Iraqi Journal of Chemical and Petroleum Engineering

University of Baghdad
College of Engineering

Prediction of Fractional Hold-Up in RDC Column Using Artificial Neural Network

Adel Al-Hemiri and Suhayla Akkar

Chemical Engineering Department - College of Engineering - University of Baghdad - Iraq

Abstract

In the literature, several correlations have been proposed for hold-up prediction in rotating disk contactor. However, these correlations fail to predict hold-up over wide range of conditions. Based on a databank of around 611 measurements collected from the open literature, a correlation for hold up was derived using Artificial Neiral Network (ANN) modeling. The dispersed phase hold up was found to be a function of six parameters: N, v_c, v_d, Δr, m_c / m_d, s. Statistical analysis showed that the proposed correlation has an Average Absolute Relative Error (AARE) of 6.52% and Standard Deviation (SD) 9.21%. A comparison with selected correlations in the literature showed that the developed ANN correlation noticeably improved prediction of dispersed phase hold up. The developed correlation also shows better prediction over a wide range of operation parameters in RDC columns.

Keywords: dispersed phase hold up, RDC, artificial neural networks (ANN).

Introduction

In the design and scale up of RDC, it is necessary to explore the hydrodynamic behavior, mass transfer mechanism, and hold up effect within the equipment under different operating conditions. Dispersed phase hold up represents the total drop population in RDC column is defined as the ratio of dispersed phase to the volume of the column. The effect of the hold up on the performance of an extraction column is the most important hydrodynamic characteristic, because hold up is related to the interfacial area between the phases by:

$$a = \frac{6x}{d_{32}} \qquad (1)$$

Where x is the dispersed phase hold up and d_{32} is the sauter mean diameter. And the hold up is related to the rate of mass transfer (W) via (a) by:

$$W = K.V.a.\Delta c \qquad (2)$$

Wher K is the mass transfer coefficient, V is volume of the column and Δc is the concentration driving force.

In solvent extraction the re lationship between mass transfer and hydrodynamic performance is complex and there are many types of contactors each requiring a special understanding. Numerous experimental studies of dispersed phase hold up, drop size, mass transfer and mixing behavior within contactors have been reported [1].

In order to determine the interfacial area of the dispersion for the mass transfer calculation using equation (2) either of the following should be known:
1. The drop residence time in the contactor.
2. The fraction of the column occupied by the dispersed phase hold up.

In agitated contactors the residence time distribution is rather complex and dispersed phase hold up is therefore usually used for the estimation of interfacial area.

Virmijs and Karmers [2] investigated performance of RDC for various values of the rotor speed, total through put and solvent to feed ratio by comparing the separating efficiency with the fractional volume of the dispersed

phase under the same circumstances it was found that under certain condition the efficiency decreases although the hold up of the dispersed phase increases. This effect is ascribed to back mixing in continuous phase due to entrainment by the dispersed phase.

The hold up increased by increasing the solvent to feed ratio wh ile the total through put is kept constant, and the special kind of back mixing in the continuous phase impairs the efficiency of the extraction operation. Logsdail et.al [3] were the first to introduce the concept of dispersed phase hold up for the characterization of column design these authors modified the concept of relating the slip velocity v_s of the dispersed phase to the hold up in a two phase system by:

$$v_s = \frac{v_d}{x} + \frac{v_c}{1-x} \tag{3}$$

$$\bar{v}_o(1-x) = \frac{v_d}{x} + \frac{v_c}{1-x} \tag{4}$$

v_o is called the characteristic velocity and is defined as the mean velocity of the droplets extrapolated to essentially zero flow rates at a fixed rotor speed. Many correlations have been published relating the dispersed phase hold- up to the characteristic velocity in the form of equation 4 with additional factors for column size constriction and droplets coalescence and break up which could not be easily applied due to the amount of information required specially for v_o. Some selected reliable correlations are given in table (1). However these correlations fail to predict hold up over a wide range of conditions. Thus this work was initiated in order to develop a general correlation using artificial neural network.

