

Prediction of permeate flux decline in crossflow membrane filtration of colloidal suspension: a radial basis function neural network approach

Huaiqun Chen, Albert S. Kim

*Department of Civil and Environmental Engineering, University of Hawaii at Manoa, Honolulu, Hawaii 96822
Tel. +1 (808) 965-3718; Fax +1 (808) 956-5014; email: Albert.S.Kim@hawaii.edu*

Received 6 July 2005; accepted 22 July 2005

Abstract

The capability of a radial basis function neural network (RBFNN) to predict long-term permeate flux decline in crossflow membrane filtration was investigated. Operating conditions of transmembrane pressure and filtration time along with feed water parameters such as particle radius, solution pH, and ionic strength were used as inputs to predict the permeate flux. Simulation results indicated that a single RBFNN accurately predicted the permeate flux decline under various experimental conditions of colloidal membrane filtrations and eventually produced better predictability than those of the regular multi-layer feed-forward backpropagation neural network (BPNN) and the multiple regression (MR) method. We believe further development of the artificial neural network approach will enable us to design and analyze full-scale processes from results of laboratory and/or pilot-scale experiments.

Keywords: Artificial neural network; Radial basis function; Backpropagation; Multiple regression; Membrane filtration; Colloidal fouling

1. Introduction

Compared to conventional water and wastewater treatment systems, membrane separation is a more compact operation that provides higher quality products, easier operational control, and

lower maintenance costs. Microfiltration (MF) and ultrafiltration (UF), which are widely used to remove colloidal particles and microorganisms, can serve as alternative or supplementary processes to water and wastewater treatment systems.

Based on the flow direction with respect to the membrane surface, membrane filtration processes

*Corresponding author.

can be generally classified into two categories: dead-end and crossflow filtrations. The former (without periodic backwashing) is not of practical interest because of its rapid flux decline intrinsically stemming from unceasing mass accumulation of suspended particles in the feed solution, while the tangential feed flow over the membrane surfaces in the crossflow case carries suspended solids downstream due to fluid shear forces. In MF/UF processes, the major aim for efficient filtration is to minimize permeate flux decline caused by the accumulation of particles on the membrane surface and/or within membrane pores.

Fouling is an unavoidable deleterious phenomenon in the filtration process that causes serious permeate flux decline followed by deterioration of product water quality. Particle accumulation, a major fouling mechanism in MF and UF, can be typically classified into two processes: concentration polarization (CP) and cake layer formation between the CP layer and the membrane surface [1–8]. The solute (i.e., particles, in this study) CP layer reaches its steady state after a short time period when the filtration process begins. Thereafter, suspended particles continuously accumulate and form a dense cake layer which becomes the major resistance to the permeate water flux. Several studies on permeate flux decline indicate that particle accumulation during crossflow colloidal membrane filtration is affected by physical and chemical characteristics of the feed water as well as hydrodynamic operating conditions [9–12].

Design and application of membrane filtration require development of predictive models that can inter-link solution properties and operating conditions to filtration performance. To predict permeate flux decline, many theoretical models have been proposed using physical, chemical, and hydrodynamic modeling parameters such as particle size, zeta potential, Hamaker constant, temperature, solution pH, ionic strength, transmembrane pressure, shear rate, and so forth [13]. In most practical applications of MF/UF, effects of the CP layer are barely measurable since the cake layer

forms shortly after the CP layer is generated. Thus, one of the main focuses of flux decline models is on the time-dependent variation of the cake layer resistance during filtration while the CP layer's effect is considered to be negligible [8,10,14,15].

Film theory, gel layer, osmotic pressure, and boundary layer resistance models are several of the theoretical approaches that have been developed [1,2,16]. However, the accuracy of theoretical modeling has been hampered by insufficient knowledge regarding the complexity of the micro-scale phenomena occurring during the filtration processes. In addition, model designs involve specific assumptions that may lead to deviations between theoretical predictions and experimental observations. Such deviations imply that fundamental theoretical models sometimes do not have sufficient capability to depict fouling phenomena occurring in real filtration systems, although the models accurately depict basic underlying transport mechanisms. Purely empirical models, which can be built upon specific experimental observations and then used to predict the membrane system performance, appear to be new alternatives. One such method is the artificial neural network (ANN) approach.

The primary advantage of ANN over theoretical approaches is that, as a black box model, it does not require any governing equation with forceful assumptions specifically describing the underlying engineering phenomena. ANN can learn the complex transport processes of a system from given inputs and observed outputs, serving as an instrument for universal data approximation. Accordingly, a considerable number of ANN applications for prediction, classification, signal processing, and optimization can be found in various fields of finance and economics, meteorology, chemistry and chemical engineering, computer science, and so forth [17–20]. Those applications exemplify the ANN's great potential for predicting transport phenomena.

Extensive research has been performed to investigate the capability of an ANN as a tool for

membrane system modeling [21–27]. As a consequence, application of an ANN to membrane study is an attractive topic. For example, ANN has been harnessed to predict steady-state contaminant removal efficiency during nanofiltration in the drinking water treatment system [24]. ANN's potential for dynamic simulation of membrane fouling during crossflow microfiltration has also been investigated [21]. Additional applications can be found in predicting permeate flux, hydraulic resistance, and rejection for various types of feed solutions [22,23,25,26]. Moreover, ANN's ability for classification has been shown in the development of a phenomenological model to categorize membrane pore sizes using the ultrasonic signal amplitude [27].

Among assorted neural networks, the multi-layer feed-forward neural network with backpropagation training algorithm, typically called backpropagation neural network (BPNN), may be one of the most widely used networks for information extraction and classification [21–27]. However, BPNN exhibits several serious shortcomings, such as the slow convergence during its training step, susceptibility to converging to a local minimum, and inability to detect over-fitting [28]. For these aspects, the radial basis function neural network (RBFNN) seems a better alternative to BPNN as RBFNN provides easier initialization, faster training procedure, and more stable performance.

