integration of the fluid phase mass balance, as indicated in Equation 5.

$$\frac{\partial C}{\partial t} = a\frac{\partial^2 C}{\partial z^2} + b\frac{\partial C}{\partial z} + g(t) \quad (5)$$

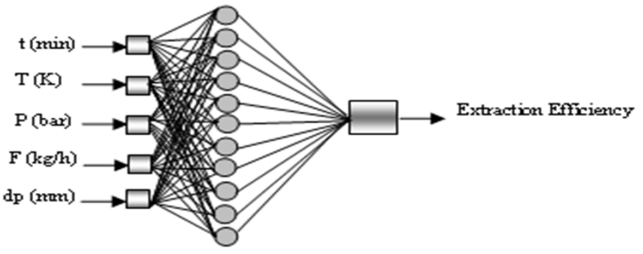


Figure 3. Schematic diagram of the neural network structure for supercritical extraction of essential oil.

where

$$a = \frac{D_{ax}}{\varepsilon} \tag{6}$$

$$b = -\frac{u_s}{\varepsilon} = -u_i \tag{7}$$

$$g(t) = \frac{M_\infty}{AL\varepsilon} \sum_{n=0}^{\infty} \frac{8D_m}{\delta^2} (e^{-D_m t[(2n+1)\pi/\delta]^2}) \tag{8}$$

where z is the axial coordinate of the bed, D_{ax} is the axial dispersion coefficient, ε is the bed porosity, and $g(t)$ is the rate of mass transfer from the particles. The initial and boundary conditions applied to Equation 5 are as follows:

$$C = C_0 \Leftarrow t = 0 \tag{9}$$

$$C = 0 \Leftarrow z = 0 \tag{10}$$

$$C_0 = \frac{f_0 M_\infty}{AL\varepsilon} \tag{11}$$

where C_0 is the initial concentration in the fluid phase and f_0 is the fraction of oil initially dissolved in the fluid phase.

4. Results and Discussions

In this section, various ANN architectures relying on different training algorithms are examined. The optimum performance of networks is empirically obtained by changing the number of neurons in the hidden layer using the trial-and-error method. The minimal number of neurons is sufficient for prediction performance without leading to over-fitting or an unreasonably long computational time. If too few neurons exist in the hidden layer, the performance of the network will not be satisfactory. Conversely, if too many neurons exist in the hidden layer, the convergence will be very slow and may be compromised by local minima or over-fitting. Figure 4 depicts the effects of the number of neurons on the train and test errors in the model. In all cases, errors decreased as the number of neurons in the hidden layer increased due to the higher number of adjustable parameters. Nonetheless, inordinate increases are a cause for over-fitting (increasing the testing errors). As evident in these figures, the optimum number of hidden layer neurons is determined to be eleven for minimum error. Moreover, the LM training algorithm was found to have a superior performance among all the best networks. Therefore, the LM method containing eleven hidden neurons with minimum error is selected as the optimum structure.

In addition, the AARE training of the algorithms (with eleven hidden layer neurons- namely, the best networks) is listed in Table 1. As seen in this table, the Levenberg-Marquardt has the minimum AARE. The trained ANN models were also tested and evaluated against the new data. Table 2 shows the AARE testing of these networks. According to Tables 1 and 2, the LM algorithm provides the best minimum error average (i.e., AARE) in both training and testing. Figure 5 shows the relative error fluctuations in this algorithm.

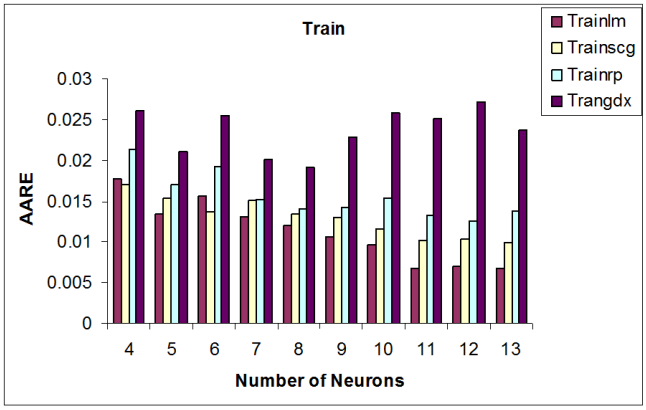
Because the initial weights are selected randomly in the network training, the performance of neural networks trained on the same data set will depend on these values.