Artificial Neural Network (ANN)

From an engineering view point ANN can be viewed as non linear empirical models that are especially useful in

Table (1) Correlations of Hold up in RDC

No.	Author	Correlation
1	Kasatkin et al (1962)	$x = 1.58\left(\frac{ND_r}{v_c}\right)\left(\frac{v_d}{v_c}\right)^{0.96}\left(\frac{D_x^2 - D_r^2}{D_t^2}\right)^{-0.7}$ $\left(\frac{\Delta\rho}{\rho_c}\right)^{-1.31}\left(\frac{z_c}{z_t}\right)^{-0.426}\left(\frac{\rho_c v_c D_t}{\mu_c}\right)^{-0.13}$ $\left(\frac{\rho_c D_t v_c^2}{\sigma}\right)^{0.245}\left(\frac{v_c^2}{gD_t}\right)^{0.96}$
2	Murakami et al (1978)	$x = 3.3\left(\frac{ND_c}{v_c}\right)^{0.55}\left(\frac{v_d}{v_c}\right)^{0.8}\left(\frac{D_x^2 - D_r^2}{D_t^2}\right)^{-0.5}$ $\left(\frac{z_c}{D_t}\right)^{-0.66}\left(\frac{D_r}{D_t}\right)^{0.4}\left(\frac{\Delta\rho}{\rho_c}\right)^{-0.13}\left(\frac{\rho_c D_t v_c^2}{\sigma}\right)^{0.18}$ $\left(\frac{v_c^2}{gD_t}\right)^{0.6}$
3	Kumar and Hartland (1987)	$x = [570.53 + 747.78(\frac{N^2 D_c}{g})^{1.28}]$ $[D_c^2 h^2 \rho_c g /(D_t^2 \sigma)]^{-0.45}(\Delta\rho/\rho_c)^{-0.58}$ $[\mu_c g^{1/4}/(\rho_c^{1/4}\sigma^{3/4})]^{0.85}[v_d^4 \rho_c /(g\sigma)]^{0.22}$ $(1 + v_c / v_d)^{0.35}$
4	Kalaichelvi, (1998)	$x = C_2(\psi_2)^{n_1}[G_f \psi_1]$ $\psi_2 = (Fr)^{-0.33}(Mo)^{-0.07}\left(\frac{\Delta\rho}{\rho_c}\right)^{-0.2}$ Where :- $\psi_1 = [(v_d/(\sigma\Delta\rho g/\rho_c^2)^{0.25}][1 + v_c/v_d]^{0.15}$ $Fr = (g/N^2.Dr), Mo = (\sigma^3.\Delta\rho)/(\mu_c^4.g)$ $G_f = (D_c/D_r)^{-2.5}(D_r/D_t)^{-2.5}(z_c/D_r)^{-0.75}$ No transfer $\quad\psi_2\langle 0.225 \quad C_2 = 0.79 \quad n_1 = 0.14$ $\quad\quad\quad\quad\quad\quad\psi_2\rangle 0.225 \quad C_2 = 52 \quad\quad n_1 = 3.2$ $c \rightarrow d \quad\quad\quad\psi_2\langle 0.17 \quad C_2 = 0.45 \quad n_1 = 0.05$ $\quad\quad\quad\quad\quad\quad\psi_2\rangle 0.17 \quad C_2 = 5.4 \quad n_1 = 1.5$ $d \rightarrow c \quad\quad\quad\psi_2\langle 0.2 \quad C_2 = 0.57 \quad n_1 = 0.04$ $\quad\quad\quad\quad\quad\quad\psi_2\rangle 0.2 \quad\quad C_2 = 8.1 \quad\quad n_1 = 1.7$

representing input-output data. Making predication, classifying data, reorganization patterns, and control process. ANN which will be referred to as a node in this work and is analogous to a single neuron in the human brain. The advantages of using artificial neural network in contrast with first principles models or other empirical models are [4-6],

1. ANN can be highly non linear.
2. The structure can be more complex and hence more representative than most other empirical models.
3. The structure does not have to be prespecified.
4. Quite flexible models.

(ANN) have been increasingly applied to many problems in transport planning and engineering, and the feed forward network with the error back propagation learning rule, usually called simply Back propagation (Bp), has been the most popular neural network [7].

Back-propagation

Back propagation was one of the first general techniques developed to train multi-layer networks, which does not have many of the inherent limitations of the earlier, single -layer neural nets. A back propagation net is a multilayer, feed forward network that is trained by back propagating the errors using the generalized Delta rule [8].

