ELSEVIER

Chemical Engineering Journal 141 (2008) 27–34

Chemical Engineering Journal

www.elsevier.com/locate/cej

Artificial neural network simulation of combined humic substance coagulation and membrane filtration

Mohammed Al-Abri, Nidal Hilal *

Centre for Clean Water Technologies, School of Chemical and Environmental Engineering, University of Nottingham, Nottingham NG7 2RD, UK

Received 3 August 2007; received in revised form 27 September 2007; accepted 4 October 2007

Abstract

Backpropagation artificial neural network (BPNN) was utilized to predict membrane performance. The network was used to predict and compare humic substance (HS) retention and membrane fouling with previously obtained experimental data. BPNN simulation results show high network reliability, if the network is implemented correctly. The difference between the predicted and experimental data was lower than 5%. Low number of training data input has been shown to hinder the learning process. A high number of training data input has lead to over-fitting or memorization of the training data set, reducing the networks predictability. The number of neurons in the hidden layers needs to be chosen carefully to obtain a reliable network. This paper shows that a lower number of neurons result in low reliability, while a higher number of neurons leads to data over-fitting. The best performance was obtained with 2-10 neurons for HS and heavy metals agglomeration and 5-15 neurons for HS coagulation with and without heavy metals.

Keywords: Membrane separation; Artificial neural network; Prediction

1. Introduction

Mathematical models derived from physical descriptions and understanding of the membrane process to characterize the membranes to predict membrane performance and fouling are conventionally undertaken using mathematical models [1]. These models are mathematically complex, computationally expensive and they ideally require a very detailed knowledge of the filtration process. Efficient alternatives are required to predict the process performance by simulating available data and extending it to unavailable data [2,3]. Artificial neural networks (ANNs) operate like black box models. They offer an attractive alternative to conventional black box models in dealing with complex processes. They are capable of modeling highly complex and non-linear systems with many interrelated parameters. They do not require detailed information about the physical parameters of the system. Instead, they use available data to predict the relationship between input and output parameters [2,4]. ANN with enough number of neurons can theoretically approximate any function to any level of accuracy [5,6]. Recent breakthrough in the computing industry promoted the use of ANN models in desalination and membrane processes [7–11].

An ANN is composed of a number of highly interconnected processing elements (neurons) working in parallel to solve a specific problem. ANN cannot be programmed to perform a specific task. It requires selective examples to learn from otherwise the network will be unreliable and time consuming. The ways the neurons interact with each other determine the operation of the ANN (connection formula). There are several types of ANNs such as feed-forward networks (perceptron network) and feedback networks (recurrent network) [12,13]. Feedback networks can have neurons signals traveling in both directions by introducing loops in the network enabling them to route back to previous neurons. Recurrent networks are typically used for classification problems with binary pattern vectors (pattern recognition) [14,15]. Feed-forward ANN allows signals to travel in one direction: from input to output. Feed-forward ANNs tend to be straightforward networks that associate inputs with outputs if there is no feedback (loops). They are extensively used for function approximation [16].

The feed-forward network is commonly used with an error correction algorithm such as backpropagation. The standard backpropagation neural network (BPNN) algorithm relies on a search technique (e.g. gradient descent), in which the network

* Corresponding author. Fax: +44 115 9514115.
E-mail address: nidal.hilal@nottingham.ac.uk (N. Hilal).

1385-8947/$ – see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.cej.2007.10.005

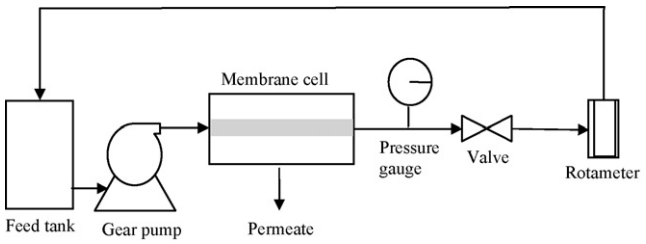

Fig. 1. Cross flow filtration model.

weights are changed along the negative of the gradient of the performance function [17]. The learning process of BPNN is based on iterative weight adjustments. Information from inputs is fed forward through the network to optimize the weights between neurons. The BPNN compares output and target values and modify the weight values according to a specific learning algorithm to reduce the overall error. The modified weights are then propagated backwards into the system. This forward–backward process is carried out for each epoch (set of training patterns used to compute the global error) and is repeated until the difference between predicted output and target value reach an accepted range [12,18].

This study investigates the use of BPNN to predict membrane retention and fouling. A BPNN model is designed and different architectures are compared to determine the best architecture to use in data prediction. When the architecture was chosen, a comparison between previously obtained experimental data and BPNN data was undertaken. Previous experimental data includes agglomeration of humic substances (HS) and heavy metals, HS coagulation and HS and heavy metals coagulation.

2. Experimental methodology

Membrane filtration experiments were carried out in a laboratory scale filtration cell. Schematic diagram of the experimental set-up is shown in Fig. 1. Complete description of the materials used and experimental procedure are presented in previous articles [19,20]. Complete description of the materials and operating conditions used during the experimental part of this study are presented in a previous article [21].