Table 1. AARE training of the algorithms for the best network.

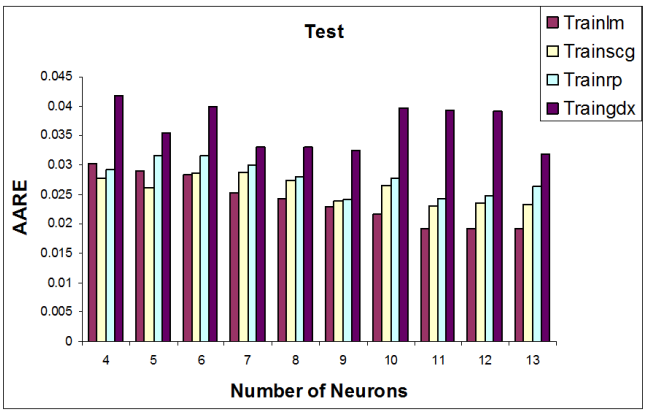
Algorithm	AARE of network Training
trainlm	0.0067543
trainscg	0.010237
trainrp	0.013249
traingdx	0.025152

Table 2. AARE testing of the algorithms for the best network.

Algorithm	AARE of network Training
trainlm	0.019164
trainscg	0.023167
trainrp	0.02434
traingdx	0.039313



(a)



(b)

Figure 4. Determining the optimum number of neurons for the selected algorithms: (a) Networks training, (b) Networks testing.

Therefore, in order to obtain the best network, the training was iterated. The initial weights matrix and the biases that result in minimum error are as follows:

$$I.W. = \begin{bmatrix} 0.0099 & 1.7511 & 0.3518 & -0.7319 & 1.4703 \\ -0.0675 & 0.0900 & 1.6232 & -0.2980 & 0.7320 \\ -10.4413 & -5.7708 & -2.4070 & -10.4114 & -5.9390 \\ -0.8183 & -0.1719 & 0.7770 & -5.0717 & 0.6001 \\ -0.7552 & -2.4575 & 5.4409 & 0.4388 & -7.2859 \\ -0.9447 & -1.2405 & 3.1074 & -6.2169 & 8.4075 \\ 1.1575 & 1.7612 & -1.2325 & -0.3857 & -0.1893 \\ 1.5919 & 8.8967 & -4.6248 & -0.6034 & -0.5668 \\ 5.9545 & 4.4024 & -9.4300 & 3.7825 & 1.4865 \\ -4.7599 & 0.8441 & -0.2862 & -0.3167 & 1.0689 \\ 18.2122 & -1.4739 & 0.1190 & -1.9121 & -0.7775 \end{bmatrix} \quad I.b. = \begin{bmatrix} 1.1629 \\ -0.3549 \\ 7.2220 \\ -1.1323 \\ -6.7744 \\ 7.1504 \\ 0.7801 \\ 3.9673 \\ 4.9087 \\ -4.1849 \\ 15.3825 \end{bmatrix} \tag{12}$$

Figure 6 shows the scatter diagrams that compare the experimental data versus the computed neural network data in both training and testing networks. As evident in the figure, a tight cluster of points around the 45° line occurs for the data points, indicating excellent agreement between the experimental and calculated data. Almost all data fall along this line, which confirms the accuracy of the ANN model.

To verify the performance of the ANN model, its estimation ability was compared with a suitable mathematical model. The results of the mathematical model were compared with the ANN as well as with the experimental data not used in the ANN training. As shown in Figures 7a-d, good agreement occurred between the experimental data and the predicted ANN results. Thus, the ANN has excellent overlap with laboratory experimental data. Figure 8 shows the ANN and mathematical model error rates. The estimated error of ANN is unbelievably low. The obtained ANN model can be updated when new data are available, which can be applied in retraining the ANN using old ANN weights as initial weights for the new ANN. The results indicate that ANN has the best performance with minimum error, which confirms its application in predicting efficiency.

