



Modeling of effluent quality parameters in a submerged membrane bioreactor with simultaneous upward and downward aeration treating municipal wastewater using hybrid models

Majid Bagheri^{a,*}, Sayed Ahmad Mirbagheri^a, Ali Morad Kamarkhani^b, Zahra Bagheri^c

^aDepartment of Civil Engineering, K.N. Toosi University of Technology, Vanak square, Tehran, Iran, Tel. +98 9181331137; Fax: +98 2188770006; email: bagherimajead@yahoo.com (M. Bagheri), Tel. +98 9121374357; email: Mirbagheri@kntu.ac.ir (S.A. Mirbagheri)

^bDepartment of Chemical Engineering, Razi University, Kermanshah, Iran, Tel. +98 9186171061; email: Ali.ka1361@gmail.com

^cDepartment and Faculty of Basic Sciences, PUK University, Kermanshah, Iran, Tel. +98 8325245574; email: zahra90_bagheri@yahoo.com

Received 10 August 2014; Accepted 11 February 2015

ABSTRACT

This research was an effort to develop hybrid multilayer perceptron and radial basis function artificial neural network–genetic algorithm (MLPANN-GA and RBFANN-GA) models to accurately predict effluent biochemical oxygen demand (BOD), chemical oxygen demand (COD), total nitrogen (TN), and total phosphorus (TP) in a submerged membrane bioreactor. The input variables of the networks were influent BOD, influent COD, influent TN or influent TP, sludge retention time (SRT), mixed liquor suspended solid, membrane permeability, and transmembrane pressure. Training procedures of all effluent quality parameters were successful for both the MLPANN-GA and RBFANN-GA models. The training and testing models showed an almost perfect match between the experimental and predicted values. Based upon the statistical analysis, results indicated that there is a very little difference between predicted and experimental values of the effluent BOD, COD, TN, and TP. The predicted and experimental values of the effluent concentrations gave a very low root mean squared error and a high coefficient of determination very close to one demonstrated high accuracy of these models to predict output variables. It became clear that the models based on the genetic algorithm (GA) were much better than those models without GA from the viewpoint of the achievement of an accurate prediction of the effluent BOD, COD, TN, and TP. The results indicated that the accuracy of all models increased when GA was applied to neural networks. The mean average error for the hybrid models varied from 3 to 8%.

Keywords: Membrane bioreactor; Effluent concentrations; Multilayer perceptron; Radial basis function; Genetic algorithm

*Corresponding author.

1. Introduction

The activated sludge process (ASP) is widely used for both municipal and industrial wastewater treatment. The submerged membrane bioreactor (SMBR) is an improvement on the conventional ASP, where the traditional secondary clarifier is replaced by a membrane unit for the separation of treated wastewater from the mixed solution in the bioreactor [1,2]. The SMBR offers significant advantages, such as smaller footprint, high sludge concentration, high-quality of effluent, complete separation of the hydraulic retention time, and sludge retention time (SRT) over conventional activated sludge systems [2,3]. Moreover, the treated wastewater (effluent) is free from bacteria and has the potential for municipal and industrial reuse.

Treatment process models are essential tools to assure proper operation and better control of the wastewater treatment plants [4]. Some deterministic models have been developed based on the fundamental biokinetics, such as activated sludge model number one (ASM1) [5]. Following ASM1, ASM2, ASM2d, and ASM3 models were developed. Parameter estimation and calibration of the ASM models require expertise and significant effort. Moreover, calibration has to be performed for each specific treatment system. Therefore, application of ASM models to real systems can be cumbersome and problematic [6,7]. Understanding and optimizing a system as complex as a membrane bioreactor is difficult and time consuming [1,8]. It is composed of many subprocesses that are highly coupled. Next to the biokinetic processes for bioconversion of pollutants, the separation process takes place as well as hydrodynamic flows that develop both in the bioreactor and membrane module [8].

In recent years, artificial neural networks (ANNs) have been used for monitoring, controlling, and simulation of the ASP for the wastewater treatment plants [6]. Geissler et al. [9] used an ANN model to predict the filtration performance in a submerged capillary hollow fiber membrane treating municipal wastewater. Cinar et al. [10] have also proposed an ANN model for an SMBR treating cheese whey and evaluated its performance at different SRTs. Pendashteh et al. [11] modeled a membrane sequencing batch reactor treating hypersaline oily wastewater, which operated at different total dissolved solids (TDS), various organic loading rates, and cyclic time. The training, validating, and testing procedures for the chemical oxygen demand (COD), total organic carbon, and oil and grease concentrations were successful and a good correlation was observed between the measured and the predicted values. Badrnezhad et al. [12] performed a

precise ANN model for flux decline under various operating parameters in cross-flow ultrafiltration of oily wastewaters. The ANN model received feed temperature (T), feed pH, transmembrane pressure (TMP), cross-flow velocity (CFV), and filtration time as inputs; and gave permeate flux as an output. The results obtained validate the estimates of the ANN technique with a good accuracy. According to the result of the sensitivity analysis based on the correlation coefficient, the filtration time was the most significant one, followed by T , CFV, feed pH, and TMP.

In order to achieve the objective of this study, two types of feedforward artificial neural network (FANN) were employed, which are most commonly used in classification problems, namely multilayer perceptron (MLP) and radial basis function (RBF). They were selected because of their ability to detect complex non-linear relationships in the data, representing two different approaches to solving problems. The MLP employs hyperplanes to divide the pattern space into various classes, while RBF uses hyperspheres [13]. The RBFs have been successfully applied for solving dynamic system problems because they can predict the behavior directly from input/output data [14,15]. Many researchers have studied the MLPs and recommend them for water quality prediction measures, such as total suspended solids (TSS), biochemical oxygen demand (BOD), COD, dissolved oxygen (DO), and ammonia [10,11].

The ANN is typically used as a black-box approach, hiding the physics of the model process, and lacks for extrapolative capacity [16]. In addition, the gradient algorithm usually used in the backpropagation (BP) neural network is a local search algorithm and may tend to fall into a local minimum and results are inconsistent and in unpredictable performances [16,17]. Genetic algorithm (GA) based on the principles of survival of the fittest strategy, has been proven to be a powerful optimization method to solve problems with objective functions that are not continuous or differentiable [18]. The introduction of GA might avoid trapping into the local minimum of the ANNs. Hybrid models employ a combination of ANNs and GAs. The ANN-GA coupled models have been developed to optimize the various biological processes [12,16]. Badrnezhad et al. [12] modeled and optimized flux decline using hybrid ANN-GA under various operating parameters in the cross-flow ultrafiltration of oily wastewaters. By applying GA, optimum conditions, which lead to the highest and lowest flux value were achieved when feed pH were in the range of alkaline solutions and at about isoelectric point, respectively. Fang et al. [16] developed an integrated dynamic model through combining a mechanistic

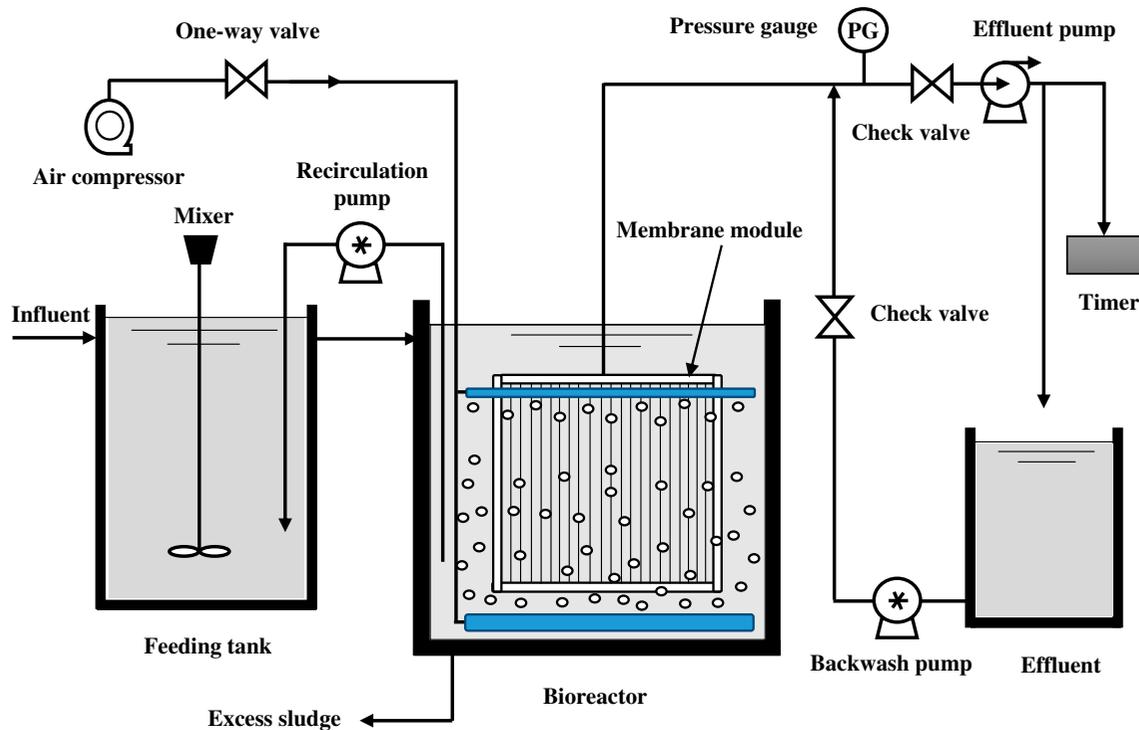


Fig. 1. Configuration of the submerged membrane bioreactor.

model, an ANN model, and a GA approach, in order to simulate the performance of a full-scale municipal wastewater treatment plant (WWTP) with substantial influent fluctuations. They concluded that compared with the mechanistic model and the ANN model, the integrated model was able to capture sufficient residual information to compensate for the inaccuracy of the mechanistic model and improve the extrapolative capability of the ANN model.

However, this hybrid model used to optimize the weights and thresholds of multilayer perceptron artificial neural networks (MLPANNs) and radial basis function artificial neural networks (RBFANNs) in the prediction of the effluent quality parameters of an SMBR treating municipal wastewater has not been reported. This research was an effort to develop hybrid MLPANN-GA and RBFANN-GA models to accurately predict effluent BOD, effluent COD, effluent total nitrogen (TN), and effluent total phosphorus (TP) in an SMBR treating municipal wastewater under various operating parameters. The operating parameters, including influent BOD, influent COD, influent TN or influent TP, SRT, mixed liquor suspended solids (MLSS), membrane permeability (Perm), and TMP were utilized in the ANN modeling processes. To the best of our knowledge, this is the first research utilizing hybrid MLPANN-GA and RBFANN-GA models to predict the effluent quality parameters for an

SMBR. Furthermore, sensitivity analyses were performed to determine the effect and importance order of each operating parameter on the effluent concentrations.

2. Materials and methods

2.1. Pilot plant configuration and operating conditions

An SMBR with simultaneous upward and downward aeration was used to treat municipal wastewater

Table 1
Specifications of the hollow fiber membrane module

Description	Value
Material	Polypropylene
Capillary thickness	40–50 μm
Capillary outer diameter	450 μm
Capillary pore diameter	0.01–0.2 μm
Gas permeation	$7.0 \times 10^{-2} \text{ cm}^3/\text{cm}^2 \text{ S cm Hg}$
Porosity	40–50%
Lengthways strength	120,000 kPa
Designed flux	6–9 $\text{L}/\text{M}^2/\text{H}$
Area of membrane module	8 m^2/module
Operating pressure	–10 to –30 kPa
Flow rate	1.0–1.2 m^3/d

in this research. Fig. 1 shows the schematic diagram of the SMBR. The pilot plant comprised a feeding tank, an aeration tank or a bioreactor, and an effluent tank. The feeding tank was made of plastic with a size of 0.8 by 0.6 m. It was located 1.5 m above the ground level to establish a continuous flow. The membrane module was placed in the aeration tank in order to achieve simultaneous aeration/filtration process. The vertically oriented hollow fiber SMBR had an aeration reactor with the size of 0.8 by 0.8 m. It was attached to two tubes with 1 cm in diameter, which conducted treated wastewater to the effluent tank. The polypropylene hollow fiber membrane had a nominal pore size of 0.04 μm and the overall membrane surface area was 8 m^2 per module. Table 1 shows the detailed specifications of the hollow fiber membrane. The experiments were totally executed for 60 d, so that they were divided into two experimental periods. The conventional aeration from the bottom of the bioreactor or upward aeration was performed for the first 30 d. The downward aeration was performed in addition to the upward aeration for the next 30 d. The air flow was continuously provided 14 min for the upward aeration and 1 min for downward aeration during the simultaneous upward and downward aeration.