To the best of our knowledge, any RBFNN application to membrane systems has not been reported yet. The goal of this study is therefore to study an information-integrated RBFNN for membrane filtration analysis and to explore its predictability of permeate flux under various operating conditions and solution characteristics. A brief introduction to RBFNN is given, and its effectiveness is compared to those of BPNN and the conventional multiple regression (MR) method.

2. Artificial neural networks

Artificial neural network (ANN) is a parallel-

distributed information-processing system [29] with a large number of neurons and connections. The uniqueness of ANN lies in its ability to learn and generate interrelationships between the input and output of observed (or experimental) data without requiring any postulates and assumptions. ANNs can be classified into different categories by their network architecture, activation or transfer function, and training algorithm [30].

Generally, neurons in an ANN are arranged in input, hidden, and output layers and linked to others with associated weights and biases, which will be adjusted to optimal values during the training. The Neural Network Toolbox available in MATLAB (Mathworks, Inc., Natick, MA) [31] is implemented in this study to design and train the BPNN and RBFNN. Later, the two neural networks, proven to be superior to the multiple regression (MR) method, are compared in terms of their performance.

2.1. Backpropagation neural network

As described above, the multi-layer feed-forward BPNN is one of the most widely employed ANN models in membrane research, of which introductory overviews can be found in several studies [21–27]. The settings of the BPNN used in this study are briefly described below.

The tan-sigmoid function is the transfer function employed in the hidden layers and has the following form:

$$f(x) = \tan\left(\frac{1}{1 + e^{-x}}\right) \quad (1)$$

where x is the input. This function of Eq. (1) is bounded between -1 and 1 so input data are usually normalized within the same range. Neurons in the output layer receive weighted input and generate the final output using a linear transfer function.

Among the many BPNN training methods available in the MATLAB Toolbox, the Levenberg–Marquardt training algorithm is selected

because of its fast convergence rate [31]. During the backpropagation training process, the suitability of the simulation is validated by estimating the mean square error (MSE) between the observed and ANN-simulated data. In our study, the maximum number of epochs, target error goal MSE, and minimum performance gradient are set to 400, 10^{-10} , and 10^{-6} , respectively. Training stops when the maximum number of epochs is reached or when either the MSE or performance gradient is minimized to arrive at the pre-determined goal. Through a trial-and-error method, the optimal network structure for the current BPNN simulation is determined to include two hidden layers, one containing 4 and the other containing 2 hidden neurons.

2.2. Radial basis function neural network

RBFFNN is another type of neural network. Such networks have 3 layers, i.e., the input layer, the nonlinear hidden layer, and the linear output layer. RBFFNN can overcome some of the limitations of BPNN by using a rapid training phase, having a simple architecture, and maintaining complicated mapping abilities.

In most cases, the RBFFNN simulates phenomena of interest by using Gaussian basis functions in the hidden layer and linear transfer functions in the output layer. The major difference in operation between RBFFNN and BPNN specifically exists in the hidden layer. Instead of the weighted sum of the input vector used in BPNN, the distance between the input and center, explained below, is employed in the RBFFNN learning process.

The first layer of the RBFFNN collects the input data. Its training process determines the number of hidden neurons, m , which can be larger than that of BPNN to achieve a certain accuracy of prediction [31]. For each neuron in the hidden layer, the distance between the input data and the center is activated by a nonlinear radial basis function, as shown in the following equation:

$$R_i = \exp\left[-\left(\|x_i - c_i\|/b_1\right)^2\right] \quad (2)$$

where x is the input vector, and b_1 and c_i are parameters that represent the bias in the hidden layer and center vector, respectively. Each neuron in the hidden layer will produce a value between 0 and 1, according to how close the input is to the center location. Therefore, neurons with centers closer to inputs will have more contributions to outputs; on the other hand, if neurons have centers away from inputs, then their outputs are nullified and so vanished. Later, the output layer neurons receive the weighted inputs and produce results by using a linear combination, which is of a similar form to that of the BPNN:

$$\hat{y} = \sum_{i=1}^m w_i R_i(x) + b_2 \quad (3)$$

where \hat{y} is the RBFFNN simulation result (output), w_i is the optimized connection weight determined through the training process [29], and b_2 is the bias in the output layer.

Fig. 1 gives an insight into the structure and working procedure of the RBFFNN. The open circles are bias neurons that adjust the sensitivity of the network. The biases are controlled by a specific value, called “spread” number, and each bias is set to $0.8326/\text{spread}$ [31]. The selected “spread” value should be large enough for neurons in the hidden layer to include the whole range of input data.

In RBFFNN simulations, the proper initial choice of centers and weights should be regarded as key issues. Various methods are proposed to initially locate the center, e.g., random selection, K -mean technique, max-min algorithms, etc. Using the Neural Network Toolbox, we choose the weight vectors as the center parameter. Instead of being randomly generated, the initial value of the weights is set as the transpose of the input vector. Once the centers are developed, the weights linking the hidden and output layers should be updated during the training procedure.