HS retention was measured using [22]:

$$R = 1 - \frac{C_p}{C_b} \tag{1}$$

where R is retention, C_p is permeate concentration (mg/l) and C_b is bulk concentration (mg/l). C_p and C_b were measured using Total Organic Carbon (TOC) Analyzer. Membrane fouling is calculated using [21]:

$$\text{Fouling} = \left(1 - \frac{J_v}{J_0}\right) \times 100 \tag{2}$$

where 'Fouling' is membrane fouling during the period of the filtration (%), J_0 is pure water flux (l/m^2 h) and J_v is solution permeate flux (l/m^2 h).

3. Modeling

3.1. Artificial neural network

A successful BPNN requires internal parameters determination such as network architecture and initial weights to meet the required performance [23,24]. A poorly designed network will result in unreliable results. Finding a suitable architecture and the corresponding weights of the network is a complex task due to the lack of theoretical parameters or optimal values. This results in the need for trial and error approach using different initializations and architectures [25,26]. The architecture of a typical BPNN is presented in Fig. 2. Matlab 7.3 was used to construct and simulate the BPNN. The BPNN is composed of a set of elements of calculation (layers) connected to each other. Fig. 2 shows a 3-layer BPNN with n, m and p as the number of input, hidden and output layers respectively. Neurons in the hidden and the output layers calculate their inputs by performing a weighted sum of the outputs they received from the previous layer. Their outputs however are calculated by transforming their inputs using a transfer function. The most widely used transfer functions are the log-sigmoid (logsig) transfer function (Eq. (3); Fig. 3A), the tan-sigmoid (tansig) transfer function (Eq. (4); Fig. 3B) and the linear (purelin) transfer function (Fig. 3C). Logsig function produces outputs in the range of 0 to 1, tansig function produces outputs in the range of −1 to +1 and purelin function produces outputs in the range of $-\infty$ to $+\infty$ [27,28].

$$f(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{4}$$

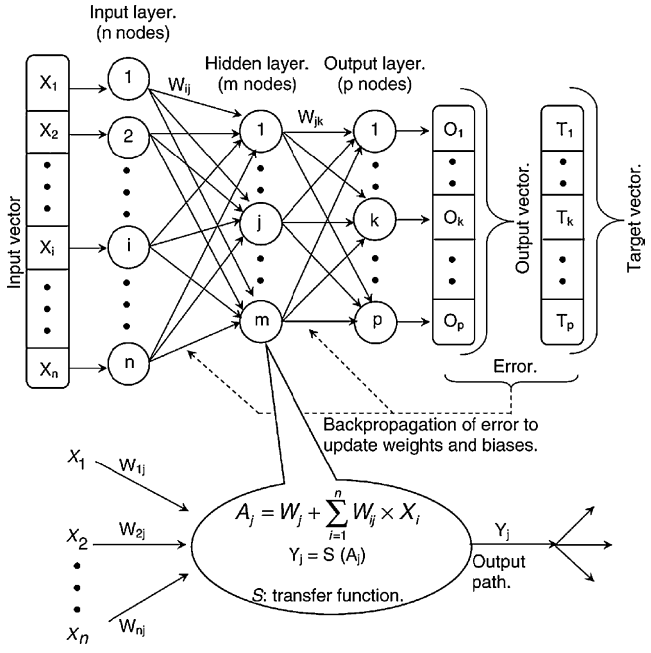

Fig. 2. Architecture of a typical backpropagation artificial neural network BPNN [17].

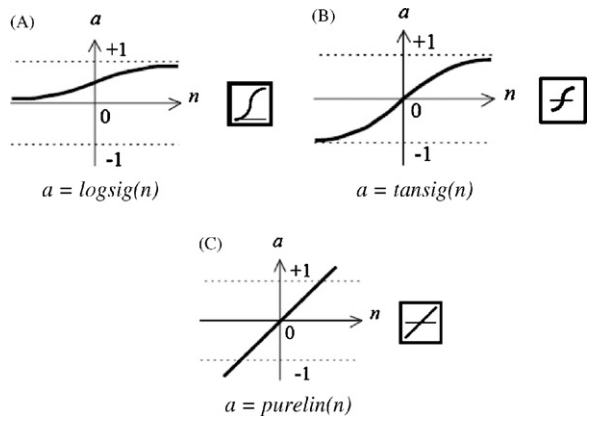

Fig. 3. Typical transfer functions used in BPNN: (A) logsig, (B) tansig and (C) purelin.

The BPNN used in this study is based on the following equation:

$$O_k = S\left(\sum_{j=1}^{m} W_{jk} x S\left(\sum_{i=1}^{n} W_{ij} X_i\right)\right) \quad (5)$$

where O_k are the output values, X_i are the input values of the network, W_{ij} are the connection weights between the input layer and the hidden layer, W_{jk} are the connection weights between the hidden layer and the output layer and S is transfer function.

The input each node in the previous layer (X_i) is multiplied by an adjustable connection weight (W_{ij}). At each node, the weighted input signals are summed and a bias value (W_j) is added. The bias is an extra input added to the neurons, which has a constant value of 1 and treated like other connection weights. This combined input (A_j) is then passed through the transfer function (S) to produce the output node (O_j) as illustrated in Fig. 2. The output of one node contributes to the input to the nodes in the next layer.