2.2. Municipal wastewater characteristics

The pilot plant was located in the Ekbatan WWTP, in Tehran, Iran. Analysis of the WWTP influent was carried out for a four-month period. According to the results obtained from the raw wastewater analysis, the maximum values were selected as critical values in

Table 2

Characteristics of municipal wastewater used in this study

Parameter (unit)	Average	Maximum	Std.
T_{inf} ($^{\circ}\text{C}$)	23	25.8	2.6
DO	0	0	0
pH_{inf} ($^{\circ}\text{C}$)	8.2	8.9	0.4
BOD_{inf} (mg l^{-1})	156	175	11
COD_{inf} (mg l^{-1})	262	285	13
TN_{inf} (mg l^{-1})	29	38	2.6
$\text{NH}_4^+ - \text{N}_{\text{inf}}$ (mg l^{-1})	18	23.1	4
$\text{NO}_3^- - \text{N}_{\text{inf}}$ (mg l^{-1})	0.8	0.96	0.2
TP_{inf} (mg l^{-1})	13	16.54	2.5
TDS_{inf} (mg l^{-1})	460	630	120
TSS_{inf} (mg l^{-1})	165	180	70

Note: Std: The standard deviation of data-sets.

order to design the pilot plant. Table 2 shows the characteristics of influent wastewater used in this study.

2.3. Analytical methods

Temperature, pH, DO, BOD, COD, TN, TP, TSS, TDS, MLSS, and mixed liquor volatile suspended solids (MLVSS) were measured in this study. The pH and temperature were measured using a digital pH meter. A DO meter (YSI 5000) was utilized to determine DO. Biodegradability was measured by five-day BOD test according to the standard methods [19]. The seed for BOD₅ test was obtained from the Ekbatan WWTP. The COD was determined according to the standard methods [19]. At the Ekbatan WWTP laboratory, TN and TP were measured with aid of a

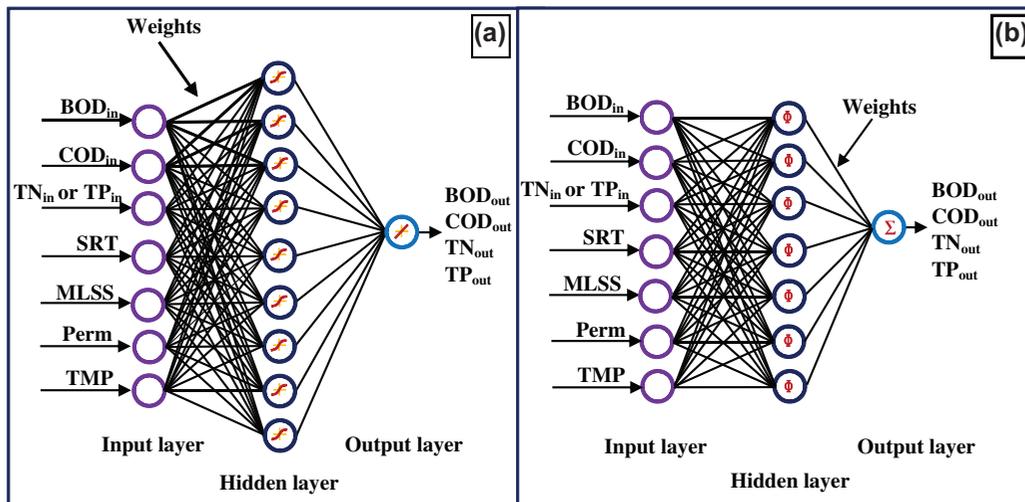


Fig. 2. Topological architectures of the neural networks: MLP (a) and RBF (b).

spectrophotometer (The Hach DR 5000 UV–vis Laboratory Spectrophotometer). Other operational parameters such as, permeate flow rate, TMP, and permeate turbidity were monitored by sensors and gages installed in the SMBR. All sensors and gages were cleaned and calibrated weekly. TMP values were normalized to a standard temperature of 20°C according to Eq. (1) [20].

$$\text{TMP} = (\text{TMP}_T) \cdot e^{0.0239 \cdot (T-20)} \quad (1)$$

where, TMP_T is the transmembrane pressure monitored at temperature T (°C), and T is the temperature of mixed liquor measured in the membrane basin.

2.4. Theory and methodology of ANN-based modeling

An ANN is composed of many single elements called neurons. An artificial neuron is a single computational processor, which has two operators (1) summing junction and (2) transfer function [21,22]. Each neuron is connected to other neurons through communication links, each with an associated weight that multiplies the signal transmitted. The weights represent information being used to solve a problem and have to be determined by a learning (training) algorithm [21]. The transfer function determines the input–output behavior and adds non-linearity and stability to the network [23]. A single neuron can be described by Eq. (2).

$$\lambda = \sum_{i=1}^n x_i \cdot w_i + b \quad (2)$$

where w_i ($i = 1, n$) is the connection weights, x_i is the input variable, n is the number of input variables, i is the integer index, and b is called bias.

The summing junction operator of a single neuron summarizes the weights and bias into a net input λ known as argument to be processed. The bias b is much like a weight, except that it has a constant input of unity. The transfer function takes the argument λ and produces the scalar output of a single neuron [23].

An MLPANN contains neurons structured in parallel layers, from inputs to outputs, as illustrated in Fig. 2. The input nodes receive the data values and pass them on to the first hidden layer nodes. Each one collects the input from all input nodes after multiplying each input value by a weight, attaches a bias to this sum, and passes on the results through a non-linear transformation [24]. This forms the input either for the second hidden layer or the output layer that operates identically to the hidden layer. The resulting transformed output from each output node is the network output [24,25]. The network needs to be trained using a training algorithm, such as Levenberg–Marquardt (LM) algorithm, incremental backpropagation, gradient descent backpropagation (GDB), gradient descent with adaptive learning rate backpropagation, and batch backpropagation. Basically, the objective of training patterns is to reduce the global error.

Fig. 2 shows that the structure of the basic RBFANN contains one input layer, one output layer, and one hidden layer. The transformation from the input nodes to the hidden nodes is a non-linear one, and training of this portion of the network is generally accomplished by an unsupervised fashion. The training of the network weights between the hidden and output layers occurs in a supervised fashion based on target outputs [24]. The hidden layer neurons in the middle of the basis function only have local reactions of input function for the RBFANN.

The performances of the ANN models are measured by coefficient of determination (R^2) and root mean squared error (RMSE) between the predicted values of

Table 3
Characteristics of measured variables used in modeling using the MLPANN-GA and RBFANN-GA

Input variable number	Input variable	Range	Avg.	Std.	Output variable	Range	Avg.	Std.
1	BOD _{in} (mg l ⁻¹)	130–175	156	11	BOD _{out} (mg l ⁻¹)	3–17	6.3	3.5
2	COD _{in} (mg l ⁻¹)	230–285	262	13	COD _{out} (mg l ⁻¹)	7–29	11.6	5.3
3	TN _{in} (mg l ⁻¹)	25–38	29	2.6	TN _{out} (mg l ⁻¹)	1.4–3	1.9	0.4
	TP _{in} (mg l ⁻¹)	5–16.54	13	2.5	TP _{out} (mg l ⁻¹)	2–6.5	3.2	1.3
4	SRT (d)	30–50	40	8				
5	MLSS (mg l ⁻¹)	3,800–5,400	4,578	477				
6	Perm (LMH/kPa)	1.02–1.65	1.3	0.18				
7	TMP (kPa)	14.3–21.5	18.6	1.9				

Note: Avg: The average of data-sets and Std: The standard deviation of data-sets.

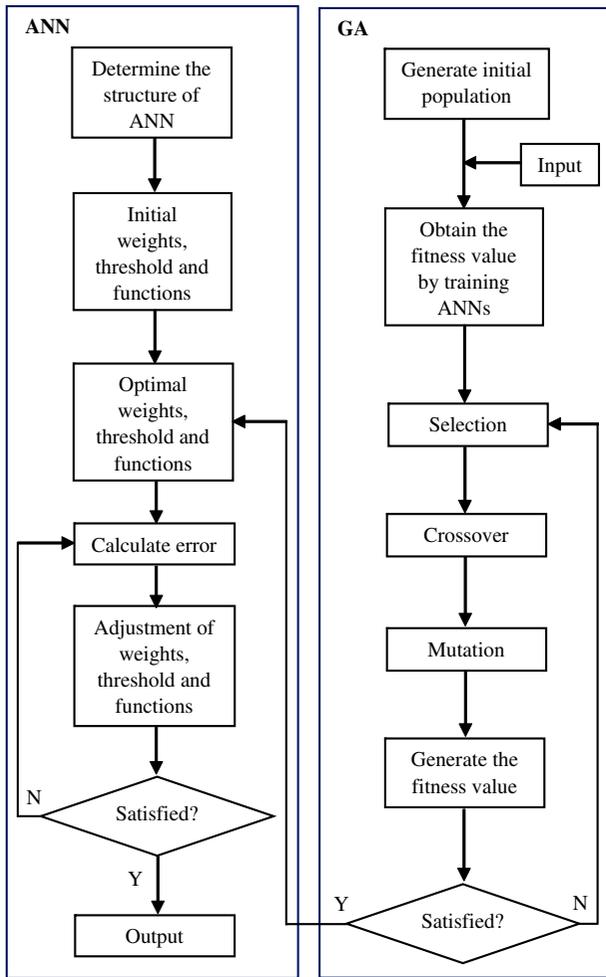


Fig. 3. Flowchart of ANN-GA hybrid methodology in the optimization process.

the network and the experimental values, which are calculated by Eqs. (3) and (4), respectively [11].

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i^* - y_p^{(i)})^2}{\sum_{i=1}^n (y_i^* - \bar{y})^2} \quad (3)$$

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

where \bar{y} is the average of y over the n data, and y_i^* and $y_p^{(i)}$ are the i th target and predicted responses, respectively.

2.5. Theory and methodology of GA-based optimization

For the first time, the concept of GA was developed by Holland at the University of Michigan [26].

GA has three key genetic operators: (1) selection, (2) crossover, and (3) mutation. Algorithm is started with a set of random solutions called population. Solutions from one population are used to form a new population. This is motivated by a hope that the new population will be better than an old population [26,27]. Selection is the survival of the fittest, which means the highest quality chromosomes will stay within the population. In the selection process, the solutions are selected according to their values of objective function (fitness). The best solution is returned to represent the optimum solution [27]. There are different selection methods as stochastic uniform, remainder, uniform, shift linear, roulette, and tournament. The tournament method can be described by Eq. (5).

$$\tau_i = F_i / \sum_{j=1}^{N_k} F_j \quad (5)$$

where τ_i is the weight of i th individual within population. Moreover, the sum of the elective probabilities of all the individuals within population is 1 as it is determined by Eq. (6).

$$\sum_{i=1}^{N_k} \tau_i = 1 \quad (6)$$

Crossover is applied on two individuals, called parents, and originates two new individuals called sons, which contain the combined traits of the parents [28]. There are different crossover methods as one-point, two-point, and uniform crossover. The most straightforward approach is one-point crossover, where those parts of the strings that exceed a randomly selected point are simply swapped. The underlying idea is that the strings with a high fitness contain building blocks of valuable genetic information [29]. Two-point crossover calls for two points to be selected on the parent organism strings. Everything between the two points is swapped between the parent organisms, rendering two child organisms. The uniform crossover uses a fixed mixing ratio between two parents. Unlike one- and two-point crossover, the uniform crossover enables the parent chromosomes to contribute the gene level rather than the segment level [30].

In general, mutation operator specifies how the GA makes small random changes in the individuals in the population to create mutation children. Mutation provides genetic diversity and enables the GA to search a broader space [31]. In binary encoding, mutation can be achieved simply by flipping between 0 and 1.