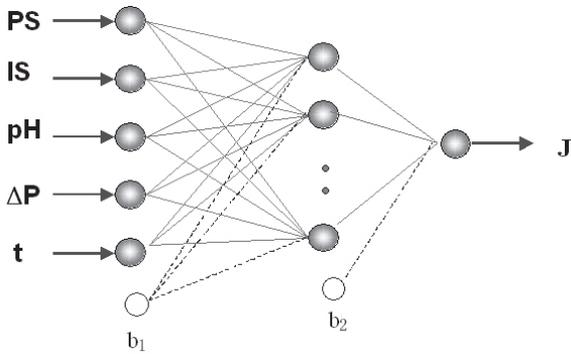


Fig. 1. Architecture of RBFNN employed in this study. RBFNN inputs consist of particle size (PS), ionic strength (IS), solution pH, transmembrane pressure (ΔP), and elapsed time (t), which are used as input parameters to predict the permeate flux (J). b_1 and b_2 are biases in the input and hidden layers, respectively.

Training is an optimization procedure in which the network weights are adjusted in order to minimize the selected error value. The training procedure of the RBFNN (unlike the BPNN) determines the number of hidden neurons required for the simulation. In this study, the root mean square error (RMSE) is the function used to estimate the performance of the neural network:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4)$$

where y is the target output value (or observed data), \hat{y} is the neural network output, and n is the total numbers of data patterns (input and output pairs) used. The training of RBFNN is initiated by generating a single neuron in the hidden layer, followed by continuously adding neurons to the hidden layer one at a time.

Before training starts, the key training parameters should be assigned. In our study, the target error goal MSE and the maximum number of neurons are set to 10^{-2} and 80, respectively. Training is stopped if the MSE is minimized to less than the pre-determined goal or if the maximum

number of neurons is reached. Because the spread value affects the quality of prediction, a trial-and-error method is employed, wherein several spread numbers are tested and the best one is accepted. By doing so, we find that the optimal spread number is 80.

2.3. ANN inputs and outputs

The data used in this study are adopted from Faibish et al.'s work [9]. In their experimental study, effects of solution pH, ionic strength (IS), particle size (PS), and transmembrane pressure (ΔP) on the permeate flux (J) in crossflow membrane filtration were examined. In order to investigate the effect of a particular parameter in each experiment, other parameters were fixed and the permeate flux was measured accordingly. The particle sizes of suspended solids (SiO_2) were greater than the membrane pore size so that complete rejection was observed. Various other operational parameters were also involved, such as particle bulk concentration, temperature, and cross-flow velocity.

In order to reduce the complexity and improve the performance of the network simulation in our study, however, the least number of inputs is selected: particle size (PS), ionic strength (IS), solution pH, transmembrane pressure (ΔP), and elapsed time (t) of the filtration process. The IS value is expressed in logarithmic scale in the simulation due to its incongruous value. The network is constructed to predict the variation of permeate flux with respect to filtration time under the above physical, chemical, and hydrodynamic operation parameters.

Approximately 17% of all the experimental data is used for training, and the remaining 83% is used for model performance verification. As all empirical models are only able to interpolate between the application boundary values [24], the training data should be representative of the entire range of experimental conditions. Therefore, extreme values of the experimental data need to be included as part of the training set.

3. Results and discussion

Changing the physical, chemical, and operating conditions is a useful technique to determine their effects on the permeate flux. Experimental data, according to Faibish et al.'s work [9], are divided into three sets to represent the different parameters' effects on permeate flux. To ensure that the trained RBFNN can be used for performance prediction under various conditions, sets of six training points (equally spaced in time) are selected from permeate samples in those three data sets. In this section, our model simulation results are presented and compared to Faibish et al.'s experimental data.

3.1. Effect of solution pH

In order to study the effects of pH on the permeate flux, the solution pH value was regulated by adding NaHCO_3 or NaOH to the feed solution [9]. Figs. 2 and 3 show comparisons of the RBFNN predictions and the experimentally ob-

served permeate fluxes at various pH values of the feed solution (10^{-1} and 10^{-3} M KCl). Intermittent data (as a part of the experimental data), shown as \times in Figs. 2 and 3, are selected from the permeate flux profile of pHs 6.1 and 10.0 and then used for training to predict all the flux decline behaviors at pHs of 6.1, 8.3, and 10.0. The results generally depict the excellent performance of the RBF network as it accurately interpolates and reproduces the complex nonlinear profiles of the transient permeate flux decline under various solution pHs and ionic strengths.

The correlation coefficient between the observed and predicted permeate flux is as high as 0.993, although minor departures are observed during the early stages of filtration, when permeate flux degenerates dramatically. The paired t-tests indicated that there is no statistical difference between the experimental values and neural network predictions at the 99% confidence level for both high (10^{-1} M) and low (10^{-3} M) ionic strengths.

Both model prediction and experimental data

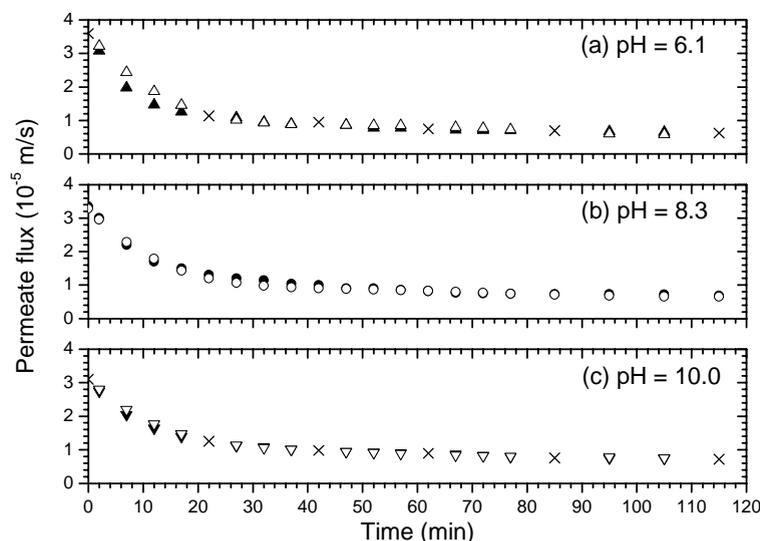


Fig. 2. Experimental data and ANN predictions on the effect of solution pH on permeate flux decline with ionic strength of 10^{-1} M KCl. The filtration conditions employed are particle size 47 nm, transmembrane pressure 41.4 kPa (6 psi), particle volume concentration 0.01%, crossflow velocity 0.246 m/s, and temperature 20°C . At each pH value, the solid and the hollow symbols represent experimentally observed and ANN-simulated data while \times indicates points selected from the experimental data and used for the training process.