5. Conclusion

This work investigated the ability of ANN in modeling and predicting the supercritical extraction of an essential oil. Specifically, the extraction efficiency in oregano bract was modeled using the MLP neural network architectures, obtaining good agreement with experimental data. An important feature of the model is that it requires no theoretical knowledge or human experiences during the training process. The model was trained based on the experimental data only, with no prior knowledge. All unknown relationships were represented approximately using neural networks instead of traditional relationships. Finally, the ANN ability was compared with a suitable mathematical model and found to be the most accurate model.

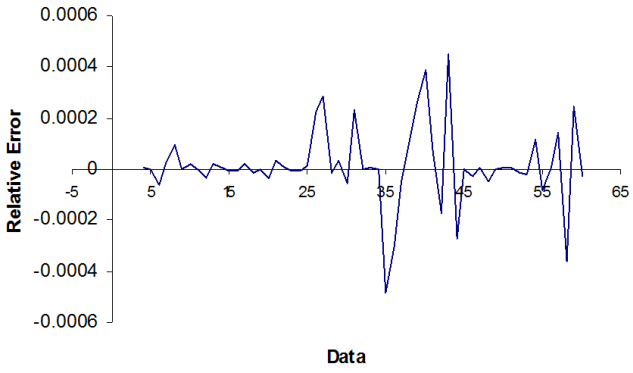
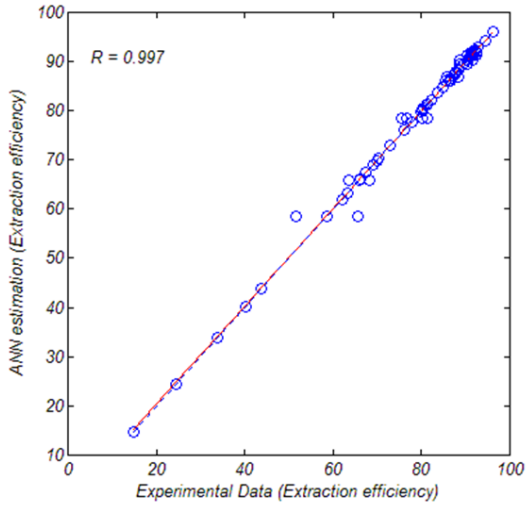
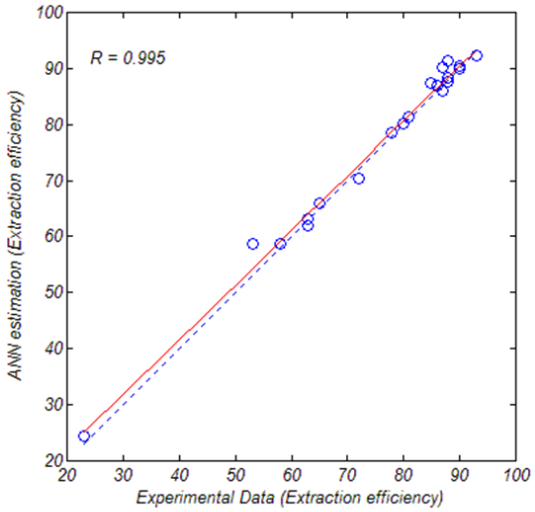


Figure 5. Typically relative error between predicted data by ANN and experimental data for the best network (LM algorithm).



(a)



(b)

Figure 6. Typical evaluation of ANN performance; a scatter plot of typically measured experimental data against the ANN model for (a) training and (b) testing.