There are usually four steps involved in ANN modeling: (1) Assembly of dataset, defining the input and output data, (2) deciding the network architecture, (3) training (network learning) and (4) simulating the network response to new inputs.

3.2. Principle component analysis

Conventional BPNN suffers from slow convergence to local and global minima and from random settings of initial values of weights, which reduces the prediction accuracy of the network. Fuzzy hybrid neural networks such as principle component analysis (PCA) were introduced to increase the efficiency of neural networks [29–32]. PCA is used when the dimension of the input vector is large and consists of redundant values. PCA reduces the dimension of the input vectors to the BPNN by producing uncorrelated data input and by eliminating the components that contribute the least to the variations of the data input. It orders the resulting dimensions so that the data, which contribute the most, come first. Therefore, the training period of the neural network is decreased. PCA is a well-established technique for feature extraction and dimensionality reduction [33].

A number of input vectors produced using PCA were introduced into the BPNN. The PCA-BPNN efficiency was assessed by comparing it with membrane retention and fouling experimental results.

4. Results and discussion

4.1. BPNN optimization

The quality of different network architectures is measured using average absolute error (AE), maximum error (ME) and correlation coefficient (R). Maximum error is the highest difference between predicted and target results. The network was run 20 times to average out any possible errors or unexpected values. The overall coefficient factor (R), the average absolute error (AE) and the maximum error (ME) for the simulation were 0.97, 0.02 and 0.07 respectively. The training datasets used in this study were randomly arranged to ensure the absence of data fitting, i.e. overriding of BPNN predictions. The five input vectors used in HS and heavy metals agglomeration BPNN training were membrane type, salinity level, HS concentration, heavy metals concentration and trans-membrane pressure. While the six input vectors for HS coagulation excluded heavy metals concentration and included polyelectrolyte type and concentration. Finally, HS and heavy metals coagulation BPNN simulation included heavy metals concentration as well as polyelectrolyte type and concentration providing a total of seven input vectors. Membrane retention was used as the output vector to test the network.

The results of training data set, number of neurons in hidden layers, training algorithm and PCA input vectors variations on BPNN prediction are shown in Tables 1–3. Results showed an optimum training data set range, where lower or higher than optimum data input points provided unreliable results as shown in Runs 1–3 in Table 1, Runs 1–5 in Tables 2 and 3. These results suggest that lower number of training data hinder the learning process and high number of training data can depress the generalizing abilities of the network through over-fitting or memorization of the training data set [12].

The performance of varying the number of neurons was assessed as shown in Runs 2, 4–6 in Table 1 and Runs 3, 6–8 in Tables 2 and 3. HS and heavy metals agglomeration simulation achieved high prediction accuracy at 2-10 neurons. Prediction accuracy decreased with further increase in number of neurons to 15. The reduction in prediction accuracy is due to the network memorizing the training set instead of trying to generalize and predict, i.e. data over-fitting [12]. HS coagulation and HS and heavy metals coagulation simulation results showed low prediction accuracy at 2 neurons. All other neurons variations (5-15) produced high prediction accuracy. Low number of neurons prevents the BPNN from appropriately learning and approximating the target values.

Runs 2, 7, 8 in Table 1 and Runs 3, 9, 10 in Tables 2 and 3 illustrate the effect of using different training algorithms on network performance. All studied training algorithms (trainlm, trainscg and trainbr) showed high prediction accuracy.

Runs 2, 9–11 in Table 1 and Runs 3, 11–13 in Tables 2 and 3 show the effect of input vectors produced using PCA on BPNN

Table 1
HS and heavy metals ANN optimization: initial 5 input vectors, 108 data input set points, 30 predicted set points, 0.0001 performance goal, 500 epochs and using tansig transfer function

Run	Architecture						
	PCA input vectors	Training set point	Training algorithm	No. of neurons	*R*	ME	AE
1	–	40	Trainlm	5-1	0.68	0.15	0.02
2	–	60	Trainlm	5-1	0.98	0.04	0.01
3	–	80	Trainlm	5-1	0.95	0.07	0.01
4	–	60	Trainlm	2-1	0.98	0.09	0.03
5	–	60	Trainlm	10-1	0.98	0.04	0.01
6	–	60	Trainlm	15-1	0.87	0.11	0.04
7	–	60	Trainscg	5-1	0.97	0.03	0.01
8	–	60	Trainbr	5-1	0.99	0.04	0.01
9	2	60	Trainlm	5-1	0.57	0.12	0.05
10	3	60	Trainlm	5-1	0.84	0.10	0.05

Table 2
HS coagulation ANN optimization: initial 6 input vectors, 428 data input set points, 160 predicted set points, 0.0001 performance goal, 500 epoch and using tansig transfer function