The steps for back- propagation training can be shown as follows [9]:
1. Initialize the weights with small, random values.
2. Each input unit broadcasts its value to all of the hidden units.
3. Each hidden unit sums its input signals and applies its activation function to compute its output signal.
4. Each hidden unit sends its signal to the output units.
5. Each output unit sums its input signals and applies its activation function to compute its output signal.
6. Each-output unit updates its weights and bias:

The conventional algorithm used for training a MLFF is the Bp algorithm, which is an iterative gradient algorithm designed to minimize the mean-squared error between the desired output and the actual output for a particular input to the network [10]. Basically, Bp learning consists of two passes through the different layers of the network: a forward pass and backward pass. During the forward pass the synaptic weights of the network are all fixed. During the backward pass, on the other hand, the synaptic weights are all adjusted in accordance with an error-correction rule [11].

The algorithm of the error back-propagation training is as given below [10]:
Step 1: initialize network weight values.
Step 2: sum weighted input and apply activation function to compute output of hidden layer

$$h_j = f\left[\sum_i x_i W_{ij}\right] \tag{4}$$

Where, h_j: The actual output of hidden neuron j for input signals X.
X_i: Input signal of input neuron (i).
W_{ij}: Synaptic weights between input neuron hidden neuron j and i.
f : The activation function.

Step3: sum weighted output of hidden layer and apply activation function to compute output of output layer.

$$O_k = f\left[\sum_j h_j W_{jk}\right] \tag{5}$$

Where O_k: The actual output of output neuron k.
W_{jk}: Synaptic weight between hidden neuron j and output neuron k.

Step 4: Compute back propagation error

$$d_k = (d_k - O_k)f'\left(\sum_j h_j W_{jk}\right) \tag{6}$$

Where f': The derivative of the activation function.
d_k: The desired of output neuron k.

Step 5: Calculate weight correlation term

$$\Delta W_{jk}(n) = h d_k h_j + a\Delta W_{jk}(n-1) \tag{7}$$

Step 6: Sums delta input for each hidden unit and calculate error term.

$$d_j = \sum d_k W_{jk} f'(X_i W_{ij}) \tag{8}$$

Step 7: Calculate weight correction term

$$\Delta W_{ij}(n) = h d_j X_i + a\Delta W_{ij}(n-1) \tag{9}$$

Step 8: Update weights

$$W_{jk}(n+1) = W_{jk}(n) + \Delta W_{jk}(n) \tag{10}$$

Step 9: Repeat step 2 for a given number of error

$$MSE = \frac{1}{2p}\left[\sum_p \sum_k (d_k^p - O_k^p)^2\right]$$

Where p is the number of patterns in the training set.

Step 10: End

Bp is easy to implement, and has been shown to produce relatively good results in many applications. It is capable of approximating arbitrary non-linear mappings.

However, it is noted that two serious disadvantages in the Bp algorithm are the slow rate of convergence, requiring very long training times, and getting stuck in local minima. The success of Bp methods very much depends on problem specific parameter settings and on the topology of the network [9].

The Activation Function used with the Back-Propagation

There are three transfer functions most commonly used for back propagation, but other differentiable transfer functions can be created and used with back propagation if desired. These functions are tansig, logsig, and purelin. The function logsig generates outputs between 0 and 1 as the neuron's net input goes from negative to positive infinity. Alternatively, multilayer networks may use the tan sigmoid transfer function. Occasionally, the linear transfer function purelin is used in back propagation networks. [8].

If the last layer of a multilayer network has sigmoid neurons, then the outputs of the network are limited to a small range. If linear output neurons are used the network outputs can take any value. In the present simulation the tansig is used.

Modeling Correlation of ANN

The modeling of ANN correlation began with the collection of large data bank followed by the learning file which was made by randomly selecting about 70% of the data base to train the network. The remaining 30% of data is then used to check the generalization capability of the model. The last step is to perform a neural correlation and to validate it statistically. So that the steps of modeling are:-

Collection of Data

The first step is collection of data.. Many investigators studied the hydrodynamics of RDC based on the dispersed phase hold up. In this model about 611 experimental points have been collected for mass transfer from continuous to dispersed phase (c—>d), for mass transfer from dispersed to continuous (d—>c) and for the case of no mass transfer in RDC. The data were divided into training and test sets: the neural network was trained on 70% of the data and tested on 30%. The data includes nine chemical systems with a large range of rotary speed, velocity of both continuous and dispersed phase as well as the physical properties for each chemical system. All of these parameters are input to neural network and there is one output; it is the hold up of dispersed phase.