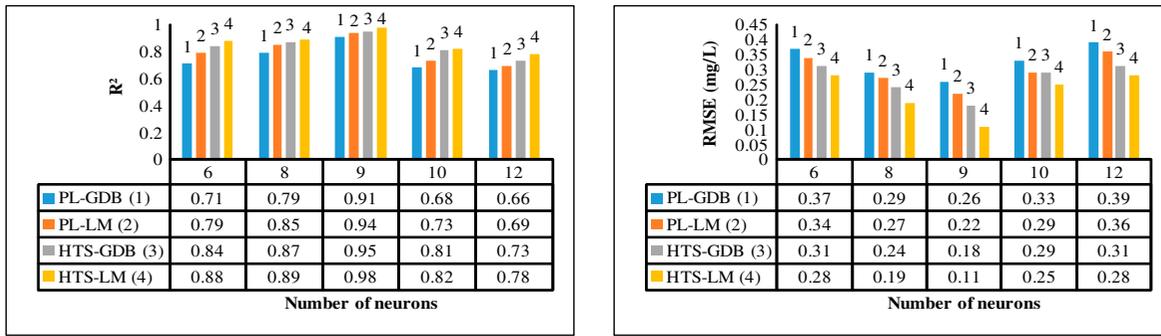


Fig. 4. Comparison of GDB and LM algorithms with HTS and pure linear function (PL), as measured by number of neurons during validation phase.

Mutation can occur at a single site or multiple sites simultaneously [29]. There are different mutation methods as random (uniform), non-uniform, and Gaussian. In uniform mutation, a gene is replaced with a random value between its lower and upper bounds. On the other hand, in non-uniform mutation, the step size decreases as the generations increase, thus making a uniform search in the initial space and very little at the later stage. In Gaussian mutation

operator, two parameters: the mean (usually set to zero), and the standard deviation of the Gaussian distribution are required [29,32].

2.6. Pre-modeling analyses

In order to obtain convergence within a reasonable number of cycles, the input and output data should

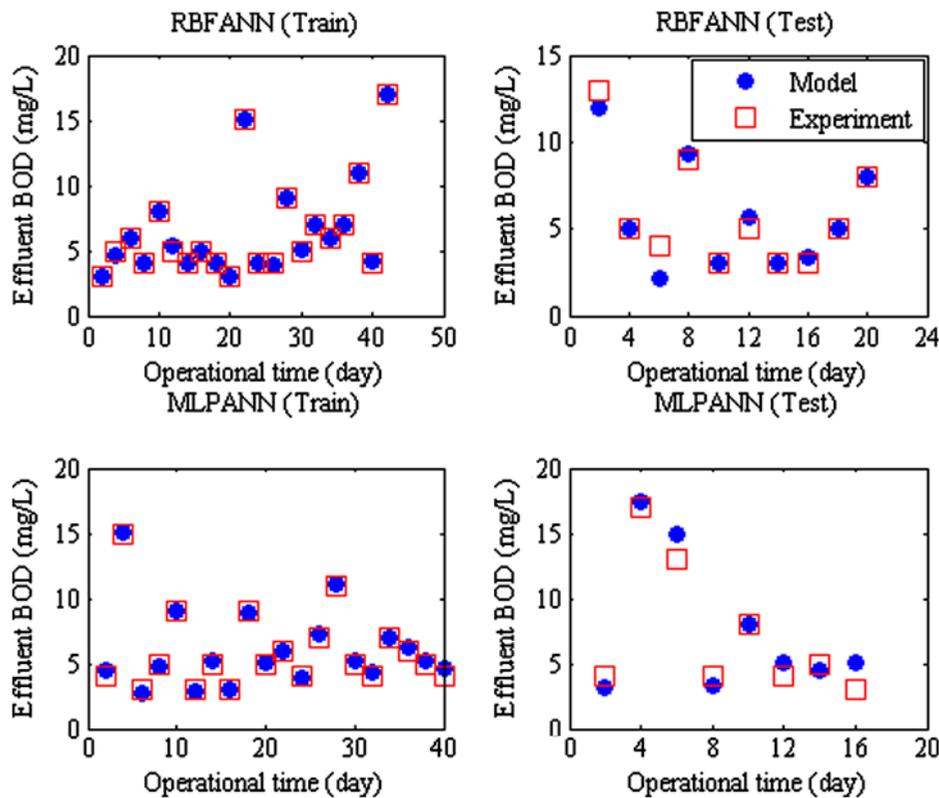


Fig. 5. Effluent BOD models by RBFANN-GA and MLPANN-GA according to train and test data-sets.

be normalized and scaled to the range of 0–1 by Eq. (7) [33]:

$$x_{ni} = (x_i - x_{\min}) / (x_{\max} - x_{\min}) \quad (7)$$

where x_i is the initial value, x_{\max} and x_{\min} are the maximum and minimum of the initial values, and x_{ni} is the scaled value. After the training and testing of the ANN, the output data were scaled to the real-world values through Eq. (8).

$$x_i = x_{ni}(x_{\max} - x_{\min}) + x_{\min} \quad (8)$$

Simulation models of operational parameters were established based on the theory of FANN, namely RBFANN and MLPANN using the mathematical software program MATLAB. Experimental data over 60 d were used in artificial neural network modeling. The statistical characteristics of the measured variables have been presented in Table 3. The flowchart diagram of ANN–GA hybridization used to predict effluent BOD, COD, TN, and TP is shown in Fig. 3.

3. Results and discussion

3.1. Optimal architectures of the neural networks

In order to predict effluent BOD, COD, TN, and TP by RBFANN-GA and MLPANN-GA, influent BOD, COD, TN or TP as well as SRT, MLSS, membrane Perm, and TMP were used as inputs of the neural networks. Each network structure was selected after running a number of preliminary experiments to explore the training speed and response time of different architectures. To keep the network structure as simple as possible, three layers were used in all networks. The optimal architectures insure training with reasonable speed and short simulation time for a specific network performance. The RBFANN regularization network employs the same number of neurons as the input data points. The number of neurons of the MLPANNs was kept equal to the number of training exemplars for better comparison of both ANN performances. A two-stage training process was applied for the RBFANNs. The K-means to assign the radial centers in the dataset and K-nearest neighbors to compute the deviation of each center were used in the first stage. The output layer was optimized with pseudo-inverse method in the second stage. To determine the best network

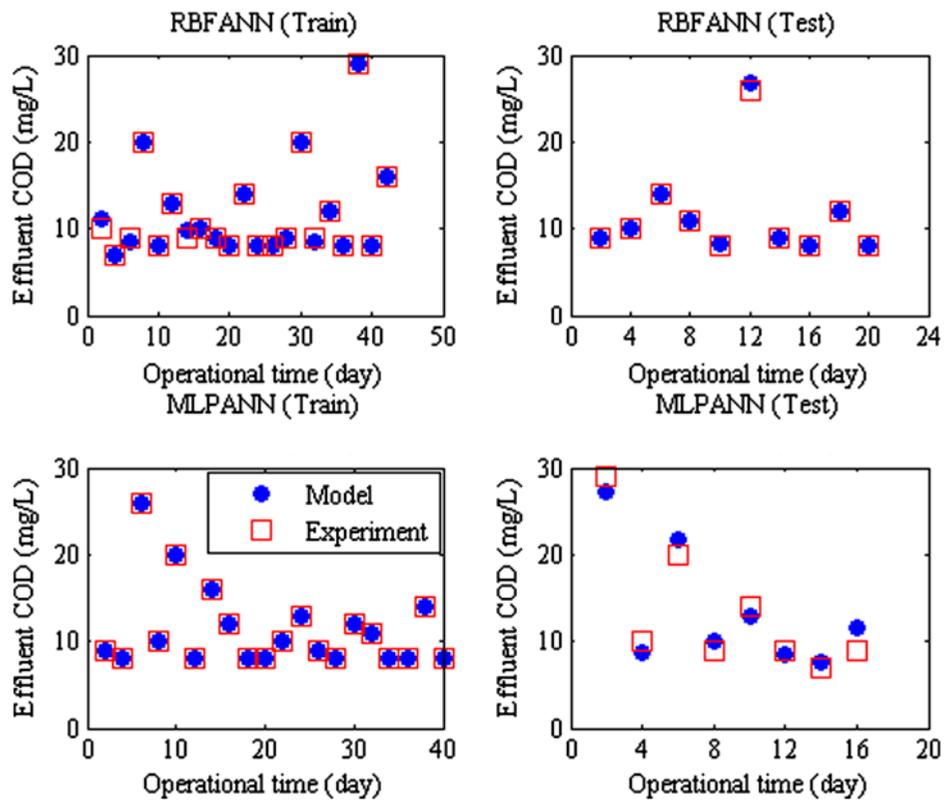


Fig. 6. Effluent COD models by RBFANN-GA and MLPANN-GA according to train and test data-sets.

Table 4

Effect of single and joint variables on the effluent BOD models by RBFANN-GA and the MLPANN-GA

Input variable No.	BOD models (RBFANN-GA)				BOD models (MLPANN-GA)				Importance order
	R^2		RMSE (mg l ⁻¹)		R^2		RMSE (mg l ⁻¹)		
	Train	Test	Train	Test	Train	Test	Train	Test	
1	0.331	0.524	3.01	1.51	0.315	0.601	3.14	2.34	4
2	0.399	0.604	2.54	1.14	0.398	0.567	2.69	1.26	3
3	0.334	0.345	2.66	3.98	0.327	0.301	2.86	4.97	7
4	0.355	0.462	2.62	3.58	0.337	0.429	3.01	4.77	6
5	0.667	0.476	2.39	3.51	0.658	0.461	2.41	4.22	5
6	0.684	0.866	1.61	2.05	0.659	0.856	1.68	2.11	1
7	0.552	0.853	2.46	2.07	0.557	0.842	2.55	2.11	2
6-1	0.54	0.63	1.99	2.65	0.49	0.55	2.48	3.15	3
6-2	0.58	0.61	1.97	2.65	0.52	0.58	2.44	3.12	2
6-3	0.32	0.31	4.12	4.19	0.31	0.31	4.13	4.24	6
6-4	0.42	0.34	3.45	3.68	0.39	0.33	3.72	3.75	5
6-5	0.47	0.45	3.04	3.11	0.45	0.46	3.25	3.09	4
6-7	0.77	0.78	1.62	2.14	0.65	0.76	1.95	1.88	1
6-7-1	0.71	0.72	1.79	1.88	0.71	0.69	1.81	1.91	3
6-7-2	0.74	0.73	1.73	1.79	0.71	0.68	1.75	1.86	2
6-7-3	0.51	0.54	2.04	2.44	0.51	0.53	2.13	2.63	5
6-7-4	0.65	0.72	1.86	1.89	0.59	0.63	1.95	2.23	4
6-7-5	0.75	0.85	1.61	1.53	0.77	0.76	1.59	1.84	1
6-7-5-1	0.81	0.81	1.29	1.26	0.78	0.76	1.31	1.44	2
6-7-5-2	0.84	0.84	1.17	1.18	0.82	0.79	1.21	1.24	1
6-7-5-3	0.69	0.68	1.72	1.54	0.68	0.65	1.79	1.82	4
6-7-5-4	0.78	0.69	1.58	1.52	0.74	0.68	1.66	1.69	3
6-7-5-2-1	0.91	0.93	0.65	0.53	0.92	0.92	0.63	0.67	1
6-7-5-2-3	0.89	0.87	0.74	0.81	0.88	0.86	0.77	0.83	3
6-7-5-2-4	0.91	0.91	0.68	0.59	0.91	0.89	0.69	0.72	2
6-7-5-2-1-3	0.92	0.94	0.51	0.49	0.92	0.92	0.57	0.64	2
6-7-5-2-1-4	0.97	0.96	0.42	0.49	0.95	0.93	0.55	0.59	1
6-7-5-2-1-4-3	0.98	0.99	0.41	0.44	0.97	0.97	0.51	0.52	1

Note: The numbers 1–7 refers to input variables identified in Table 3.

function, various algorithms were studied. The RBFANN-GA applied the newrbe function to the input data as the optimal network function. The newrbe function created a two-layer network with biases for both layers. The first layer followed a radial basis transfer function. Consequently, its weighted inputs were calculated with the Euclidean distance weight function and its net input with product net input function. The second layer followed a linear transfer function. Consequently, its weighted inputs were calculated with dot product weight function and its net input with sum net input function. The spread of radial basis function was equal to its default value, 1. A large spread value results in a smooth function approximation, but by contrast, a small spread value can result in numerical problems [13,34]. The newrbe function selected 75% of normalized data to train

and 25% to test the RBFANN-GA models. The RBFANN-GA was designed in a loop that applied the newrbe function to the data for less than 50 times in order to minimize error. The optimum network in the modeling process was chosen on the basis of the minimum average error.