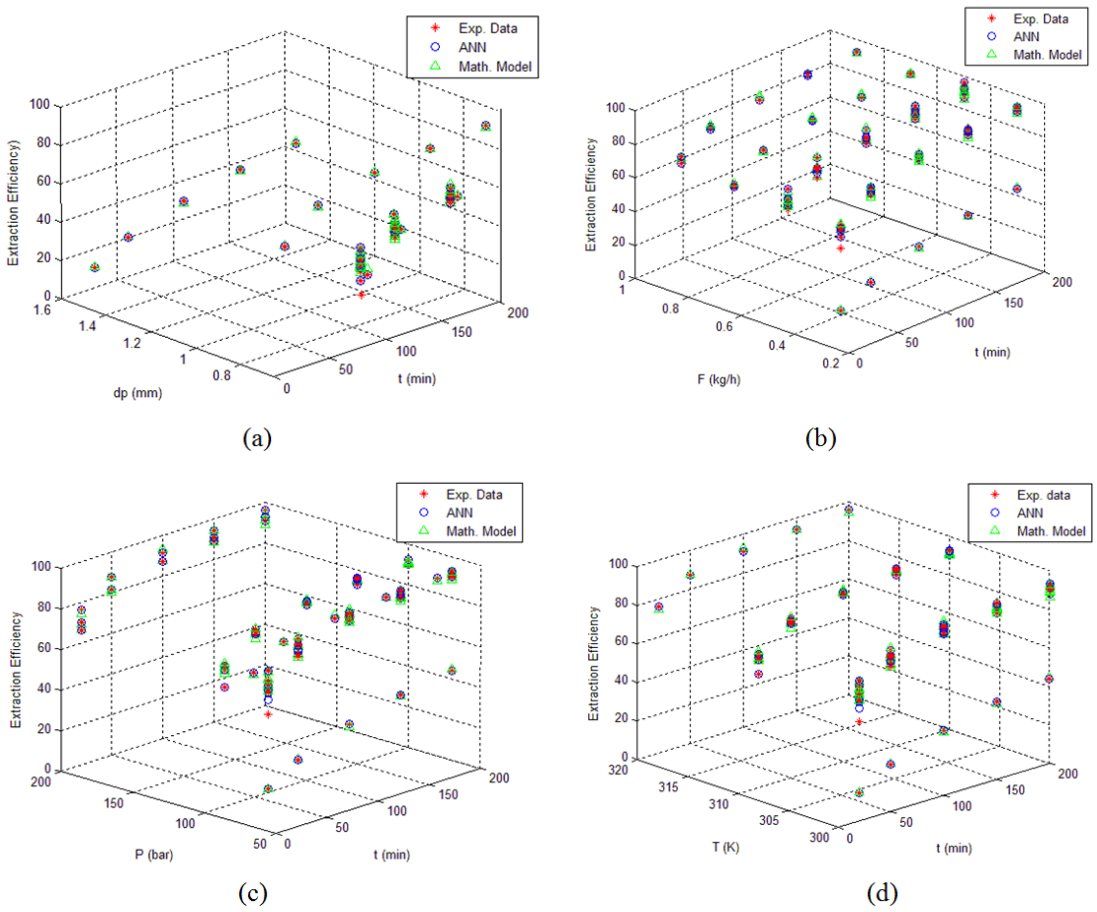


Figure 7. Typically comparison between predicted data by ANN, mathematical model and the experimental data in different conditions.

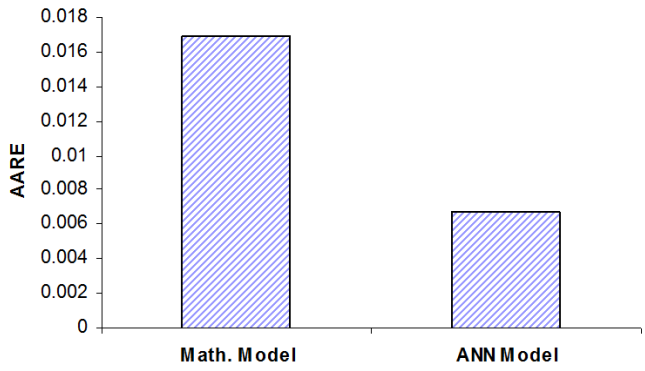


Figure 8. Comparison between relative error of ANN and mathematical models.

Acknowledgment

The authors are very thankful to A. Elkamel from University of Waterloo for frequent discussions and patient help during this research.

References

Bozorgmehry, R.B., Abdolahi, F. and Moosavian, M.A. 2005. Characterization of basic properties for pure properties for pure substances and petroleum fractions by neural network. Fluid phase equilibria. 231, 188-193.