Run	Architecture						
	PCA input vectors	Training set point	Training algorithm	No. of neurons	*R*	ME	AE
1	–	140	Trainlm	5-1	0.79	0.10	0.04
2	–	160	Trainlm	5-1	0.94	0.04	0.01
3	–	180	Trainlm	5-1	0.97	0.04	0.01
4	–	200	Trainlm	5-1	0.98	0.03	0.01
5	–	220	Trainlm	5-1	0.86	0.09	0.02
6	–	180	Trainlm	2-1	0.66	0.24	0.14
7	–	180	Trainlm	10-1	0.96	0.04	0.01
8	–	180	Trainlm	15-1	0.96	0.04	0.01
9	–	180	Trainscg	5-1	0.97	0.04	0.01
10	–	180	Trainbr	5-1	0.97	0.04	0.01
11	3	180	Trainlm	5-1	0.69	0.12	0.05
12	4	180	Trainlm	5-1	0.79	0.09	0.03
13	5	180	Trainlm	5-1	0.94	0.04	0.02

prediction. BPNN prediction accuracy increased with increasing number of input vectors. Low numbers of input vectors were insufficient to produce reliable BPNN results; while high input vectors produced reliable results compared to the normal BPNN simulation. HS and heavy metals agglomeration required 4 PCA input vectors from the original 5 input vectors, HS coagulation required 5 PCA input vectors from the original 6 vectors, and HS and heavy metals coagulation required 6 PCA input vectors

Table 3
HS and heavy metals coagulation ANN optimization: initial 7 input vectors, 222 data input set points, 100 predicted set points, 0.0001 performance goal, 500 epoch and using tansig transfer function

Run	Architecture						
	PCA input vectors	Training set point	Training algorithm	No. of neurons	*R*	ME	AE
1	–	60	Trainlm	5-1	0.74	0.19	0.04
2	–	80	Trainlm	5-1	0.96	0.06	0.02
3	–	100	Trainlm	5-1	0.98	0.04	0.01
4	–	140	Trainlm	5-1	0.97	0.05	0.01
5	–	160	Trainlm	5-1	0.81	0.12	0.04
6	–	100	Trainlm	2-1	0.55	0.27	0.14
7	–	100	Trainlm	10-1	0.96	0.04	0.02
8	–	100	Trainlm	15-1	0.94	0.07	0.01
9	–	100	Trainscg	5-1	0.99	0.04	0.02
10	–	100	Trainbr	5-1	0.99	0.03	0.02
11	4	100	Trainlm	5-1	0.66	0.17	0.09
12	5	100	Trainlm	5-1	0.86	0.11	0.05
13	6	100	Trainlm	5-1	0.97	0.05	0.02

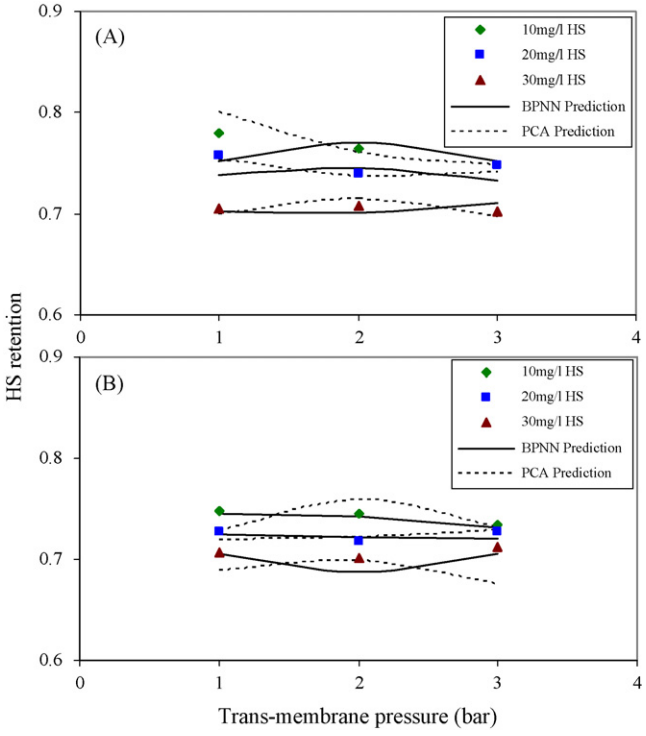

Fig. 4. BPNN and PCA-BPNN performance in predicting HS retention in HS and heavy metals agglomeration experiments using P005F membrane at initial concentration of 5 mg/l heavy metals: (A) 10,000 ppm NaCl and (B) 25,000 ppm NaCl.

from the original 7 vectors to produce comparable results to the normally operated BPNN. The results show that the dataset fed into PCA had little redundancy, which can be removed to increase the speed of network conversion.

4.2. HS retention BPNN simulation

On the basis of the considerations described in Section 4.1, a BPNN and a PCA-BPNN were chosen to compare their prediction with HS retention determined experimentally. Fig. 4 compares BPNN and PCA-BPNN predicted values to HS and heavy metals agglomeration experimental results using simulation Runs 2 and 11 in Table 1. The predicted values are within good agreement with the experimental values. HS retention was 0.75, 0.75 and 0.73 at experimental, BPNN and PCA-BPNN values, respectively, using P005F membrane at initial feed concentration of 25,000 ppm NaCl, 10 mg/l HS and 5 mg/l heavy metals.