The MLPANN-GA applied the newff function to the input data as the optimal network function. Therefore, it created a feedforward backpropagation neural network. The first layer of the network had weights coming from the inputs and each subsequent layer had a weight coming from the previous layer. The transfer function of this network was a differentiable transfer function. The LM was the default training function because it was very fast, but it required a lot of memory to run. The newff function selected 60% of normalized data to train, 20% to test, and 20% to

Table 5
Effect of single and joint variables on the effluent COD models by RBFANN-GA and the MLPANN-GA

Input variable No.	COD models (RBFANN-GA)				COD models (MLPANN-GA)				Importance order
	R^2		RMSE (mg l ⁻¹)		R^2		RMSE (mg l ⁻¹)		
	Train	Test	Train	Test	Train	Test	Train	Test	
1	0.586	0.533	4.97	5.48	0.554	0.514	5.26	5.37	4
2	0.611	0.518	4.77	5.12	0.577	0.523	4.82	4.95	3
3	0.325	0.325	6.95	6.87	0.322	0.317	7.05	7.12	7
4	0.411	0.348	6.49	6.22	0.409	0.313	6.55	6.75	6
5	0.572	0.519	5.12	5.85	0.555	0.501	5.48	5.99	5
6	0.695	0.875	3.14	2.95	0.686	0.817	3.68	3.13	1
7	0.612	0.788	3.87	3.49	0.584	0.741	4.11	4.05	2
6-1	0.661	0.675	3.25	3.83	0.662	0.651	3.57	3.94	3
6-2	0.707	0.721	3.18	3.42	0.695	0.687	3.35	3.58	2
6-3	0.528	0.475	4.83	5.37	0.514	0.388	4.94	5.55	6
6-4	0.543	0.572	4.79	4.63	0.548	0.531	4.68	4.97	5
6-5	0.613	0.594	4.12	4.55	0.599	0.586	4.43	4.89	4
6-7	0.711	0.789	2.99	2.86	0.699	0.714	3.02	2.96	1
6-7-1	0.701	0.601	2.92	3.42	0.688	0.587	3.25	3.79	3
6-7-2	0.703	0.608	2.87	3.35	0.674	0.604	2.93	3.52	2
6-7-3	0.512	0.527	4.04	3.93	0.496	0.486	4.44	4.62	5
6-7-4	0.675	0.569	3.44	3.71	0.619	0.585	3.68	4.19	4
6-7-5	0.699	0.815	2.68	2.75	0.701	0.648	2.67	3.11	1
6-7-5-1	0.807	0.814	2.33	2.17	0.798	0.802	2.47	2.59	2
6-7-5-2	0.812	0.826	2.24	2.12	0.811	0.813	2.43	2.37	1
6-7-5-3	0.623	0.622	2.85	3.11	0.622	0.617	3.15	3.23	4
6-7-5-4	0.744	0.726	2.61	2.28	0.717	0.703	2.66	2.82	3
6-7-5-2-1	0.915	0.927	1.76	1.73	0.894	0.899	1.85	1.89	1
6-7-5-2-3	0.785	0.774	2.34	2.41	0.712	0.771	3.41	2.36	3
6-7-5-2-4	0.871	0.862	1.89	2.18	0.852	0.826	2.11	2.37	2
6-7-5-2-1-3	0.945	0.932	1.76	1.95	0.933	0.929	1.84	2.01	2
6-7-5-2-1-4	0.975	0.968	1.64	1.72	0.966	0.965	1.72	1.75	1
6-7-5-2-1-4-3	0.996	0.998	1.12	1.12	0.989	0.993	1.21	1.19	1

Note: The numbers 1–7 refers to input variables identified in Table 3.

validate the MLPANN-GA models. To train the MLPANN, we need to choose a proper structure for the ANN along with suitable activation functions for its neurons. To determine the best BP training algorithm, various BP algorithms were studied. Hyperbolic tangent sigmoid transfer function (tansig) at hidden layer and linear transfer function (purelin) at output layer were found to be the optimal functions. In addition, four neurons were used in the hidden layer as initial value for all BP algorithms. The MLPANN-GA was trained by different learning algorithms for a maximum of 250 epochs. Nevertheless, the LM algorithm resulted in the optimum models for train and test data after less than 20 iterations. The LM had smaller RMSE values compared to other backpropagation algorithms. So, the LM was considered the training algorithm in this research. The predictive accuracy of networks in the ANN models depend on the

number of hidden neurons, learning functions, and learning rate [35], so these variables were chosen to optimize the ANN structure by the GA program. The optimum number of neurons in the hidden layer was determined based on the minimum value of RMSE for the training and prediction set. Based on the result of this study, the optimum models for the prediction of the effluent BOD, COD, TN, and TP are obtained with the hidden layer consisting of nine neurons.

The results of two training algorithms, including the LM and the GDB were compared in order to determine the effect of GA on the MLPANN-GA models. Furthermore, two transfer functions, including hyperbolic tangent sigmoid (HTS) function and pure linear (PL) function were also used to examine the effect of GA on the MLPANN-GA models. The statistical criteria of RMSE and R^2 for each structure in validation phase of MLPANN modeling are given in

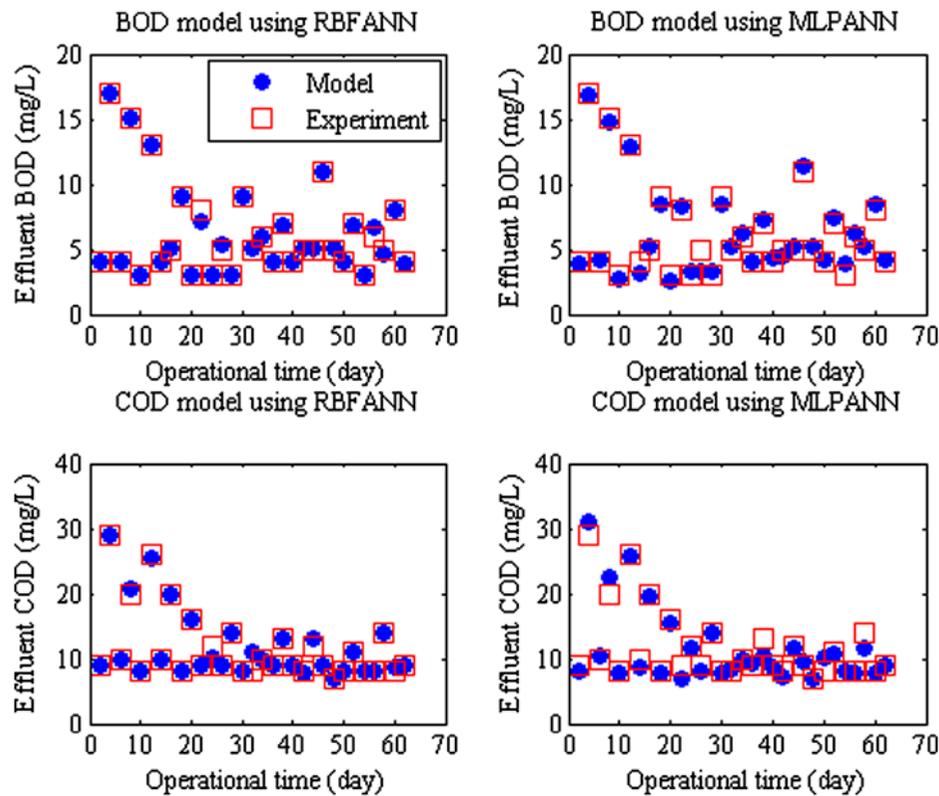


Fig. 7. Effluent BOD and COD models by RBFANN-GA and MLPANN-GA according to all data-set.

Fig. 4. The RMSE of HTS transfer function and nine neurons with LM learning algorithm is the lowest value. Moreover, Fig. 4 shows that the coefficient of determination value of this structure is low. Comparing the RMSE and R^2 values for all neural network structures, the optimized structure was found to be with nine neurons in hidden layer, and with LM algorithm and HTS function. Because gradient decent usually slows down near minima, so the LM method can be used to obtain faster convergence. LM is a blend of simple gradient descent and the Gauss–Newton method [11]. The LM has found to be the fastest method for training moderate-sized feedforward neural networks, where the training rate is 10 to 100 times faster than the usual GDB method [36]. The HTS function was selected for the hidden neurons due to its better prediction performance than other transfer functions among various transfer functions available in the Matlab. The HTS function was bounded between 0 and 1, so the input and the output data were normalized to the same range as the transfer function used. The optimal structure had the highest R^2 value equal to 0.98 and the least RMSE value equal to 0.11 mg^{-1} for the denormalized data.

3.2. BOD and COD models by RBFANN-GA and MLPANN-GA

The selected neural networks were used to predict the effluent BOD and COD for different inputs in the domain of training, testing, and all data-sets. The effluent BOD and COD values for the train and test models by RBFANN-GA and MLPANN-GA are plotted against operational time in Figs. 5 and 6, respectively. As can be seen, the generalization performances of the three-layer ANNs show no oscillation. The results confirm excellent prediction performance of the ANNs based on the training and testing models for the effluent BOD and COD. The training procedures in the prediction of the effluent BOD and COD were successful for both RBFANN-GA and MLPANN-GA models. The train and test models by RBFANN-GA and MLPANN-GA showed an almost perfect match between the experimental and the predicted values of the effluent BOD and COD. The results of the effluent BOD and COD modeling using the RBFANN-GA and the MLPANN-GA for the training and testing data were in a good agreement with the results of previous researches [7,10].

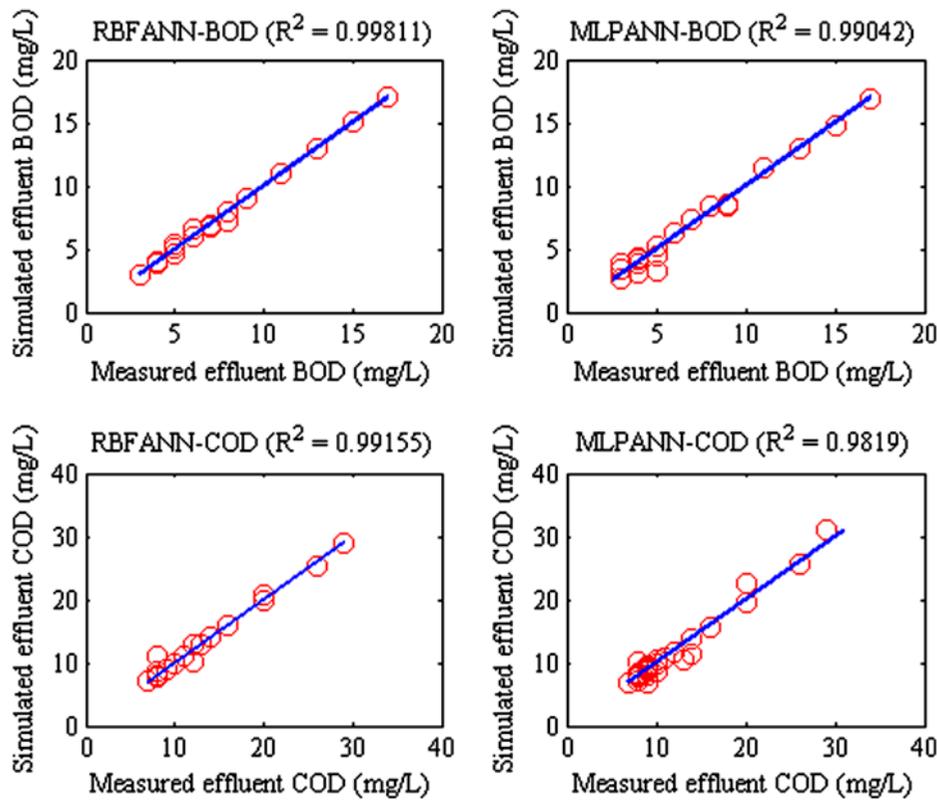


Fig. 8. Regression plots of RBFANN-GA and MLPANN-GA for the prediction of effluent BOD and COD.