The Structure of Artificial Neural Network

In this work, a multilayer neural network has been used, as it is effective in finding complex non-linear relationships. It has been reported that multilayer ANN models with only one hidden layer are universal approximates. Hence, a three layer feed forward neural network is chosen as a correlation model. The weighting coefficients of the neural network are calculated using MATLAB programming. Structure of artificial neural network built as:-

1. Input layer: A layer of neurons that receive information from external sources and pass this information to the network for processing. These may be either sensory inputs or signals from other systems outside the one being modeled. In this work six input neurons in the layer and there is a set of (427) data points available for the training set.

2. Hidden layer: A layer of neurons that receives information from the input layer and processes them in a hidden way. It has no direct connections to the outside world (inputs or output). All connections from the hidden layer are to other layers within the system. The number of neuron in the hidden layer is twenty one neurons. This gave best results and was found by trial and error. If the number of neurons in the hidden layer is more, the network becomes complicated. Results probably indicate that, the present problem is not too complex to have a complicated network routing. Hence, the results can be satisfactorily achieved by keeping the number of neurons in hidden layer at a best value of twenty one neurons.

3. Output layer: A layer of one neuron that receives processed information and sends output signals out of the system. Here the output is the hold up of dispersed phase in RDC.

4. Bias: The function of the bias is to provide a threshold for activation of neurons. The bias input is connected to each of hidden neurons in network.

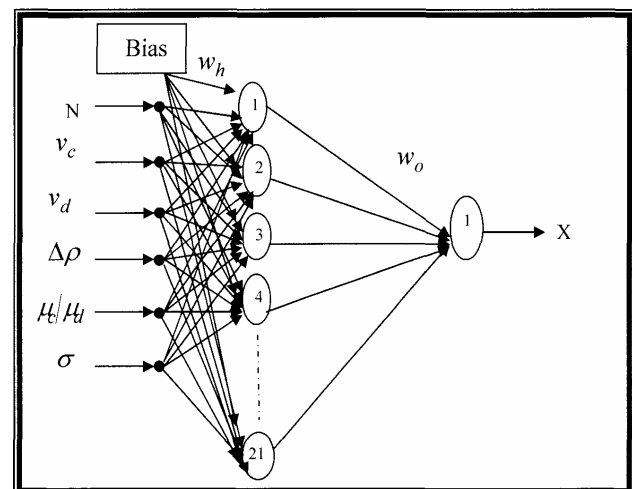The structure of muiti layer ANN modeling is illustrated in figure (1).

Figure (1) Structure of a layer neural network

Training of Artificial Neural Network

The training phase starts with randomly chosen initial weight values. Then a back-propagation algorithm is applied after each iteration, the weights are modified so that the cumulative error decreases. In back-propagation, the weight changes are proportional to the negative gradient of error. More details about this learning algorithm is shown in figure (1). Back-propagation may have an excellent performance. This algorithm is used to calculate the values of the weights and the following procedure is then used (called "supervised learning") to determine the values of weights of the network:-

1. For a given ANN architecture, the value of the weights in the network is initialized as small random numbers.
2. The input of the training set is sent to the network and resulting outputs are calculated.
3. The measure of the error between the outputs of the network and the known correct (target) values is calculated.
4. The gradients of the objective function with respect to each of the individual weights are calculated.
5. The weights are changed according to the optimization search direction.
6. The procedure returns to step 2.
7. The iteration terminates when the value of the objective function calculated using the data in the test approaches experimental value.

The trial and error to find the best ANN correlation model is shown in table 2.

Table (2) Network parameters in ANN model

Structure	Network Parameters				
	MSE	No. of iteration	Learning rate	Momentum coefficient	Transfer function
[6-16-1]	0.1	2590	0.7	0.9	Tan sigmoid
[6-18-1]	0.01	4321	0.65	0.9	Tan sigmoid
[6-21-1]	0.0001	9103	0.75	0.9	Tan sigmoid

With reduced MSE (Mean Square Error) the network is more accurate, because MSE is defined as:

$$MSE = \frac{1}{2p}\left[\sum_p \sum_k \left(d_k^p - O_k^p\right)^2\right] \qquad (13)$$

Where p is the number of patterns in training set, k is the number of iterations, d_k^p is the desired output, O_k^p is the actual output.

The learning process includes the procedure when the data from the input neurons is propagated through the network via the interconnections. Each neuron in a layer is connected to every neuron in adjacent layers. A scalar weight is associated with each interconnection.