Bulsari, A.B. 1995. Neural Networks for Chemical Engineers, Elsevier, Science Press, Amsterdam, Netherlands.

Chouai, A., Laugier, S. and Richon, D. 2002. Modeling of thermodynamic properties using neural networks: Application to refrigerants. Fluid Phase Equilibria. 199, 53-62.

Demuth, H. and Beale, M. 2002. Neural Network Toolbox User's Guide, the Mathworks Inc, MA.

Gaspar, F., Lu, T., Santos, R. and Al-Duri, B. 2003. Modelling the extraction of essential oils with compressed carbon dioxide. Journal of Supercritical Fluids. 25, 247-260.

Ganguly, S. 2003. Prediction of VLE data using radial basis function network. Computers & Chemical Engineering. 27, 1445-1454.

Guimaraes, P.R.B. and McGreavy, C. 1995. Flow of information through an artificial neural network. Computers & Chemical Engineering. 19, 741-746.

Gharbi, R. 1997. Estimating the isothermal compressibility coefficient of under saturated Middle East crude's using neural networks. Energy and Fuels. 11, 372-376.

Habiballah, W.A., Startzman, R.A. and Barrufet, M.A. 1994. Use of Neural Networks for prediction of vapor liquid equilibrium K-values for light hydrocarbon mixtures. Proceedings of Society of Petroleum Engineers (SPE) annual technical conference and exhibition. Reservoir Engineering 1, 347 (paper no. #28597).

Hagan, M.T., Demuth, H.B. and Beal, M. 1996. Neural Network Design, PWS Publishing Company, Boston, U.S.A.

Izadifar, M. and Abdolahi, F. 2006. Comparison between neural network and mathematical modeling of supercritical CO_2 extraction of black pepper essential oil. Journal of Supercritical Fluids. 38, 37–43.

Lang, R.I.W. 2000. A Future for Dynamic Neural Networks, Department of Cybernetics, University of Reading, U.K.

Moghadassi, A.R., Parvizian, F., Hosseini, S.M. and Fazlali, A.R. 2009a. A New Approach for Estimation of PVT Properties of Pure Gases Based on Artificial Neural Network Model. Brazilian Journal of Chemical Engineering. 26, 199-206.

Moghadassi, A., Parvizian, F. and Hosseini, S.M. 2009b. A New Approach Based on Artificial Neural Networks for Prediction of High Pressure Vapor-liquid Equilibrium. Australian Journal of Basic and Applied Science. 3(3), 1851-1862.

Mohanty, S. 2006. Estimation of vapour liquid Equilibria for the system, carbon dioxide- difluormethane using artificial neural networks. International Journal of Refrigeration. 29, 243-249.

Osman, E.A. and Al-Marhoun, M.A. 2002. Using artificial neural networks to develop new PVT correlations for Saudi crude oils. 10th International Petroleum exhibition and conference, Abu Dhabi, U.A.E.

Petersen, P., Fredenslund, A. and Rasmussen, P. 1994. Artificial neural networks as a predictive tool for vapour-liquid equilibrium. Computers & Chemical Engineering. 18, Suppl. S63-S67.

Sharma, R., Singhal, D., Ghosh, R. and Dwivedi, A. 1995. Potential applications of artificial neural networks to thermodynamics: vapour-liquid equilibrium predictions. Computers & Chemical Engineering. 23, 385-390.

Sozen, A., Arcakilioglu, E. and Ozalp, M. 2004. Investigation of thermodynamic properties of refrigerant /absorbent couples using artificial neural networks. Chemical Engineering and Processing: Process Intensification. 43, 1253-1259.

Urata, S., Takada, A., Murata, J., Hiaki, T. and Sekiya, A. 2002. Prediction of vapour-liquid equilibrium for binary systems containing HFEs by using artificial neural network. Fluid Phase Equilibria. 199, 63-78.

Zahedi, G., Fazlali, A.R., Hosseini, S.M., Pazuki, G.R. and Sheikhattar, L. 2009. Prediction of asphaltene precipitation in crude oil. Journal of Petroleum Science and Engineering. 68, 218–222.