Fig. 5 compares BPNN and PCA-BPNN predicted values (Runs 3 and 13 in Table 2) with HS coagulation experimental results. Similar to previous results, the predicted values are within good agreement with the experimental values. HS retention determined using BPNN, PCA-BPNN and experimental results using P005F membrane at initial feed concentration of 25,000 ppm NaCl, 10 mg/l HS and 2 mg/l PDADMAC concentration were 0.89, 0.91 and 0.91, respectively.

Fig. 6 compares BPNN and PCA-BPNN predicted values using simulation Runs 3 and 13 in Table 3 with HS and heavy metals coagulation experimental results. The predicted values

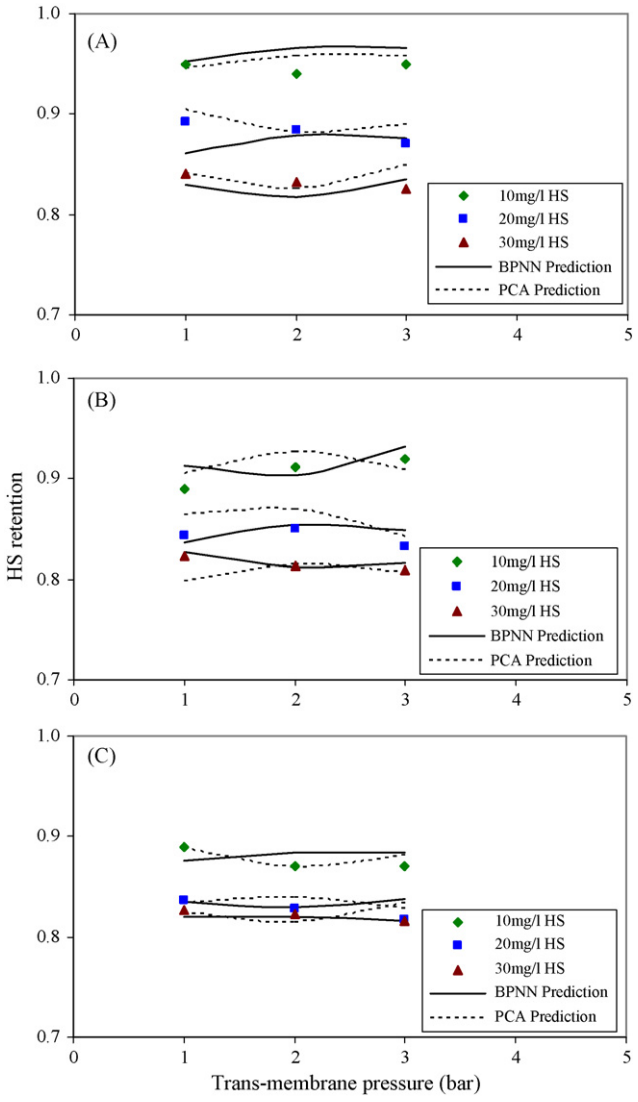

Fig. 5. BPNN and PCA-BPNN performance in predicting HS retention in HS coagulation experiments using P005F membrane at initial concentration of 2 mg/l PDADMAC: (A) 10,000 ppm NaCl, (B) 25,000 ppm NaCl and (C) 35,000 ppm NaCl.

are within good agreement with the experimental values. HS retention determined using BPNN, PCA-BPNN and experimental results using P005F membrane at initial feed concentration of 25,000 ppm NaCl, 10 mg/l HS, 5 mg/l heavy metals and 2 mg/l PDADMAC concentration were 0.91, 0.91 and 0.92, respectively.

4.3. Membrane fouling BPNN simulation

The simulation used in Section 4.2 is repeated to compare BPNN and PCA-BPNN prediction with membrane fouling determined experimentally. All conditions are kept unchanged from Section 4.2 except using purelin transfer function instead of tansig. Tansig produces outputs in the range of −1 to +1, while logsig produces outputs in the range of 0 to 1. In all previous simulations, using logsig or purelin instead of tansig did not improve the prediction accuracy of the network. Membrane

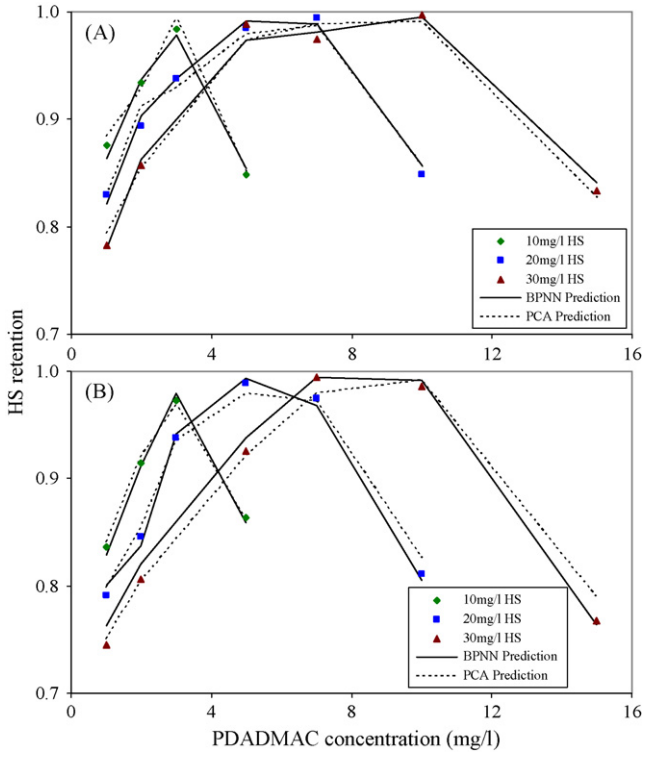

Fig. 6. BPNN and PCA-BPNN performance in predicting HS retention in HS and heavy metals coagulation experiments using P005F membrane operating at 3 bar and initial concentration of 5 mg/l heave metals: (A) 10,000 ppm NaCl and (B) 25,000 ppm NaCl.