Neurons in the hidden layers receive weighted inputs from each of the neurons in the previous layer and they sum the weighted inputs to the neuron and then pass the resulting summation through a non-linear activation function (tan sigmoid function).

Artificial neural networks learn patterns can be equated to determining the proper values of the connection strengths (i.e. the weight matrices w_h and w_o illustrated in figure 1) that allow all the nodes to achieve the correct state of activation for a given pattern of inputs. The matrix, bias, and vector, given equations (14), (15), and (16) illustrate the result of coefficient weights for ANN correlation , where wl is the matrix containing the weight vectors for the nodes in the hidden layer, W_o is the vector containing the weight for the nodes in the output layer and is the bias.

$$w_h = \begin{pmatrix} 56.7246 & 10.4097 & 200.6449 & -6.8164 & -114.1237 & -136.2041 \\ 1.9171 & 76.9097 & -9.2793 & 0.0571 & 30.6955 & -40.6522 \\ -1.7947 & 0.9306 & 4.0325 & -0.0589 & 10.1928 & 26.5990 \\ -0.2412 & 17.9286 & -5.7796 & 0.0345 & 1.5511 & -17.5543 \\ -0.0163 & -0.8190 & 18.0777 & 0.0100 & -0.2493 & -30.5079 \\ 0.3733 & -0.0747 & -5.2589 & 0.0713 & 0.4694 & -31.8241 \\ -3.1290 & -7.8760 & 0.3263 & 0.5763 & 2.8053 & 13.4445 \\ -0.2070 & 7.0679 & -34.0529 & -0.0474 & -0.7174 & 15.6315 \\ -0.2649 & 1.4468 & -4.1561 & 0.0634 & 2.5667 & 17.9431 \\ 0.0386 & -10.9673 & 12.3793 & -0.0063 & 1.4831 & 64.3704 \\ 0.1677 & -2.4429 & -7.3706 & -0.0557 & 0.2196 & -41.2561 \\ -0.1999 & 34.8542 & -11.3994 & 0.0816 & -1.0828 & -19.7664 \\ 0.2910 & 1.8847 & 5.8097 & -0.0018 & 0.3520 & 38.8415 \\ 0.2341 & -43.5690 & 17.9737 & -0.0645 & -1.2331 & 46.0181 \\ 0.0665 & -6.5307 & -10.7709 & -0.0630 & 1.6957 & -41.6636 \\ -0.3306 & 41.8971 & -10.5881 & 0.1190 & -1.7422 & -15.1618 \\ -2.5697 & 47.8813 & -34.2013 & 0.4440 & 4.7455 & 78.7878 \\ -0.0267 & -3.3442 & 15.5450 & 0.0058 & -0.4950 & -41.1809 \\ -2.0764 & 32.1499 & -149.6137 & -0.0518 & 5.2072 & 17.0380 \\ -0.0368 & 10.8577 & -13.2069 & 0.0043 & -1.4947 & -62.9708 \\ 0.3421 & 12.0678 & -14.6781 & 0.0035 & 5.678 & -7.9851 \end{pmatrix} \qquad (14)$$

$$b = \begin{pmatrix} -56.2581 \\ -7.7745 \\ -5.9822 \\ 2.5460 \\ -0.6105 \\ -12.1450 \\ 2.9893 \\ 10.9867 \\ -11.7293 \\ -1.0443 \\ 7.4893 \\ -3.2022 \\ -3.9623 \\ 4.8630 \\ 6.9656 \\ -4.2633 \\ -15.2787 \\ 1.0178 \\ 2.1668 \\ 1.4045 \\ 1.3420 \end{pmatrix} \qquad (15)$$

$$W_0 = [\,2.7831 \;\; -70.8743 \;\; 6.2755 \;\; -91.8218 \;\; 14.9142 \;\; 1.4873 \;\; 15.321 \;\; 1.0126 \;\; -2.6559 \\ 75.3846 \;\; -5.3861 \;\; 60.5749 \;\; -2.0201 \;\; 3.7267 \;\; 5.6162 \;\; -63.0492 \;\; 6.9976 \;\; -12.8382 \\ -51.6626 \;\; 76.4074 \;\; 8.2310\,] \qquad (16)$$