The implementation of the modeling started with the determination of the variable combinations exhibiting better results in the ANN model output [6]. For this purpose, in this research, the effluent BOD and COD were modeled separately by considering various single and joint variables as inputs of RBFANN-GA and MLPANN-GA in order to examine the effect of each variable on the effluent BOD and COD values. The joint inputs were used in groups of two, three, four, five, and six variables. Analysis of R^2 and RMSE values showed that Perm among single input variables, and Perm and TMP among groups of two variables significantly affected the variation of the effluent BOD and COD models. The most important variables among groups of three variables were Perm, TMP, and MLVSS, and the noteworthy variables among groups of four variables were Perm, TMP, MLVSS, and influent COD. In addition, Perm, TMP, MLVSS, influent COD, and influent BOD among groups of five variables, and Perm, TMP, MLVSS, influent COD, influent BOD, and SRT between groups of six variables majorly affected the effluent BOD and COD models (Tables 4 and 5). The results showed high collaboration for the RBFANN-GA and MLPANN-GA

models with single and joint input variables. The RBFANN-GA modeled effluent BOD and COD was more accurate compared with the MLPANN-GA due to higher R^2 and lower RMSE values. The simulated values of the effluent BOD and COD using the RBFANN-GA models did not fluctuate noticeably for single input and joint input variables. On the other hand, the RBFANN-GA models indicated more uniform results comparing with the MLPANN-GA models. It was observed that both RBFANN-GA and MLPANN-GA show more accurate and uniform models with an increase in the number and correlation of input data-sets.

Sensitivity analyses [37] were performed to examine the sensitivity of the effluent BOD and COD to changes of input variables. The effect of a variable on the RBFANN-GA and MLPANN-GA models compared with the other variables was determined by its importance order. Tables 4 and 5 show the importance order of each input variable and the joint variables for the prediction of the effluent BOD and COD. The variable with higher rank of importance indicted less RMSE and more R^2 values for the obtained models as well as a good fitting between experimental and

predicted values of the effluent BOD and COD. The sensitivity analyses of input variables for the effluent BOD and COD models by RBFANN-GA and MLPANN-GA indicated that the effluent BOD and COD are influenced by Perm, TMP, MLVSS, influent COD, and influent BOD, SRT, and influent TN, respectively. The values of sensitivity to Perm, TMP, MLVSS, influent COD, influent BOD, SRT, and influent TN for effluent BOD were 68.1, 66.3, 54.2, 32.5, 24.9, 12.4, and

6.4%, respectively. The values of sensitivity to mentioned variables for effluent COD were 69.7, 67.5, 55.1, 31.9, 24.7, 13.5, and 7.8%, respectively. This study shows that the Perm and TMP as well as MLVSS significantly affect the effluent BOD and COD models.

Fig. 7 shows the effluent BOD and COD models by RBFANN-GA and MLPANN-GA according to all experimental data-set. The prediction of the effluent BOD and COD were successful for both the

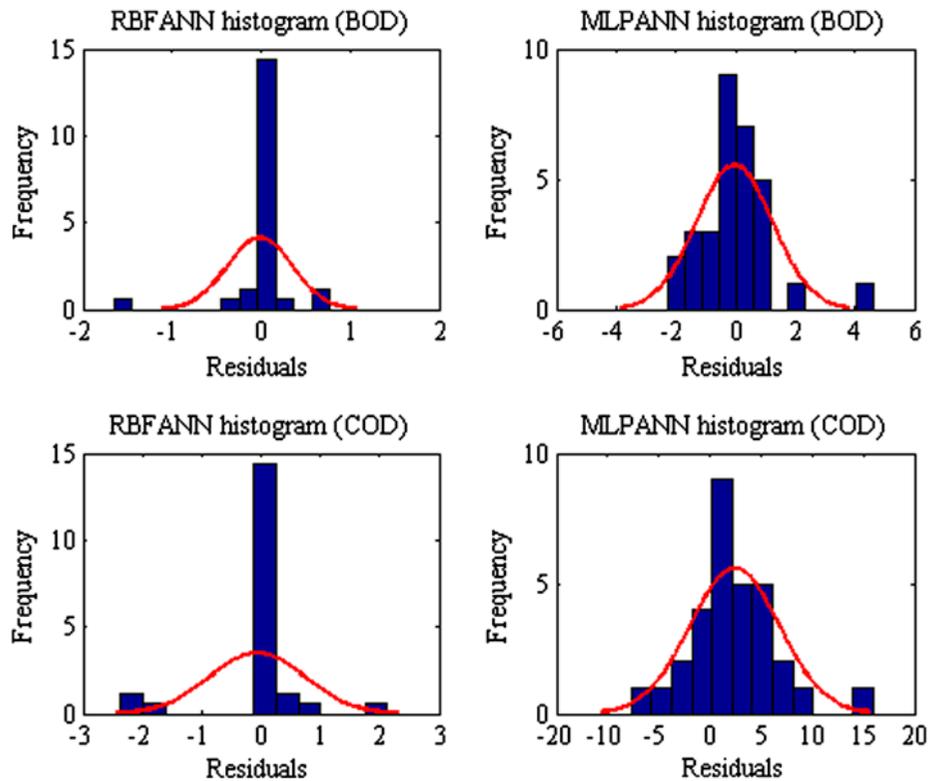


Fig. 9. Residuals of RBFANN-GA and MLPANN-GA for the prediction of effluent BOD and COD.

Table 6

Comparison of RBFANN and MLPANN with and without applying GA for the BOD and COD models

Model	R^2			RMSE (mg l^{-1})			Percentage of error (%)	Rank of model (accuracy)
	All	Train	Test	All	Train	Test		
Biochemical oxygen demand (BOD) models								
RBFANN	0.921	0.911	0.914	0.59	0.61	0.61	10	3
RBFANN-GA	0.998	0.980	0.990	0.35	0.41	0.44	5	1
MLPANN	0.872	0.888	0.875	0.79	0.68	0.71	13	4
MLPANN-GA	0.990	0.970	0.970	0.42	0.51	0.53	9	2
Chemical oxygen demand (COD) models								
RBFANN	0.904	0.932	0.938	1.68	1.27	1.23	15	3
RBFANN-GA	0.991	0.996	0.998	1.15	1.12	1.12	8	1
MLPANN	0.885	0.843	0.827	2.21	2.28	2.32	18	4
MLPANN-GA	0.981	0.989	0.993	1.24	1.21	1.19	10	2

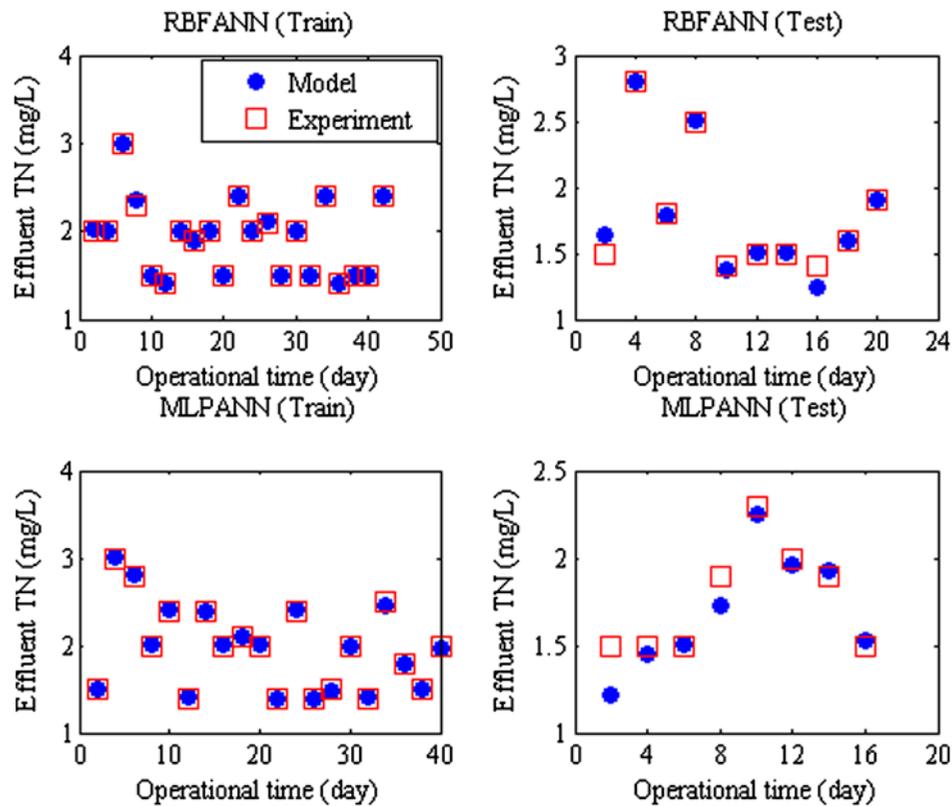


Fig. 10. Effluent TN models by RBFANN-GA and MLPANN-GA according to train and test data-sets.

RBFANN-GA and the MLPANN-GA models. Both models showed an almost perfect match between the experimental and the predicted values of the effluent BOD and COD. This study indicates that the RBFANN-GA has stronger approximation and generalization ability than the MLPANN-GA with regard to our models for the effluent BOD and COD.

The verification of an ANN model is substantiating that the model is transformed from one form into another with sufficient accuracy. The ANN models were verified by evaluating their performance in accurately predicting the statistical features of the observed data. The autocorrelation functions of the predicted values were compared with the measured values. Another criterion applied for judging the validity of the models is the assessment of the goodness of fit according to various available criteria. Fig. 8 shows the regression lines for all data-sets based on the RBFANN-GA and MLPANN-GA models for the effluent BOD and COD values. The values of R^2 for BOD models by RBFANN-GA and MLPANN-GA were 0.998 and 0.990, respectively, compared with the findings of previous studies [7,10], where R^2 varied from 0.85 to 0.91. The values of R^2 for COD

models were 0.991 and 0.981, respectively, compared with the findings of previous studies [7,38], where R^2 varied from 0.82 to 0.92.

Additionally, a way to measure the predictive capability of a model is to test it on a set of data not used in the simulation process [39]. This has been described in literature as test set and the data used for simulation is training set. In order to verify our models, a set of the effluent BOD and COD data was used to investigate the predictive ability of the models. The RMSE values for the train and test (verification) models by RBFANN-GA for BOD were 0.41 and 0.44 mg^{-1} , respectively. The RMSE values for the train and test models by MLPANN-GA were 0.51 and 0.52 mg^{-1} , respectively. The RMSE values for both train and test (verification) models by RBFANN-GA for COD were 1.12 mg^{-1} . The RMSE values for the train and test models by MLPANN-GA were 1.21 and 1.19 mg^{-1} , respectively. The mean average error for the prediction of effluent BOD by RBFANN-GA and MLPANN-GA were 5 and 9% of input values, respectively. The mean average error for the prediction of effluent COD by RBFANN-GA and MLPANN-GA were 8 and 10%, respectively.