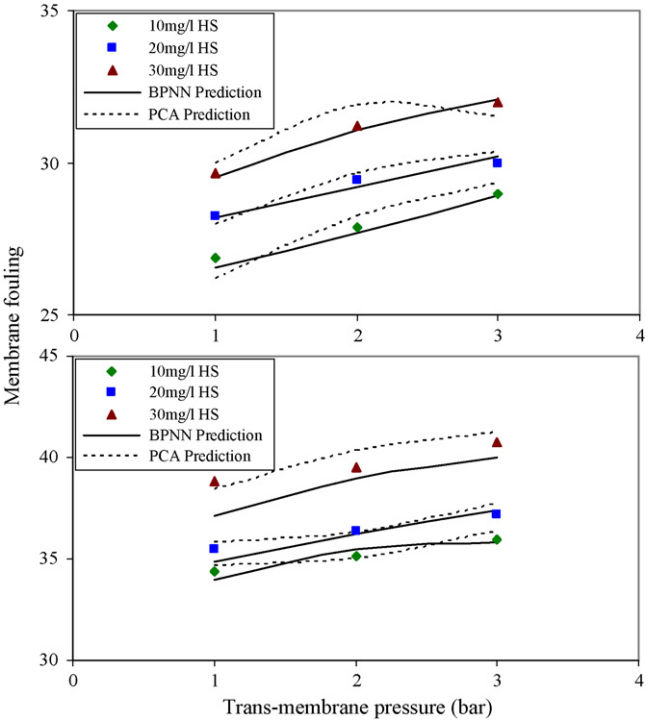

Fig. 7. BPNN and PCA-BPNN performance in predicting membrane fouling in HS and heavy metals agglomeration experiments using P005F membrane at initial concentration of 5 mg/l heavy metals: (A) 10,000 ppm NaCl and (B) 25,000 ppm NaCl.

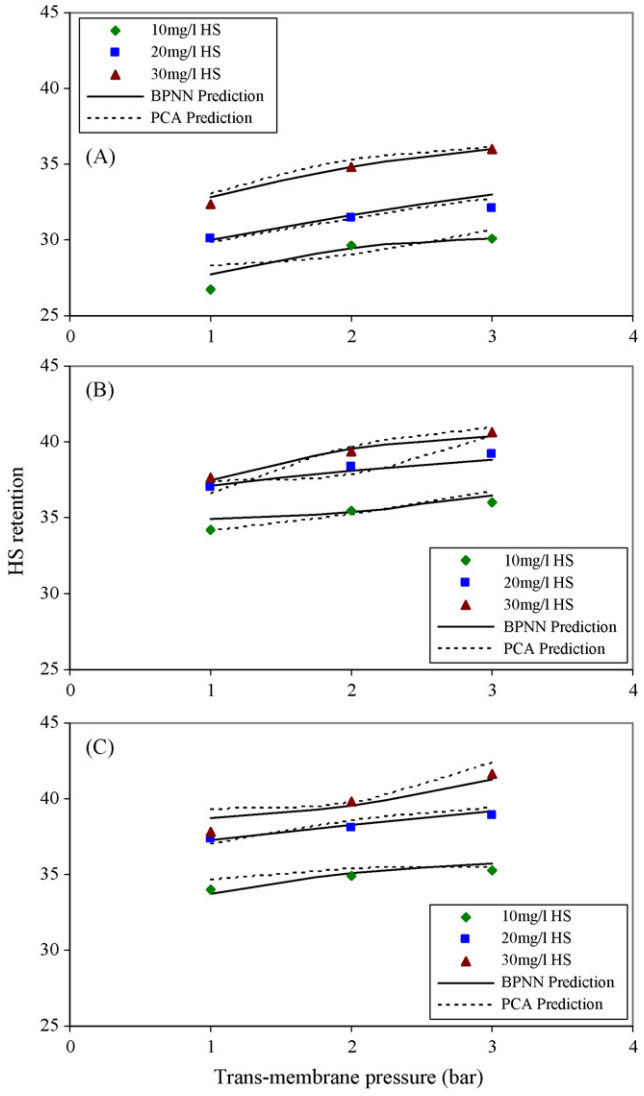

Fig. 8. BPNN and PCA-BPNN performance in predicting membrane fouling in HS coagulation experiements using P005F membrane at initial concentration of 2 mg/l PDADMAC: (A) 10,000 ppm NaCl, (B) 25,000 ppm NaCl and (C) 35,000 ppm NaCl.

fouling results have values higher than 1. Purelin is the only transfer function that can be used at this range. Purelin produces outputs in the range of $-\infty$ to $+\infty$.