demonstrate that the feed water pH has only minor effects on the permeate flux. Comparing Figs. 2 and 3 indicates that the initial permeate flux slightly decreases with pH, and this behavior is captured by the RBFNN. However, the trained RBFNN does not accurately duplicate the almost identical permeate fluxes at pHs of 8.3 and 10.0 (Fig. 3). The neural network slightly over-predicts the initial permeate flux at a pH of 8.3 for the first 10 min of filtration. We think this over-prediction stems from the preparing the training data set only from pHs 6.1 and 10.0 cases, excluding pH 8.3 data.

Comparison of Figs. 2 and 3 also shows that the initial permeate flux increases at higher ionic strength and lower pH(s) due to the narrow capillary effect on fluid viscosity. At the initial stage, when particle deposition is negligible, the intrinsic membrane resistance dominates. Under low pH and high ionic strength conditions, the decreasing fluid viscosity near membrane pores allows water to penetrate the membrane pores with relative ease, leading to the initial enhanced permeate flux.

However, at the later stages of membrane filtration when cake resistance becomes dominant, the steady-state permeate fluxes become independent of the solution pH.

3.2. Effect of ionic strength

The RBFNN’s predictability of the influence of ionic strength on the permeate flux was investigated using three different particle sizes of 47 nm, 110 nm, and 310 nm and four ionic strength values of 10^{-4} , 10^{-3} , 10^{-2} , and 10^{-1} M KCl [9]. The experimental data and modeling results are shown in Fig. 4, which clearly shows the excellent predictability of the RBFNN for permeate flux over a wide range of ionic strengths and particle sizes. The predictions and the experimental data do not show a noticeable difference with a correlation coefficient of 0.9944 and RMSE of 0.076 at the 99% confidence level throughout the entire range of transient permeate flux.

As expected, increasing ionic strength pro-

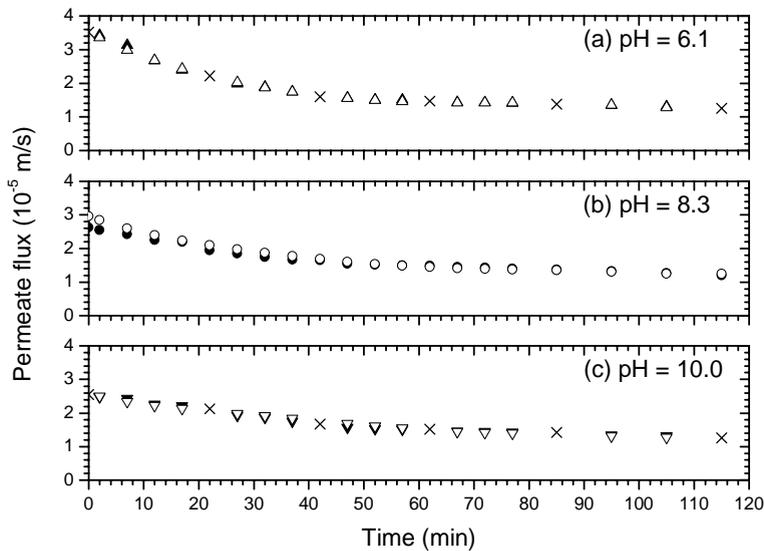


Fig. 3. Experimental data and ANN predictions on the effect of solution pH on permeate flux decline with ionic strength of 10^{-3} M KCl. The filtration conditions employed are particle size 47 nm, transmembrane pressure 41.4 kPa (6 psi), particle volume concentration 0.01%, crossflow velocity 0.246 m/s, and temperature 20°C. At each pH value, the solid and the hollow symbols represent experimentally observed and ANN-simulated data while × indicates points selected from the experimental data and used for the training process.