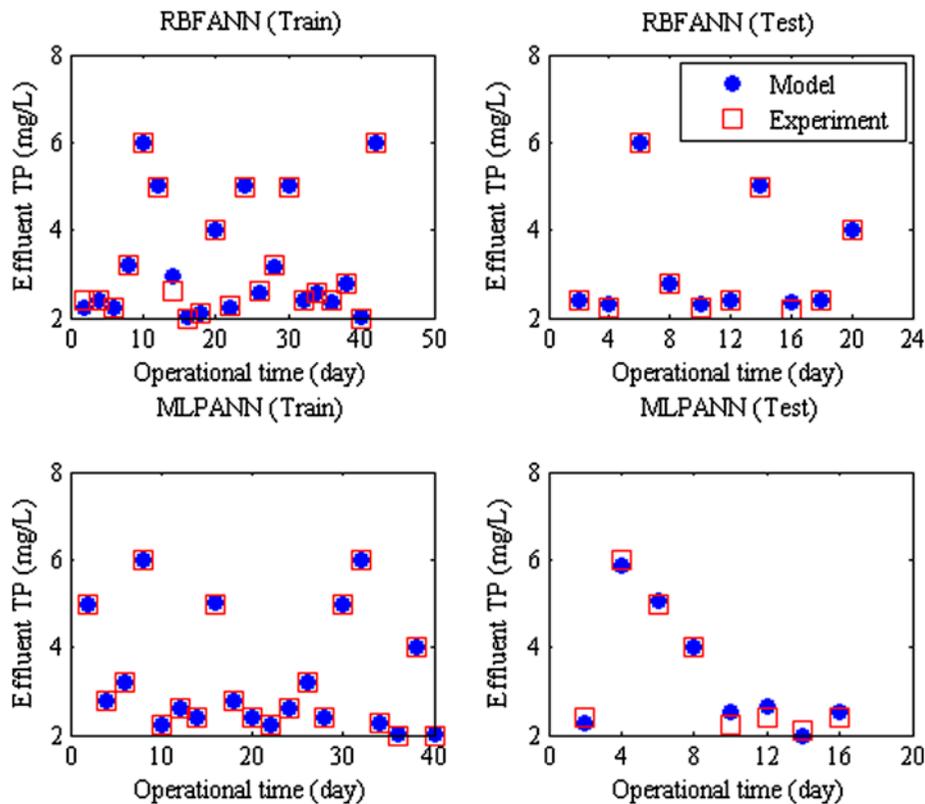


Fig. 11. Effluent TP models by RBFANN-GA and MLPANN-GA according to train and test data-sets.

The residuals of the effluent BOD and COD models attained by the RBFANN-GA and the MLPANN-GA for all data-set have been plotted out against the frequency of data in Fig. 9. A normal distribution of variation results in a Gaussian curve, with the highest point in the middle and smoothly curving symmetrical slopes on both sides of center. The results of this study illustrate an approximately normal distribution of residuals produced by the RBFANN-GA and the MLPANN-GA models. The Gaussian curve demonstrates that our results are symmetrical and their axis round around zero for all data-sets. In these hybrid ANN-GA models, GA was used to optimize the weights and thresholds of ANNs for minimizing the error between the actual and target outputs. The ANN models in this study were optimized with GA because GA is good at effectively searching large and complex spaces to find nearly global optima. GA indicates an increasingly attractive alternative to gradient-based techniques, such as RBFANN and MLPANN as the complexity of the search space increases [40]. The effects of applying GA on the RBFANN and the MLPANN in the prediction of effluent BOD and COD have been presented in Table 6. The results showed

that the precision and accuracy of all effluent BOD and COD models increased when GA was applied to the ANN models. The results of RMSE and R^2 for the effluent BOD and COD models indicated that RBFANN-GA is the most precise model with a mean average error from 5 to 8%.

3.3. TN and TP models by RBFANN-GA and MLPANN-GA

The results of effluent TN and effluent TP modeling using the RBFANN-GA and MLPANN-GA for the train and test models have been plotted versus operational time in Figs. 10 and 11. The prediction performance of the RBFANN-GA and MLPANN-GA models show no oscillation for the effluent TN and effluent TP. The results showed excellent prediction performance of the ANNs for the models according to the train and test data-sets. For both RBFANN-GA and MLPANN-GA models predicting the effluent TN and effluent TP, the training procedures were successful. The train and test models showed an almost perfect match between the experimental and the predicted values of effluent TN and effluent TP.

Table 7

Effect of single and joint variables on the effluent TN models by RBFANN-GA and the MLPANN-GA

Input variable No.	TN models (RBFANN-GA)				TN models (MLPANN-GA)				Importance order
	R^2		RMSE (mg l ⁻¹)		R^2		RMSE (mg l ⁻¹)		
	Train	Test	Train	Test	Train	Test	Train	Test	
1	0.399	0.455	0.332	0.697	0.393	0.434	0.345	0.751	7
2	0.385	0.483	0.277	0.588	0.381	0.476	0.275	0.653	6
3	0.388	0.777	0.305	0.218	0.372	0.761	0.318	0.221	5
4	0.449	0.751	0.248	0.201	0.437	0.742	0.255	0.212	4
5	0.485	0.742	0.211	0.185	0.483	0.733	0.215	0.204	3
6	0.682	0.792	0.147	0.132	0.633	0.765	0.153	0.139	1
7	0.622	0.781	0.153	0.138	0.615	0.764	0.158	0.142	2
6-1	0.321	0.301	0.429	0.445	0.302	0.282	0.405	0.426	6
6-2	0.324	0.315	0.421	0.438	0.311	0.313	0.421	0.459	5
6-3	0.518	0.541	0.357	0.339	0.489	0.512	0.366	0.357	4
6-4	0.615	0.587	0.214	0.237	0.606	0.532	0.226	0.248	3
6-5	0.675	0.699	0.172	0.164	0.666	0.671	0.178	0.169	2
6-7	0.685	0.756	0.144	0.141	0.657	0.694	0.159	0.151	1
6-7-1	0.686	0.692	0.181	0.176	0.669	0.654	0.186	0.189	5
6-7-2	0.699	0.701	0.178	0.171	0.674	0.668	0.182	0.179	4
6-7-3	0.726	0.743	0.152	0.161	0.712	0.724	0.166	0.163	2
6-7-4	0.725	0.744	0.152	0.169	0.708	0.722	0.169	0.168	3
6-7-5	0.745	0.762	0.138	0.135	0.744	0.749	0.147	0.142	1
6-7-5-1	0.803	0.785	0.144	0.158	0.783	0.758	0.152	0.162	4
6-7-5-2	0.817	0.788	0.142	0.154	0.793	0.767	0.149	0.163	3
6-7-5-3	0.862	0.837	0.124	0.127	0.855	0.829	0.131	0.139	1
6-7-5-4	0.861	0.835	0.122	0.132	0.848	0.819	0.137	0.146	2
6-7-5-3-1	0.849	0.892	0.135	0.129	0.827	0.864	0.135	0.139	3
6-7-5-3-2	0.888	0.901	0.127	0.122	0.889	0.886	0.131	0.134	2
6-7-5-3-4	0.921	0.932	0.114	0.111	0.919	0.925	0.125	0.119	1
6-7-5-3-4-1	0.966	0.975	0.109	0.105	0.942	0.961	0.121	0.117	2
6-7-5-3-4-2	0.968	0.976	0.101	0.098	0.953	0.971	0.108	0.103	1
6-7-5-3-4-2-1	0.992	0.994	0.081	0.074	0.971	0.976	0.101	0.098	1

Note: The numbers 1–7 refers to input variables identified in Table 3.

Analysis of the R^2 and RMSE values showed that Perm among single input variables, and Perm and TMP among groups of two variables majorly affected the effluent TN models. The most important variables among groups of three variables were Perm, TMP, and MLVSS, and the noteworthy variables among groups of four variables were Perm, TMP, MLVSS, and influent TN. Furthermore, Perm, TMP, MLVSS, influent TN, and SRT among groups of five variables, and Perm, TMP, MLVSS, influent TN, influent SRT, and effluent COD between groups of six variables majorly affected the variation of the effluent TN models (Table 7). The results showed high collaboration of the RBFANN-GA and MLPANN-GA in the modeling of effluent TN, the same as effluent BOD and COD models. The analysis of the R^2 and RMSE indicated that the RBFANN-GA modeled effluent TN was more accurate as compared with MLPANN-GA. Table 7

shows the importance order of each input variable and the joint variables for the prediction of effluent TN. The sensitivity analyses of input variables for effluent TN models by RBFANN-GA and MLPANN-GA indicated that the effluent TN is influenced by Perm, TMP, MLVSS, influent TN, SRT, influent COD, and influent BOD, respectively. The values of sensitivity to Perm, TMP, MLVSS, influent TN, SRT, influent COD, and influent BOD for effluent TN were 61.2, 54.8, 41.5, 35.8, 26.3, 9.9, and 7.6%, respectively. This study indicates that the Perm and TMP as well as MLVSS significantly affect the effluent TN models.

For the effluent TP models, influent TP among single input variables, and influent TP and MLSS among groups of two variables were the most important variables. The most correlated and effective combination among groups of three variables was combination of influent TP, MLSS, and influent BOD, and the

Table 8
Effect of single and joint variables on the effluent TP models by RBFANN-GA and the MLPANN-GA

Input variable No.	TP models (RBFANN-GA)				TP models (MLPANN-GA)				Importance order
	R^2		RMSE (mg l^{-1})		R^2		RMSE (mg l^{-1})		
	Train	Test	Train	Test	Train	Test	Train	Test	
1	0.515	0.595	0.743	0.651	0.487	0.512	0.856	0.734	3
2	0.511	0.585	0.752	0.659	0.494	0.523	0.871	0.757	4
3	0.682	0.725	0.591	0.558	0.646	0.698	0.612	0.583	1
4	0.428	0.482	0.841	0.755	0.421	0.446	0.894	0.813	5
5	0.526	0.613	0.711	0.628	0.503	0.566	0.749	0.704	2
6	0.421	0.458	0.849	0.768	0.411	0.405	0.899	0.908	6
7	0.408	0.414	0.857	0.791	0.386	0.374	0.931	0.943	7
3–1	0.579	0.687	0.697	0.614	0.543	0.628	0.742	0.690	2
3–2	0.567	0.666	0.728	0.637	0.526	0.568	0.847	0.719	3
3–4	0.562	0.655	0.737	0.645	0.534	0.581	0.862	0.742	4
3–5	0.751	0.812	0.579	0.546	0.698	0.775	0.606	0.571	1
3–6	0.471	0.541	0.824	0.738	0.455	0.495	0.885	0.797	5
3–7	0.463	0.513	0.832	0.751	0.444	0.451	0.891	0.894	6
3–5–1	0.845	0.891	0.506	0.412	0.808	0.866	0.521	0.511	1
3–5–2	0.776	0.797	0.602	0.521	0.689	0.782	0.624	0.604	2
3–5–4	0.618	0.666	0.661	0.562	0.609	0.635	0.771	0.661	3
3–5–6	0.613	0.702	0.669	0.567	0.618	0.649	0.784	0.681	5
3–5–7	0.514	0.578	0.748	0.649	0.526	0.553	0.805	0.732	6
3–5–1–2	0.941	0.922	0.306	0.302	0.915	0.906	0.327	0.322	1
3–5–1–4	0.901	0.893	0.319	0.365	0.877	0.819	0.385	0.367	2
3–5–1–6	0.894	0.878	0.323	0.369	0.889	0.837	0.392	0.379	3
3–5–1–7	0.749	0.723	0.362	0.423	0.758	0.714	0.402	0.407	4
3–5–1–2–4	0.984	0.975	0.235	0.245	0.966	0.962	0.304	0.317	1
3–5–1–2–6	0.963	0.946	0.245	0.254	0.935	0.927	0.385	0.331	2
3–5–1–2–7	0.956	0.931	0.258	0.267	0.928	0.917	0.392	0.347	3
3–5–1–2–4–6	0.989	0.981	0.185	0.201	0.971	0.974	0.210	0.211	1
3–5–1–2–4–7	0.968	0.962	0.193	0.208	0.949	0.943	0.228	0.224	2
3–5–1–2–4–6–7	0.998	0.997	0.124	0.129	0.988	0.986	0.136	0.139	1

Note: The numbers 1–7 refers to input variables identified in Table 3.

noteworthy combination among groups of four variables was combination of influent TP, MLSS, influent BOD, and influent COD. Moreover, influent TP, MLSS, influent BOD, influent COD, and SRT was the most noticeable combination among groups of five variables. The influent TP, MLSS, influent BOD, influent COD, SRT, and Perm between combinations of six variables majorly affected the variation of the effluent TP models (Table 8). The analysis of the R^2 and RMSE indicated that the RBFANN-GA modeled effluent TP was more accurate as compared with MLPANN-GA. Table 8 shows the importance order of each input variable and the joint variables for the prediction of effluent TP. The sensitivity analyses of input variables for effluent TP models by RBFANN-GA and MLPANN-GA showed that the effluent TP is influenced by influent TP, MLSS, influent BOD, influent COD, SRT,

and Perm and TMP, respectively. The values of sensitivity to influent TP, MLSS, influent BOD, influent COD, SRT, and Perm and TMP for effluent TP were 57.4, 51.9, 44.6, 40.2, 19.7, 7.3, and 6.9%, respectively. This study indicates that the influent TP, MLSS, influent BOD, and influent COD majorly affect the effluent TP models.