Simulation Results

The network architecture used for predicting hold up is illustrated in figure (1) consist of six inputs neurons corresponding to the state variables of the system, with 21 hidden neurons and one output neuron. All neurons in each layer were fully connected to the neurons in an adjacent layer. The prediction of ANN correlation result is plotted in figure (2) compares the predicted hold up with experimental hold up for training set
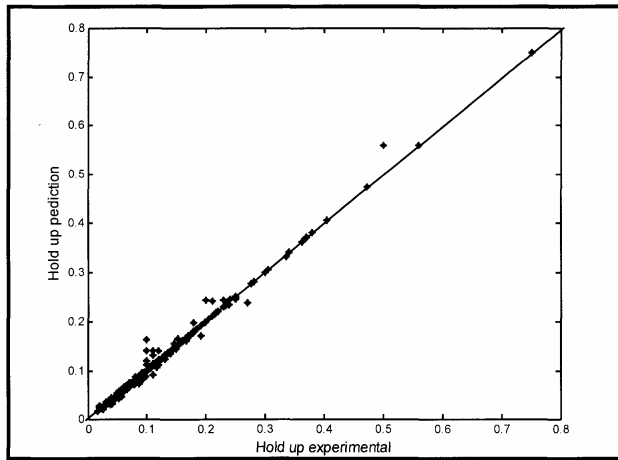

Figure (2) Comparison between experimental and predicted hold up in training set
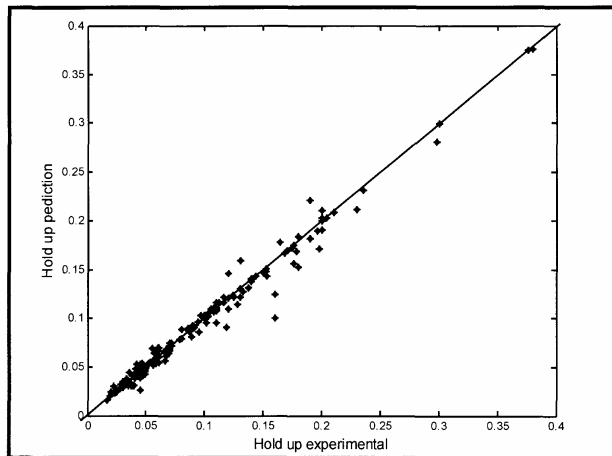

Figure (3) Comparison between experimental and predicted hold up in testing set

Test of the Proposed ANN

The purely empirical model was tested on data that were not used to train the neural network and yielded very accurate predictions. Having completed the successful training, another data set was employed to test the network prediction hold up. We made use of the same model to generate (184) new data values. The result of

prediction is plotted with experimental values as shown in figure (3).

Statistical Analysis

Statistical analysis based on the test data is calculated to validate the accuracy of the output for pervious correlation model based on ANN. The structure for each model should give the best output prediction, which is checked by using statistical analysis. The statistical analysis of prediction is based on the following criteria:

1. The AARE (Average Absolute Relative Error) should be minimum:

$$AARE = \frac{1}{N} \sum_{1}^{N} \left| \frac{x_{prediction} - x_{exp\,erimental}}{x_{exp\,erimental}} \right| \qquad (17)$$

Where N here is the number of data points. x is the hold up.

2. The standard deviation should be minimum.

$$SD = \sqrt{\sum \frac{\left[\left| (x_{prediction} - x_{experimental}) / x_{experimental} \right| - AARE \right]^2}{N-1}} \qquad (18)$$

3. The correlation coefficient R between input and output should be around unity.

$$R = \frac{\sum_{i=1}^{N} \left(x_{experimental(i)} - \overline{x}_{experimental} \right) \left(x_{prediction(i)} - \overline{x}_{prediction} \right)}{\sqrt{\sum_{i=1}^{N} \left(x_{experimental} - \overline{x}_{experimental} \right)^2} \sqrt{\sum_{i=1}^{N} \left(x_{prediction} - \overline{x}_{prediction} \right)^2}} \qquad (19)$$

Where $\overline{x}_{exp\,erimental}$ =hold up mean of experimental points, $\overline{x}_{prediction}$ =hold up mean for prediction points.

The literature correlations (in table 1) were used to estimate the hold up. These correlations show a poor agreement between the prediction and experimental hold up value compared with ANN correlation. Table (3) gives information of comparing these correlation with ANN prediction in testing set.