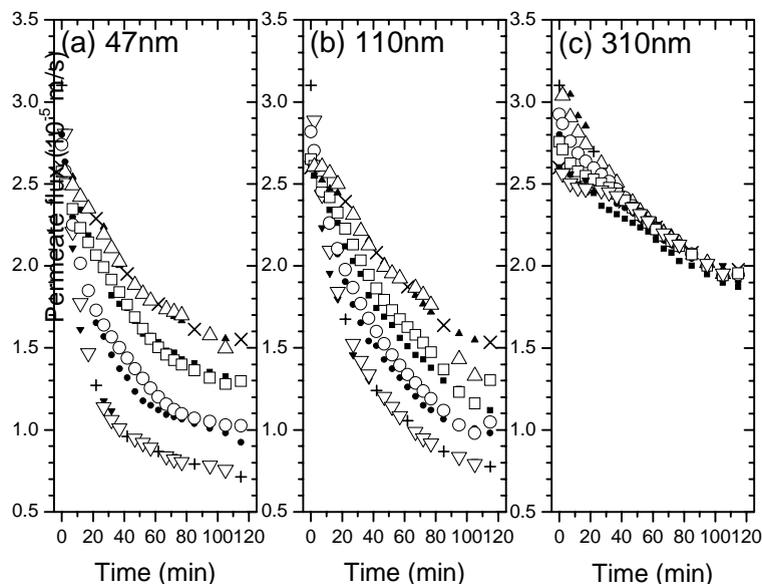


Fig. 4. Experimental data and ANN predictions on the effect of ionic strength on permeate flux decline for (a) 47 nm, (b) 110 nm, and (c) 310 nm particles. The filtration conditions employed are pH 10, transmembrane pressure 41.4 kPa (6 psi), particle volume concentration 0.01%, crossflow velocity 0.246 m/s, and temperature 20°C. For each particle size, the symbols \times and $+$ indicate the selected experimental data points of ionic strength of 10^{-4} and 10^{-1} M, respectively, for the training process. The solid symbols indicate experimentally observed data of ionic strength 10^{-4} M (\blacktriangle), 10^{-3} M (\blacksquare), 10^{-2} M (\bullet), and 10^{-1} M (\blacktriangledown) of KCl, while the corresponding hollow symbols represent the simulation results.

motes more severe permeate flux decline and accelerates the rate of reaching a limiting flux level because the electrostatic double-layer repulsion is mollified by the increasing ionic strength, causing a dense cake layer of enhanced hydraulic resistance. In Figs. 4a and 4b, the RBFNN closely captures the underlying fundamental effects of the double-layer repulsion through various ionic strengths on the permeate flux. On the other hand, Fig. 4c reveals quite a different trend of transient fluxes from those of Figs. 4a and 4b because the flux profiles of Fig. 4c seem to converge into a single limiting permeate flux level. This behavior indicates that the influence of larger particle size (inducing larger pore size) can suppress the effect of interparticle interactions, leaving hydrodynamics as the dominating mechanism in flux decline. The trained RBFNN shows its ability to predict flux decline for ionic strengths of 10^{-2} and 10^{-3} M by learning two different types of under-

lying phenomenological mechanisms, i.e., interparticle and hydrodynamic interactions.

3.3. Effect of transmembrane pressure

The ability of RBFNN to simulate the transmembrane pressure effect on the permeate flux is examined in this section. Results shown in Fig. 5 demonstrate that the RBFNN model can closely predict the permeate flux under different transmembrane pressures of 62.1, 41.4, and 20.7 kPa [9]. The accuracy is reflected by a correlation coefficient and RMSE of 0.994 and 0.092, respectively.

Fig. 5 shows that the permeate flux tends to reach an almost identical limiting flux under the same ionic strength regardless of applied transmembrane pressure and initial flux. As expected, Fig. 5 confirms that flux decline is accelerated by increasing transmembrane pressure and ionic