Figs. 12 and 13 show the effluent TN and effluent TP models by RBFANN-GA and MLPANN-GA, according to all experimental data-sets. The results indicated a successful prediction and an almost perfect match between the experimental and the predicted values of effluent TN and effluent TP for both ANN models. The results of effluent TN and effluent TP modeling show the stronger generalization ability of the RBFANN-GA compared with the MLPANN-GA. Figs. 14 and 15 show the regression lines for all

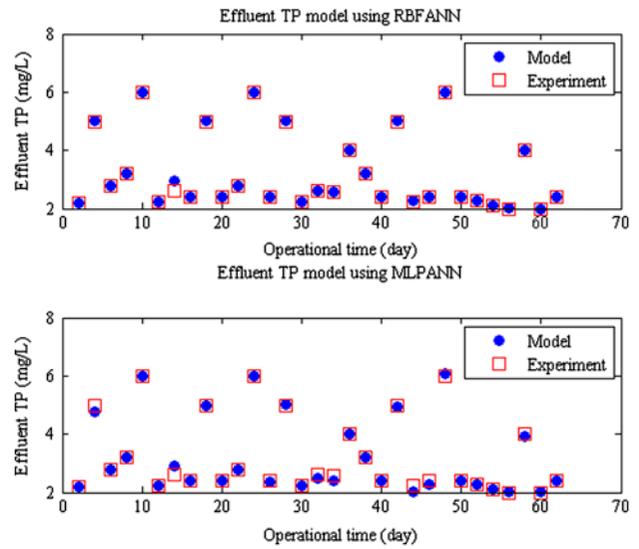
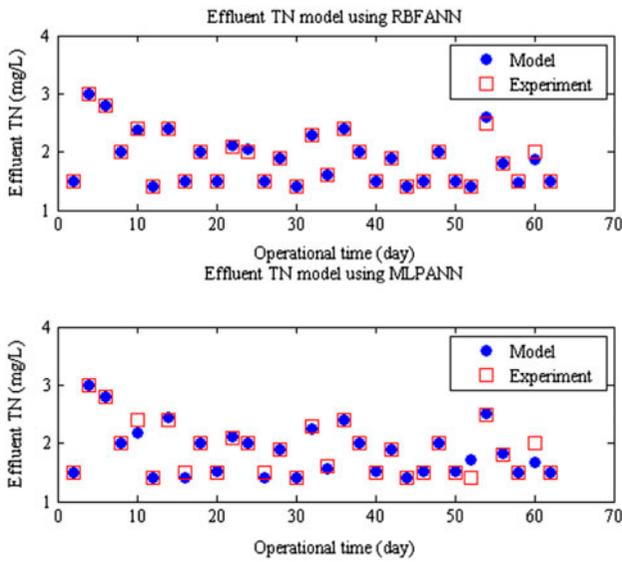


Fig. 12. Effluent TN models by RBFANN-GA and MLPANN-GA according to all data-set.

Fig. 13. Effluent TP models by RBFANN-GA and MLPANN-GA according to all data-set.

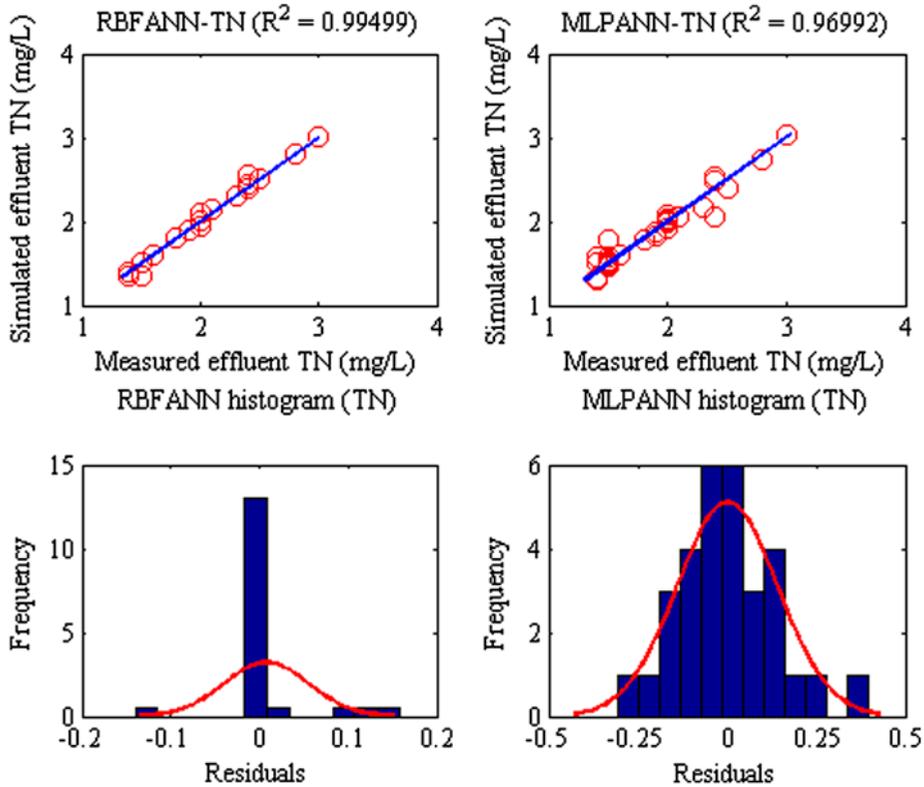


Fig. 14. Regression plots and residuals of RBFANN-GA and MLPANN-GA for the prediction of effluent TN.

data-sets based on the RBFANN-GA and MLPANN-GA models for the effluent TN and effluent TP values. The values of R^2 based on the RBFANN-GA and MLPANN-GA models for effluent TN models were 0.994 and 0.969, and for effluent TP models were 0.999 and 0.987, respectively. Furthermore, to measure the predictive capability of ANN models, a set of data

was used in the effluent TN and effluent TP simulation process [39]. The RMSE values for the train and test models by RBFANN-GA were 0.081 and 0.074 mg^{-1} for the effluent TN, and were 0.124 and 0.129 mg^{-1} for the effluent TP, respectively. The RMSE values for the train and test models by MLPANN-GA were 0.101 and 0.098 mg^{-1} for the effluent TN, and

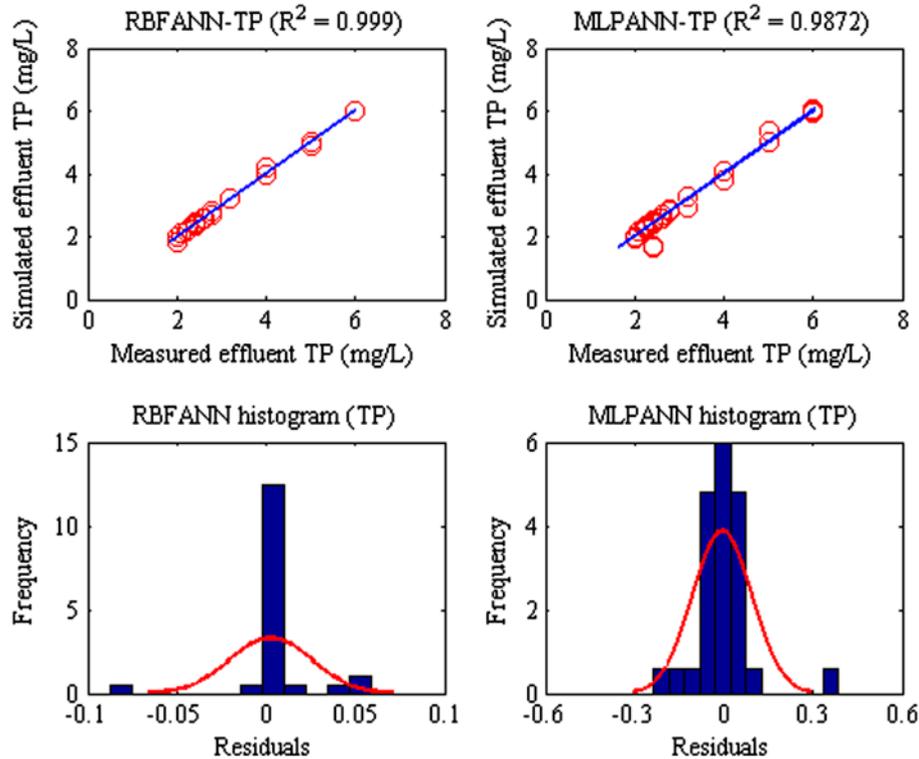


Fig. 15. Regression plots and residuals of RBFANN-GA and MLPANN-GA for the prediction of effluent TP.

Table 9

Comparison of RBFANN and MLPANN with and without applying GA for the TN and TP models

Model	R^2			RMSE (mg l^{-1})			Percentage of error (%)	Rank of model (Accuracy)
	All	Train	Test	All	Train	Test		
Total nitrogen (TN) models								
RBFANN	0.912	0.898	0.924	0.131	0.159	0.128	9	3
RBFANN-GA	0.995	0.992	0.994	0.072	0.081	0.074	4	1
MLPANN	0.864	0.853	0.784	0.161	0.167	0.184	10	4
MLPANN-GA	0.969	0.971	0.976	0.105	0.101	0.098	6	2
Total phosphorus (TP) models								
RBFANN	0.938	0.924	0.919	0.201	0.224	0.237	7	3
RBFANN-GA	0.999	0.998	0.997	0.121	0.124	0.129	3	1
MLPANN	0.928	0.921	0.893	0.233	0.247	0.325	11	4
MLPANN-GA	0.991	0.988	0.986	0.133	0.136	0.139	4	2

were 0.136 and 0.139 mg⁻¹ for the effluent TP, respectively. The mean average error by RBFANN-GA and MLPANN-GA models were 4 and 6% of input values in the prediction of effluent TN, and were 3 and 4% of input values in the prediction of effluent TP, respectively.

Figs. 14 and 15 show the residuals of effluent TN and effluent TP models attained by the RBFANN-GA and the MLPANN-GA for all data-set versus the frequency of the data. The approximately normal distribution of residuals produced by the RBFANN-GA and the MLPANN-GA models has led to a normal distribution, which is illustrated by a specific bell-shaped curve [40]. The specific bell-shaped curve demonstrates that our results are symmetrical and their axis round around zero for all data-sets. The effect of applying GA on the RBFANN and the MLPANN in the prediction of effluent TN and effluent TP has been presented in Table 9. The results indicated that the precision and accuracy of all effluent TN and effluent TP models increased when GA was applied to the ANN models. The results of the RMSE and R^2 for effluent TN and effluent TP models indicated that the RBFANN-GA is the most precise model with a mean average error about 3 to 4%.

4. Conclusion

Two hybrid artificial neural network–genetic algorithm models were developed to accurately predict the effluent BOD, COD, TN, and TP in a SBR with simultaneous upward and downward aeration system. The input variables of the networks were influent BOD, influent COD, influent TN or influent TP, SRT, MLSS, membrane permeability, and TMP. Training procedures of the effluent BOD, COD, TN, and TP were successful for both the MLPANN-GA and RBFANN-GA models. The training and testing models showed an almost perfect match between the experimental and predicted values. Based upon the statistical analysis, results indicated that there is a very little difference between the predicted and experimental values of the effluent BOD, COD, TN, and TP. The models showed low RMSE values and high R^2 values very close to one, demonstrating high accuracy of these models to predict output variables. The hyperbolic tangent sigmoid transfer function (tansig) at the hidden layer and linear transfer function (purelin) at the output layer were the optimal functions. The LM had smaller RMSE values compared with other backpropagation algo-

rithms. So, the LM was considered the training algorithm in this research. The optimum models for the prediction of the effluent BOD, COD, TN, and TP were obtained with the hidden layer consisting of nine neurons.