Fig. 7 compares BPNN and PCA-BPNN predicted values with HS and heavy metals agglomeration experimental results. The predicted values are within good agreement with the experimental values. Membrane fouling was 34.4, 34.0 and 34.7% at experimental, BPNN and PCA-BPNN values, respectively, using P005F membrane at initial feed concentration of 25,000 ppm NaCl, 10 mg/l HS and 5 mg/l heavy metals.

Figs. 8 and 9 compare BPNN and PCA-BPNN predicted values to HS coagulation with and without heavy metals experimental results. Similar to previous results, the predicted values are within good agreement with the experimental values. Membrane fouling determined using BPNN, PCA-BPNN and experimental results using P005F membrane at initial feed concentration of 25,000 ppm NaCl, 10 mg/l HS, 5 mg/l heavy

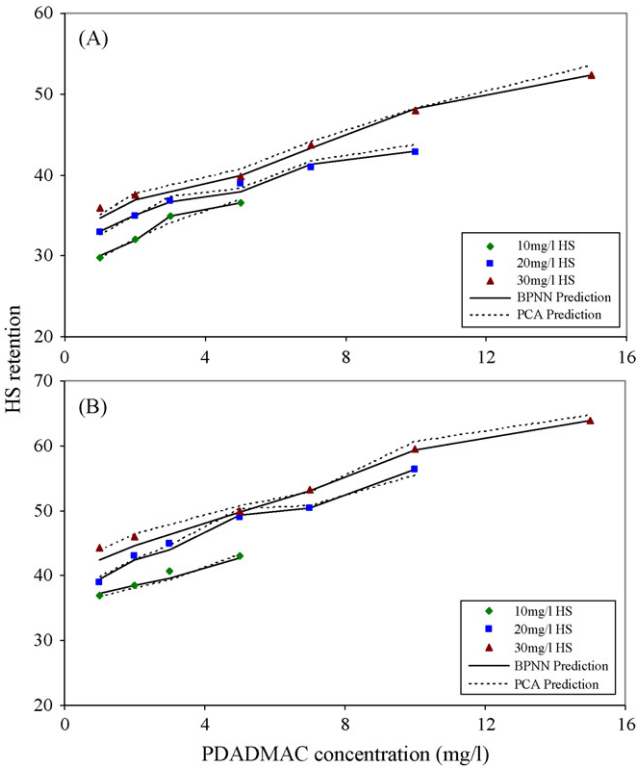

Fig. 9. BPNN and PCA-BPNN performance in predicting membrane fouling in HS and heavy metals experiments coagulation using P005F membrane operating at 3 bar and initial concentration of 5 mg/l heave metals: (A) 10,000 ppm NaCl and (B) 25,000 ppm NaCl.

metals and 2 mg/l PDADMAC concentration were 38.5, 38.5 and 38.0%, respectively.

5. Conclusions

Artificial neural network (ANN) can be a successful tool in membrane performance prediction, if developed efficiently. The developed BPNN produced high reliability; *R*-value exceeded 0.95 in all results. Training dataset results show that at low number of training dataset the learning process is hindered. A high number of training dataset can depress the generalizing abilities of the ANN through over-fitting or memorization of the training data set, reducing the networks predictability. In addition, number of neurons in hidden layers need to be chosen carefully to obtain a reliable network. Principle component analysis (PCA) can increase ANN simulation speed, albeit with a slight reduction in efficiency.