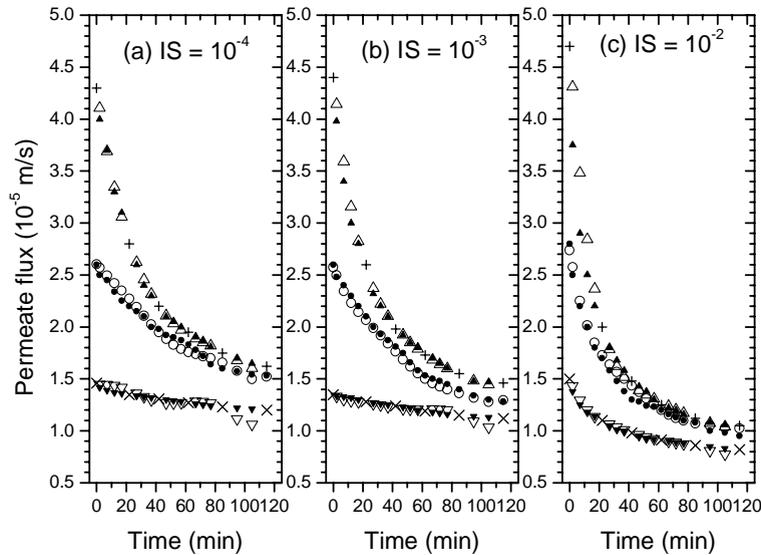


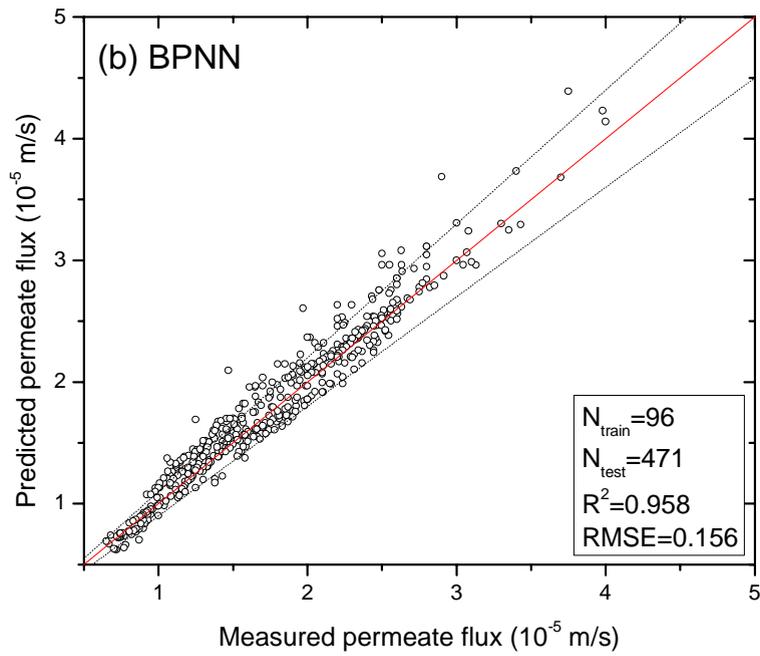
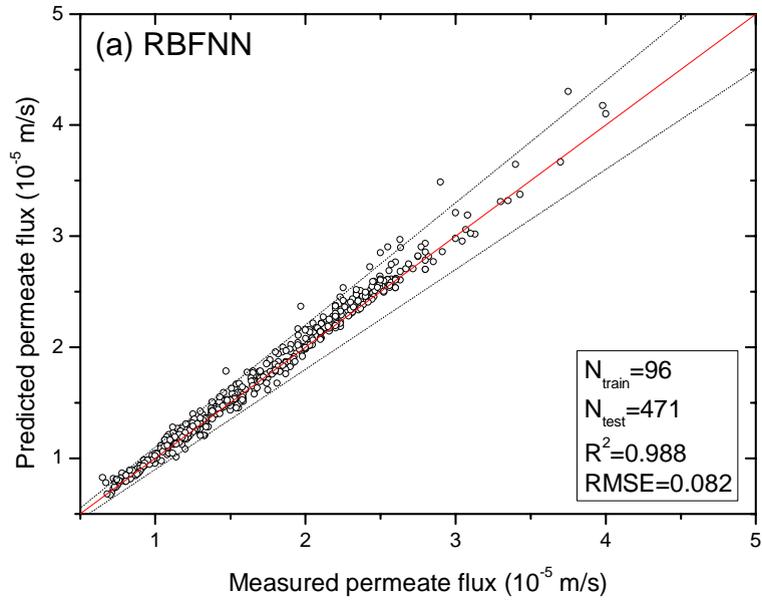
Fig. 5. Experimental data and ANN predictions on the effect of transmembrane pressure on permeate flux decline for ionic strengths of (a) 10^{-4} M, (b) 10^{-3} M, and (c) 10^{-2} M KCl. The filtration conditions employed were pH 10, particle size 47 nm, particle volume concentration 0.01%, crossflow velocity 0.246 m/s, and temperature 20°C . For each particle size, the symbols \times and $+$ indicate the selected experimental data points of transmembrane pressure of 62.1 kPa (9 psi) and 20.7 kPa (3 psi), respectively, for the training process. The solid symbols indicate experimentally observed data of transmembrane pressure 62.1 kPa (9 psi) (\blacktriangle), 41.4 kPa (6 psi) (\bullet), and 20.7 kPa (3 psi) (\blacktriangledown), while the corresponding hollow symbols represent the simulation results.

strength (as described in section 3.2), although the pressure generates higher initial flux. A higher transmembrane pressure elevates the initial permeate flux by pushing more water toward a clean membrane while the development of particle concentration polarization is still under way. Contrary to the gain in the initial stage, the higher transmembrane pressure at a later stage pulls more particles toward the membrane surface by providing faster permeate velocity and thus contributes to the rapid generation of the deposited cake layer which causes a dramatic increase of hydraulic resistance. Moreover, the higher pressure significantly compresses the pre-built cake layer so it becomes denser and additionally contributes to the rapid flux decline. In addition, gradual flux decline is observed when both the initial flux and ionic strength are low, which is reflected by the slow and sparse formation of the cake layer.

3.4. Performance comparison

As noted in the introduction, BPNN was widely used in the past decade as the standard ANN method to model membrane performance. This section describes the superiority of RBFNN over BPNN by comparing their performances using 96 ($=N_{\text{train}}$) filtration data points for training and 471 ($=N_{\text{test}}$) points for testing out of total 567 ($=N_{\text{total}}$) points of permeate flux.

Fig. 6a summarizes the RBFNN predictions for various operational conditions in Figs. 2–5 by plotting predicted values as a function of observed experimental values. The solid line in Fig. 6a represents the perfect 1:1 match of the measured and predicted values of the permeate flux, and the dotted lines depict $\pm 10\%$ relative error from the 1:1 match line. Most of data shown in Fig. 6a are located near the 1:1 match straight line. Only



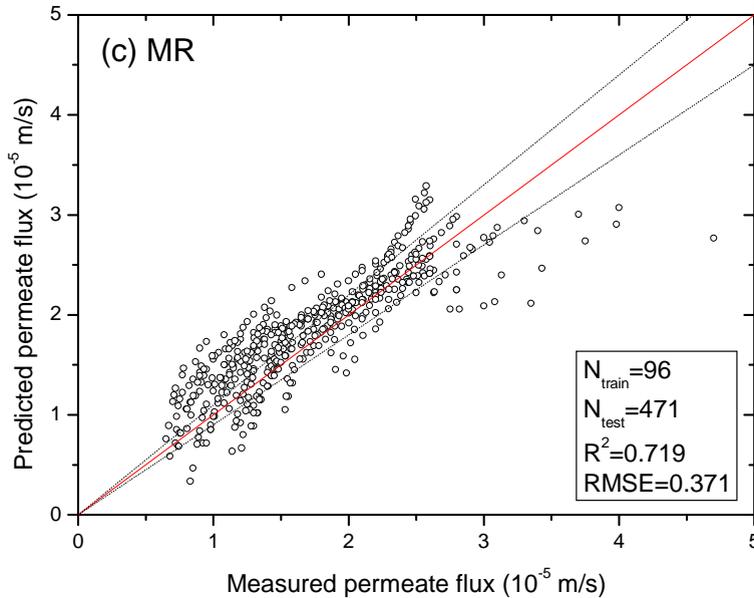


Fig. 6. (a) RBFNN, (b) BPNN, and (c) MR predictions of permeate flux with experimental measurement under different operational conditions. The solid straight line represents the line of equality and the dashed lines represent $\pm 10\%$ variance from it. The R -square and RMSE between the predicted and measured data are also provided.