The sensitivity analyses of input variables for the effluent BOD and COD models by RBFANN-GA and MLPANN-GA indicated that the effluent BOD and COD are influenced by Perm, TMP, MLVSS, influent COD, and influent BOD, SRT, and TN, respectively. The values of sensitivity to Perm, TMP, MLVSS, influent COD, influent BOD, SRT, and influent TN for effluent BOD were 68.1, 66.3, 54.2, 32.5, 24.9, 12.4, and 6.4%, respectively. The values of sensitivity to mentioned variables for effluent COD were 69.7, 67.5, 55.1, 31.9, 24.7, 13.5, and 7.8%, respectively. This study showed that the Perm and TMP as well as MLVSS significantly affect the effluent BOD and COD models. The sensitivity analyses of input variables for effluent TN models by RBFANN-GA and MLPANN-GA indicated that the effluent TN is influenced by Perm, TMP, MLVSS, influent TN, SRT, influent COD, and influent BOD, respectively. The values of sensitivity to Perm, TMP, MLVSS, influent TN, SRT, influent COD, and influent BOD for effluent TN were 61.2, 54.8, 41.5, 35.8, 26.3, 9.9, and 7.6%, respectively. This study indicated that the Perm and TMP as well as MLVSS significantly affect the effluent TN models. The results showed that the effluent TP is influenced by influent TP, MLSS, influent BOD, influent COD, SRT, Perm, and TMP, respectively. The values of sensitivity to influent TP, MLSS, influent BOD, influent COD, SRT, Perm, and TMP for effluent TP were 57.4, 51.9, 44.6, 40.2, 19.7, 7.3, and 6.9%, respectively. This study indicates that the influent TP, MLSS, influent BOD, and influent COD majorly affect the effluent TP models.

It became clear that the models based on the GA were much better than those models without GA from the viewpoint of the achievement of accurate prediction of the effluent BOD, COD, TN, and TP. The results of this study indicated that the accuracy of all ANN models increases when GA is applied to neural networks. The mean average error for hybrid models varied from 3 to 8%. The results of this study illustrate an approximately normal distribution of residuals produced by the RBFANN-GA and the MLPANN-GA models. A normal distribution of variation results in a Gaussian curve, which demonstrates that our results are symmetrical and their axis round around zero.

Acknowledgments

The authors are grateful to Ekbatan WWTP for their technical and logistical assistance during this work which was supported by authors. And also, we wish to thank Mohammad Reza Ghanbari and Ali Reza Jafari for technical help with modeling by artificial neural network.

Nomenclature

ASP	—	activated sludge process
SMBR	—	submerged membrane bioreactor
SRT	—	sludge retention time
ASM1	—	activated sludge model number one
ANN	—	artificial neural network
TDS	—	total dissolved solids
COD	—	chemical oxygen demand
T	—	temperature
TMP	—	trans-membrane pressure
CFV	—	cross-flow velocity
FANN	—	feed forward artificial neural network
MLP	—	multi-layer perceptron
RBF	—	radial basis function
TSS	—	total suspended solids
BOD	—	biochemical oxygen demand
DO	—	dissolved oxygen
BP	—	back propagation
GA	—	genetic algorithm
MLPANN	—	multi-layer perceptron artificial neural network
RBFANN	—	radial basis function artificial neural network
TN	—	total nitrogen
TP	—	total phosphorus
MLSS	—	mixed liquor suspended solids
Perm	—	permeability
WWTP	—	wastewater treatment plant
MLVSS	—	mixed liquor volatile suspended solids
LM	—	Levenberg–Marquardt
GDB	—	gradient descent backpropagation
R^2	—	coefficient of determination
RMSE	—	root mean squared error
HTS	—	hyperbolic tangent sigmoid
PL	—	pure linear

References

- [1] Z.-B. Chen, D.-X. Hu, N.-Q. Ren, Y. Tian, Z.-P. Zhang, Biological COD reduction and inorganic suspended solids accumulation in a pilot-scale membrane bioreactor for traditional Chinese medicine wastewater treatment, *Chem. Eng. J.* 155 (2009) 115–122.
- [2] A.R. Pendashteh, L.C. Abdullah, A. Fakhru'l-Razi, S.S. Madaeni, Z. Zainal Abidin, D.R. Awang Biak, Evaluation of membrane bioreactor for hypersaline oily wastewater treatment, *Process Saf. Environ. Prot.* 90 (2012) 45–55.
- [3] Z. Chen, H. Wang, N. Ren, Z. Chen, M. Cui, S. Nie, Performance and model of a novel membrane bioreactor to treat the low-strength complex wastewater, *Bioresour. Technol.* 114 (2012) 33–45.
- [4] D. Güçlü, Ş. Dursun, Artificial neural network modelling of a large-scale wastewater treatment plant operation, *Bioprocess Biosyst. Eng.* 33 (2010) 1051–1058.
- [5] M. Henze, C. Grady, W. Gujer, G. Marais, T. Matsuo, A general model for single-sludge wastewater treatment systems, *Water Res.* 21 (1987) 505–515.
- [6] H. Moral, A. Aksoy, C.F. Gokcay, Modeling of the activated sludge process by using artificial neural networks with automated architecture screening, *Comput. Chem. Eng.* 32 (2008) 2471–2478.
- [7] M. Bagheri, S. Mirbagheri, M. Ehteshami, Z. Bagheri, Modeling of a sequencing batch reactor treating municipal wastewater using multi-layer perceptron and radial basis function artificial neural networks, *Process Saf. Environ. Prot.* 93 (2015) 111–123. Available from: <<http://dx.doi.org/10.1016/j.psep.2014.04.006>>.
- [8] W. Naessens, T. Maere, I. Nopens, Critical review of membrane bioreactor models—Part 1: Biokinetic and filtration models, *Bioresour. Technol.* 122 (2012) 95–106.
- [9] S. Geissler, T. Wintgens, T. Melin, K. Vossenkaul, C. Kullmann, Modelling approaches for filtration processes with novel submerged capillary modules in membrane bioreactors for wastewater treatment, *Desalination* 178 (2005) 125–134.
- [10] Ö. Çinar, H. Hasar, C. Kinaci, Modeling of submerged membrane bioreactor treating cheese whey wastewater by artificial neural network, *J. Biotechnol.* 123 (2006) 204–209.
- [11] A.R. Pendashteh, A. Fakhru'l-Razi, N. Chaibakhsh, L.C. Abdullah, S.S. Madaeni, Z.Z. Abidin, Modeling of membrane bioreactor treating hypersaline oily wastewater by artificial neural network, *J. Hazard. Mater.* 192 (2011) 568–575.
- [12] R. Badrnezhad, B. Mirza, Modeling and optimization of cross-flow ultrafiltration using hybrid neural network-genetic algorithm approach, *J. Ind. Eng. Chem.* 20 (2014) 528–543.
- [13] B. Suchacz, M. Wesołowski, The recognition of similarities in trace elements content in medicinal plants using MLP and RBF neural networks, *Talanta* 69 (2006) 37–42.
- [14] M.T. Beheshti, A. Tehrani, B. Ghanbari, An optimized adaptive fuzzy inverse kinematics solution for redundant manipulators, *IEEE International Symposium on Intelligent Control*, October 2003, Houston, TX, pp. 924–929, doi: [10.1109/ISIC.2003.1254760](https://doi.org/10.1109/ISIC.2003.1254760).
- [15] P.-Y. Zhang, T.-S. Lü, L.-B. Song, RBF networks-based inverse kinematics of 6R manipulator, *Int. J. Adv. Manuf. Technol.* 26 (2005) 144–147.
- [16] F. Fang, B.-J. Ni, W.-M. Xie, G.-P. Sheng, S.-G. Liu, Z.-H. Tong, H.-Q. Yu, An integrated dynamic model for simulating a full-scale municipal wastewater treatment plant under fluctuating conditions, *Chem. Eng. J.* 160 (2010) 522–529.
- [17] R.S. Sexton, R.E. Dorsey, J.D. Johnson, Toward global optimization of neural networks: A comparison of the genetic algorithm and backpropagation, *Decis. Support Syst.* 22 (1998) 171–185.

- [18] M.C. Rezende, C.B. Costa, A.C. Costa, M. Maciel, Optimization of a large scale industrial reactor by genetic algorithms, *Chem. Eng. Sci.* 63 (2008) 330–341.
- [19] D. Andrew, *Standard Methods for the Examination of Water and Wastewater*, APHA-AWWA-WEF, Washington, DC, 2005.
- [20] J. Caro, Basic principles of membrane technology, *Z. für Phys. Chem.* 203 (1998) 263–263.
- [21] L. Fausett, *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*, Prentice-Hall, Inc., Upper Saddle River, NJ, 1994.
- [22] B. Ráduly, K.V. Gernaey, A. Capodaglio, P.S. Mikkelsen, M. Henze, Artificial neural networks for rapid WWTP performance evaluation: Methodology and case study, *Environ. Model. Softw.* 22 (2007) 1208–1216.
- [23] B.G. Lipták, *Instrument Engineers' Handbook, Volume Two: Process Control and Optimization*, CRC press, Radnor, PA, 2006.
- [24] M. Kashaninejad, A. Dehghani, M. Kashiri, Modeling of wheat soaking using two artificial neural networks (MLP and RBF), *J. Food Eng.* 91 (2009) 602–607.
- [25] S.S. Chiddarwar, N. Ramesh Babu, Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6R serial robot by a fusion approach, *Eng. Appl. Artif. Intell.* 23 (2010) 1083–1092.
- [26] J.H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis With Applications to Biology, Control, and Artificial Intelligence*, U Michigan Press, Oxford, 1975.
- [27] D. Prakotpol, T. Srinophakun, GAPinch: Genetic algorithm toolbox for water pinch technology, *Chem. Eng. Process: Process Intensif.* 43 (2004) 203–217.
- [28] I. Rocha, E. Parente Jr., A. Melo, A hybrid shared/distributed memory parallel genetic algorithm for optimization of laminate composites, *Compos. Struct.* 107 (2014) 288–297.
- [29] W. Rauch, P. Harremoës, Genetic algorithms in real time control applied to minimize transient pollution from urban wastewater systems, *Water Res.* 33 (1999) 1265–1277.
- [30] B. Pavez-Lazo, J. Soto-Cartes, A deterministic annular crossover genetic algorithm optimisation for the unit commitment problem, *Expert Syst. Appl.* 38 (2011) 6523–6529.
- [31] Claudio Comis Da Ronco, Ernesto Benini, A Simplex Crossover based evolutionary algorithm including the genetic diversity as objective, *Appl. Soft Comput.* 13 (2013) 2104–2123.
- [32] K. Deep, M. Thakur, A new mutation operator for real coded genetic algorithms, *Appl. Math. Comput.* 193 (2007) 211–230.
- [33] C. Waewsak, A. Nopharatana, P. Chairasert, Neural-fuzzy control system application for monitoring process response and control of anaerobic hybrid reactor in wastewater treatment and biogas production, *J. Environ. Sci.* 22 (2010) 1883–1890.
- [34] C.-M. Lee, C.-N. Ko, Time series prediction using RBF neural networks with a nonlinear time-varying evolution PSO algorithm, *Neuro. comput.* 73 (2009) 449–460.
- [35] S. Ghoreishi, E. Heidari, Extraction of Epigallocatechin-3-gallate from green tea via supercritical fluid technology: Neural network modeling and response surface optimization, *J. Supercrit. Fluids* 74 (2013) 128–136.
- [36] E. Al-Daoud, A comparison between three neural network models for classification problems, *J. Artif. Intell.*, 2 (2009) 56–64.
- [37] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, S. Tarantola, *Global sensitivity analysis: The primer*, Wiley, Chichester, 2008.
- [38] D. Hanbay, İ. Turkoglu, Y. Demir, Prediction of chemical oxygen demand (COD) based on wavelet decomposition and neural networks, *CLEAN – Soil, Air, Water* 35 (2007) 250–254.
- [39] K.V. Gernaey, M. van Loosdrecht, M. Henze, M. Lind, S.B. Jørgensen, Activated sludge wastewater treatment plant modelling and simulation: state of the art, *Environ. Model. Softw.* 19 (2004) 763–783.
- [40] A. Chamkalani, M. Amani, M.A. Kiani, R. Chamkalani, Assessment of asphaltene deposition due to titration technique, *Fluid Phase Equilib.* 339 (2013) 72–80.