References

[1] M. Khayet, C. Cojocaru, C. Garcia-Payo, Application of response surface methodology and experimental design in direct contact membrane distillation, Ind. Eng. Chem. Res. 46 (2007) 5673–5685.
[2] R. Bowen, G. Jones, J. Welfoot, H.N. Yousef, Predicting salt rejections at nanofiltration membranes using artificial neural networks, Desalination 129 (2000) 147–162.
[3] W.R. Bowen, M.G. Jones, H.N.S. Yousef, Prediction of the rate of crossflow membrane ultrafiltration of colloids: a neural network approach, Chem. Eng. Sci. 53 (1998) 3793–3802.
[4] I.A. Basheer, N.R. Lakshmi, Y. Najjar, Site characterization by neuronets: an application to the landfill siting problem, Ground Water 34 (1996) 610–617.
[5] M. Leshno, V.Y. Lin, A. Pinkus, S. Schocken, Multilayer feedforward networks with a nonpolynomial activation function can approximate any function, Neural Networks 6 (1993) 861–867.
[6] S.S. Tambe, B.D. Kulkami, P.B. Deshpande, Elements of Artificial Neural Networks with Selected Applications in Chemical Engineering, and Chemical & Biological Sciences, Simulation & Advanced Controls Ltd., Louisville, KY, USA, 1996.
[7] H. Chen, A.S. Kim, Prediction of permeate flux decline in crossflow membrane filtration of colloidal suspension: a radial basis function neural network approach, Desalination 192 (2006) 415–428.
[8] S. Curcio, V. Calabro, G. Iorio, Reduction and control of flux decline in cross-flow membrane processes modelled by artificial neural networks, J. Membr. Sci. 286 (2006) 125–132.
[9] N.A. Darwish, N. Hilal, H. Al-Zoubi, A.W. Mohammad, Neural networks simulation of the filtration of sodium chloride and magnesium chloride solutions using nanofiltration membranes, Chem. Eng. Res. Des. 85 (2007) 417–430.
[10] G.B. Sahoo, C. Ray, Predicting flux decline in crossflow membranes using artificial neural networks and genetic algorithms, J. Membr. Sci. 283 (2006) 147–157.
[11] Y. Zhao, J.S. Taylor, S. Chellam, Predicting RO/NF water quality by modified solution diffusion model and artificial neural networks, J. Membr. Sci. 263 (2005) 38–46.
[12] A. Ghaffari, H. Abdollahi, M.R. Khoshayand, I.S. Bozchalooi, A. Dadgar, M. Rafiee-Tehrani, Performance comparison of neural network training algorithms in modeling of bimodal drug delivery, Int J Pharma 327 (2006) 126–138.
[13] M.T. Hagan, H.B. Demuth, M.H. Beale, Neural Network Design, PWS Published Company, Boston, MA, 1996.
[14] Assaad M, Bone R, Cardot H Study of the behavior of a new boosting algorithm for recurrent neural networks lecture notes in computer science 3697 (2005) 169-174.
[15] Q.P. Hu, M. Xie, S.H. Ng, G. Levitin, Robust recurrent neural network modeling for software fault detection and correction prediction, Reliab. Eng. Syst. Saf. 92 (2007) 332–340.
[16] S. Icer, S. Kara, A. Guven, Comparison of multilayer perceptron training algorithms for portal venous doppler signals in the cirrhosis disease, Expert Syst. Appl. 31 (2006) 406–413.
[17] H. Abdi, D. Valentin, B. Edelman, J. O'Toole A, A WidroweHoff, learning rule for a generalization of the linear auto-associator, Math. Psychol. 40 (1996) 175–182.
[18] E. El Tabach, L. Lancelot, I. Shahrour, Y. Najjar, Use of artificial neural network simulation metamodelling to assess groundwater contamination in a road project, Math. Comput. Modell. 45 (2007) 766–776.
[19] N. Hilal, M. Al-Abri, H. Al-Hinai, C. Somerfield, Combined humic substance and heavy metals agglomeration, and membrane filtration under saline conditions. Sep. Sci. Technol. (2008), in press.
[20] N. Hilal, M. Al-Abri, H. Al-Hinai, Combined humic substance coagulation and membrane filtration under saline conditions. Desalination, (2008), in press.
[21] N. Hilal, M. Al-Abri, A. Moran, H. Al-Hinai, Effects of heavy metals and polyelectrolytes in humic substance coagulation under saline conditions, Desalination 220 (2008) 85–95.
[22] P. Pradanos, J.I. Arribas, A. Hernandez, Mass transfer coefficient and retention of PEGs in low pressure cross-flow ultrafiltration through asymmetric membranes, J. Membr. Sci. 99 (1995) 1–20.
[23] H.K. Cigizoglu, M. Alp, Generalized regression neural network in modeling river sediment yield, Adv. Eng. Software 37 (2006) 63–68.
[24] H. Ghedira, M. Bernier, The effect of some internal neural network parameters on SAR texture classification performance, IEEE Geosci. Rem. Sens. Symp. 6 (2004) 3845–3848.
[25] N. Garcia-Pedrajas, D. Ortiz-Boyer, C. Hervas-Martinez, An alternative approach for neural network evolution with a genetic algorithm: crossover by combinatorial optimization, Neural Networks 19 (2006) 514–528.

[26] H. Saxen, F. Pettersson, Method for the selection of inputs and structure of feedforward neural networks, Comput. Chem. Eng. 30 (2006) 1038–1045.
[27] N. Delgrange, C. Cabassud, M. Cabassud, L. Durand-Bourlie, J.M. Lain, Neural network for prediction of ultrafiltration transmembane pressure—application to drink water, J. Membr. Sci. 150 (1998) 111–123.
[28] H. Niemi, A. Bulsari, S. Palosaar, Simulation of membrane separation by neural networks, J. Membr. Sci. 102 (1995) 185–191.
[29] R. Ceylan, Y. Ozbay, Comparison of FCM, PCA and WT techniques for classification ECG arrhythmias using artificial neural network, Expert Syst. Appl. 33 (2007) 286–295.
[30] J.E. Jackson, A User's Guide to Principal Components, first ed., John Wiley and Sons, 1991.
[31] Y. Lim, Y. Moon, T. Kim, Artificial neural network approach for prediction of ammonia emission from field-applied manure and relative significance assessment of ammonia emission factors, Eur. J. Agron. 26 (2007) 425–434.
[32] S. Osowski, T.H. Linh, ECG beat recognition using fuzzy hybrid neural network, IEEE Trans. Biomed. Eng. 48 (2001) 1265–1271.
[33] H. Demuth, M. Beale, Neural Network Toolbox User's Guide, vol. 4, first ed., The MathWorks, 2002.