14 ($=N_{10}$) out of $N_{\text{test}} = 471$ predicted (tested) points lie outside the region enclosed by the two dotted lines, indicating that 97% of the points are within the $\pm 10\%$ relative error. (N_{10} stands for the number of points beyond $\pm 10\%$ relative error.) This excellent performance of RBFNN is also quantified by an RMSE of 0.082 and an R -square (R^2) of 0.988. In spite of broad variations of feed water parameters and operational conditions, which cause complexity to the filtration system, overall the results indicate that employing only 17% (96 out of 567 points) of the total data for training renders satisfactory prediction of the remaining 83% (471 out of 567 points) of the data.

Previous investigations have suggested that RBFNNs have multiple advantages in comparison with the performance of BPNNs. In order to verify this concept, an identical simulation was carried out using BPNN, and the result is presented in Fig. 6b. The hollow circles are crowded near the 1:1 match line, suggesting that BPNN is also able to predict

the permeate flux under different operational status. However, when compared with the RBFNN prediction in Fig. 6a, BPNN produces a somehow more scattered plot. Sixty-one ($=N_{10}$) out of 471 points are not confined within the two dotted lines. (Note that N_{10} of the RBFNN is 14.) Only 87% of the predicted points are within acceptable error in this case, whereas 97% were within $\pm 10\%$ relative error in the RBFNN case.

To quantitatively prove the superiority of the neural networks, we do the same calculation using the conventional MR method, of which its linear equation, obtained by the identical training data, has the following form:

$$J = 0.943 + 0.025 [\Delta P] + 0.1663 \log_{10} [\text{IS}] - 0.0192 [\text{pH}] + 0.0028 [\text{PS}] - 0.0133 [t] \quad (5)$$

The regression equation is also used to predict the permeate flux of the remaining data. Results for MR are shown in Fig. 6c. It is clearly evident that,

Table 1

Comparison of the performance of RBFNN, BPNN, and MR in terms of R -square (R^2), root mean square error (RMSE), points beyond $\pm 10\%$ relative error (N_{10}), and simulation time, given the same number of total data set (N_{total}), training set (N_{train}), and testing set (N_{test})

	N_{total}	N_{train}	N_{test}	R^2	RMSE	N_{10}	Simulation time (s)
RBFNN	567	96	471	0.988	0.082	14	2.47
BPNN	567	96	471	0.958	0.156	61	96.32
MR	567	96	471	0.755	0.332	244	0.046

given the same information, MR provides significantly less accurate performance, which is mainly because the complex nonlinear membrane filtration process could not be presented by the simple linear method. Compared with those in Figs. 6a and 6b, hollow circles in Fig. 6c are much more deviated from the 1:1 match line. More than half (51.6%) of the predicted points (471) are out of the $\pm 10\%$ relative error range.

Finally, Table 1 compares the performance of RBFNN, BPNN, and MR using four parameters — R -square (R^2), RMSE, the number of points beyond the $\pm 10\%$ error (N_{10}) and required simulation time with identical conditions of N_{total} , N_{train} , and N_{test} . RBFNN predicts the experimental observations much more closely in comparison to BPNN and MR. Additionally, the RBFNN training method eventually speeds up the training process of the neural network by as much as 40 times the BPNN training rate, providing higher adaptability when a large set of observation data is used to investigate more complex filtration systems.

4. Conclusions

Due to the complex physicochemical micro-phenomena occurring during membrane filtration, conventional theoretical models have been able to predict the filtration procedure only under limited conditions with specific and/or inevitable assumptions. An artificial neural network method provides a unified approach for analysis and prediction of the membrane performance under

various combinations of operational conditions. In this study, the RBFNN is applied to predict the permeate flux on crossflow membrane filtration as a function of transmembrane pressure, ionic strength, solution pH, particle size, and elapsed filtration time. Results show that the transient profiles of the permeate flux during crossflow membrane filtration can be predicted by a single RBFNN with acute accuracy, given a limited number of training points. Comparison of the performances of RBFNN, BPNN, and MR confirms the superiority of RBFNN in terms of the R -square, RMSE, N_{10} and simulation time required of higher accuracies.

As a consequence, it may not be necessary to carry out an entire series of expensive pilot or full-scale tests to collect and verify filtration data. The RBFNN can interpolate the performance of membrane filtration under other conditions of interest by using widely ranged and sparse data points to reduce the time and costs required. Although pilot-scale experimental tests restrict the number of operating parameters that generally are of importance in larger-scale applications, they still can closely simulate full-scale operations. In the same manner, RBFNN can also serve as potential prediction tools for full-scale operations, possibly combining several trained networks of different scales into one smart neural network.

Acknowledgement

This research was supported by Saehan Indust-

ries, Seoul, Korea and the US Geological Survey (Grant 01HQGR0079), through the Water Resources Research Center, University of Hawaii at Manoa, Honolulu, HI. This is contributed paper CP-2005-11 of the Water Resources Research Center, University of Hawaii at Manoa, Honolulu.