Table (3) Comparison of ANN and previous literature correlations in testing set

Correlation	AARE%	S.D%	R
Kastkin(1962)	51.93	32.55	0.695
Murakami(1978)	41.29	23.94	0.7914
Hartland(1987)	32.79	22.59	0.778
Kalaichelvi(1998)	32	27.63	0.726
ANN (this work)	6.52	9.21	0.998

Conclusions

The ANN correlation shows noticeable improvement in the prediction of dispersed phase hold up. The neural network correlation yield an AARE of 6.52% and standard deviation of 9.21%, which is better than those, obtained for the selected literature correlations. Also ANN correlation yielded improved predictions for variety of liquid systems and a wide range of operating parameters. The number of input units and output units are fixed to a problem (here, 6 and 1 respectively) but the choice of the number of the hidden units is flexible. In this work best results were obtained employing 21 hidden neurons.

Nomenclature

a	Interfacial mass transfer area	m^2/m^3
b	Bias	
Δc	Concentration driving force	kg/m^3
d_{32}	Sauter mean diameter	
D_r	Diameter of rotary disk	m
D_s	Stator ring opening	m
D_t	Diameter of RDC column	M
f	The activation function	
f'	The derivation of the activation function	
g	Gravitational constant	m/s^2
h_i	The actual output of hidden neuron j	
K	Mass transfer coefficient`	m/s
n	Number of input neurons	
N	Speed of rotor dist	rps
O_k	The actual output of neuron k	
P	The number of patterns in the training set	
R	Correlation coefficient	
V	Volume of column	m^3
v_c	Velocity of continuous phase	m/s
v_d	Velocity of dispersed phase	m/s
v_o	Characteristic velocity	m/s
v_s	Slip velocity	m/s
W	Rate of mass transfer	kg/s
W_{ij}	Synaptic weights between input and hidden neuron	
W_{jk}	Synaptic weights between input and output neuron	
x	Hold up	
X_i	Input vector	
$\overline{x}$	Mean hold up	
z_c	Height of compartment	m
z_t	Height of RDC column	M

Greek symbols

a	Momentum to accelerate the network convergence process	
d_k	The error term	
h	The learning rate	
m	Viscosity	kg/m.s
s	Interfacial tension	N/m
r	Density	kg/m^3
Δr	Density difference	kg/m^3

Subscripts

c	Continuous phase
d	Dispersed phase

References

1. Bailes, P.J., Gledhill, J., Godfrey, J.C and Slater, M.J, "Hydrodynamic behavior of packed, rotating disc contactor, and Kuhn", Chem. Eng. Res. Des, 64, 43-55, (1986).
2. Vermijs, H.J., and Kramers, H., "Liquid-liquid extraction in rotating disc contactor", Chem. Eng. Sci., 3, 55-64, (1954).
3. • Logsdail, D.H., Thomton, J.D., and Pratt, H.R.C., "liquid-liquid extraction part XII: flooding rates and performance data for a rotating disc contactor", Trans. Inst. Chem. Eng., 35, 301-315, (1957).
4. David M.H., "Applications of artificial neural networks in chemical engineering", Korean. J. Chem. Eng., 17, 373-392, (2002).
5. Patterson, "Artificial neural networks theory and applications". Prentice Hall, (1996).
6. Sivanadam, S.N., "Introduction to artificial neural networks", Vikas publishing House Pvt. Ldt, (2003).
7. Freeman, J.A., and Skapura, D.M., "Neural networks", Jordan University of Science and Technology, July (1992).
8. MATLAB, Version 7, June 2003, "Neural network toolbox"
9. Leonard, J., and Kramer, M.A., "Improvement of the back-propagation algorithm for training neural networks", Comp. Chem. Eng, 14, 337-341, (1990).
10. Lendaris, G., "Supervised learning in ANN from introduction to artificial intelligence". New York, April 7, (2004).
11. Lippmann, R.P., "An introduction to computing with neural nets", IEEE Magazine, April, pp.4-22,(1987).
12. Kalaichelvi, P., Murugesan, T., "Dispersed phase hold up in rotating disc co ntactor", Bioprocess Engineering, 18,105-111, (1998).
13. Kasatkin, A.G., and Kagan, S.Z Appl., Chem., USSR, 35, 1903, (1962) [Cited in Murakami, A., and Misonou, A., and Inoue, A., 1978].
14. Kumar, A., and Hartland, S., "Independent prediction of slip velocity and hold up in liquid-liquid extraction columns", Can. J. Chem. Eng., 67, 17, (1987).
15. Murakam, A., Misonou, A., Inoue, K., " Dispersed phase hold up in a rotating disc extraction column", International Chemical Engineering, 18,1, (1978).