References

- [1] S. Bhattacharjee, A.S. Kim and M. Elimelech, Concentration polarization of interacting solute particles in cross-flow membrane filtration, *J. Colloid Interf. Sci.*, 212 (1999) 81–99.
- [2] M. Elimelech and S. Bhattacharjee, A novel approach for modeling concentration polarization in crossflow membrane filtration based on the equivalence of osmotic pressure model and filtration theory, *J. Membr. Sci.*, 145 (1998) 223–241.
- [3] A.S. Kim and E.M.V. Hoek, Cake structure in dead-end membrane filtration: Monte Carlo simulation, *Environ. Eng. Sci.*, 19 (2002) 373–386.
- [4] J.C. Chen, M. Elimelech and A.S. Kim, Monte Carlo simulation of colloidal membrane filtration: model development with application to characterization of colloid phase transition, *J. Membr. Sci.*, 255 (2005) 291–305.
- [5] L. Song, Flux decline in crossflow microfiltration and ultrafiltration: mechanisms and modeling of membrane fouling, *J. Membr. Sci.*, 139 (1998) 183–200.
- [6] S. Hong, R.S. Faibish and M. Elimelech, Kinetics of permeate flux decline in crossflow membrane filtration of colloidal suspensions, *J. Colloid Interf. Sci.*, 196 (1997) 267–277.
- [7] S. Bhattacharjee, J.C. Chen and M. Elimelech, Coupled model of concentration polarization and pore transport in crossflow nanofiltration, *AIChE J.*, 47 (2001) 2733.
- [8] L. Song and M. Elimelech, Theory of concentration polarization in crossflow filtration, *J. Chem. Soc., Faraday Trans.*, 91 (1995) 3389–3398.
- [9] R.S. Faibish, M. Elimelech and Y. Cohen, Effect of interparticle electrostatic double layer interactions on permeate flux decline in crossflow membrane filtration of colloidal suspensions: an experimental investigation, *J. Colloid Interf. Sci.*, 204 (1998) 77–86.
- [10] C.A. Romero and R.H. Davis, Transient model of crossflow microfiltration, *Chem. Eng. Sci.*, 45 (1990) 13–25.
- [11] P. Bacchin, P. Aimar and V. Sanchez, Influence of surface interaction on transfer during colloid ultrafiltration, *J. Membr. Sci.*, 115 (1996) 49–63.
- [12] R.M. McDonogh, A.G. Fane and C.J.D. Fell, Charge effects in the cross-flow filtration of colloids and particulates, *J. Membr. Sci.*, 43 (1989) 69–85.
- [13] W.R. Bowen and F. Jenner, Theoretical descriptions of membrane filtration of colloids and fine particles: an assessment and review, *Advances Colloid Interf. Sci.*, 56 (1995) 141–200.
- [14] C.A. Romero and R.H. Davis, Global model of crossflow microfiltration based on hydrodynamic particle diffusion, *J. Membr. Sci.*, 39 (1988) 157–185.
- [15] P. Bacchin, D. Si-Hassen, V. Starov, M.J. Clifton and P. Aimar, A unifying model for concentration polarization, gel-layer formation and particle deposition in cross-flow membrane filtration of colloidal suspensions, *Chem. Eng. Sci.*, 57 (2002) 77–91.
- [16] J.G. Wijmans, S. Nakao and C.A. Smolders, Flux limitation in ultrafiltration — osmotic pressure model and gel layer model, *J. Membr. Sci.*, 20 (1984) 115–124.
- [17] M.C. Valverde Ramirez, H.F. De Campos Velho and N.J. Ferreira, Artificial neural network technique for rainfall forecasting applied to the Sao Paulo region, *J. Hydrology*, 301 (2005) 146–162.
- [18] O. Maas, J.-P. Boulanger and S. Thiria, Use of neural networks for predictions using time series: illustration with the El Nino southern oscillation phenomenon, *Neurocomputing*, 30 (2000) 53–58.
- [19] G.J. Bowden, H.R. Maier and G.C. Dandy, Input determination for neural network models in water resources applications. Part 2. Case study: forecasting salinity in a river, *J. Hydrology*, 301 (2005) 93–107.
- [20] G.J. Bowden, G.C. Dandy and H.R. Maier, Input determination for neural network models in water resources applications. Part 1. Background and methodology, *J. Hydrology*, 301 (2005) 75–92.
- [21] M. Dornier, M. Decloux, G. Trystram and A. Lebert, Dynamic modeling of crossflow microfiltration using neural networks, *J. Membr. Sci.*, 98 (1995) 263–273.
- [22] M.A. Razavi, A. Mortazavi and M. Mousavi, Dynamic modelling of milk ultrafiltration by artificial neural network, *J. Membr. Sci.*, 220 (2003) 47–58.

- [23] W.R. Bowen, M.G. Jones and H.N.S. Yousef, Dynamic ultrafiltration of proteins — a neural network approach, *J. Membr. Sci.*, 146 (1998) 225–235.
- [24] G.R. Shetty, H. Malki and S. Chellam, Predicting contaminant removal during municipal drinking water nanofiltration using artificial neural networks, *J. Membr. Sci.*, 212 (2003) 99–112.
- [25] H. Niemi, A. Bulsari and S. Palosaari, Simulation of membrane separation by neural networks, *J. Membr. Sci.*, 102 (1995) 185–191.
- [26] M. Chakraborty, C. Bhattacharya and S. Dutta, Studies on the applicability of artificial neural network (ANN) in emulsion liquid membranes, *J. Membr. Sci.*, 220 (2003) 155–164.
- [27] S. Ramaswamy, A.R. Greenberg and M.L. Peterson, Non-invasive measurement of membrane morphology via UFDR: pore-size characterization, *J. Membr. Sci.*, 239 (2004) 143–154.
- [28] R.J. Schilling, J.J. Carroll and A.F. Al-Ajlouni, Approximation of nonlinear systems with radial basis function neural networks, *Trans. on Neural Networks*, 12 (2001) 1–15.
- [29] American Society of Civil Engineers (ASCE), Task Committee on Application of Artificial Neural Networks in Hydrology, *Artificial Neural Networks in Hydrology. I: Preliminary concepts*, *J. Hydrol. Eng.*, 5 (2000) 115–123.
- [30] L. Fausett, *Fundamentals of Neural Networks*. Prentice Hall, Englewood Cliffs, NJ, 1994.
- [31] H. Demuth and M. Beale, *Neural Network Toolbox: For Use with Matlab*. Mathworks, Inc., Natick, MA, 2003.