International Journal of Computational Intelligence and Applications

Coagulant Dosage Determination in a Water Treatment Plant Using Dynamic Neural Network Models

Leonaldo Silva Gomes, Francisco Alexandre A. Souza*, Ricardo Silva Thé Pontes, Tobias R. Fernandes Neto

DEE - Department of Electrical Engineering; Federal University of Ceará; Ceará, Brazil
GPAR - Research Group for Automation and Robotics; Federal University of Ceará; Ceará, Brazil

Rui Alexandre M. Araújo

DEEC-Department of Electrical and Computer Engineering; University of Coimbra, Pólo II; Coimbra, Portugal
ISR-Institute of Systems and Robotics; University of Coimbra, Pólo II; Coimbra, Portugal

Received (received date)
Revised (revised date)

A common step in most of water treatment plants is the chemical coagulation. The chemical coagulation is the process of destabilizing the colloidal particles suspended in raw water by the addition of coagulants. Generally, the determination of the quantity of coagulant to be added to water is made manually by jar tests. However, the manual control has slow response to changes of raw water and it requires intensive laboratory analysis. To reduce the manual effort and to improve the response to change in raw water quality, this work proposes the determination of the coagulant dosage using dynamic neural network modeling using the available sensors as input of the model. The case of study is a large scale water treatment plant in Ceará, Brazil, where the kinds of coagulants added to water are the aluminum sulphate (AS) and poly aluminum chloride (PAC). Several dynamic neural network models with different combinations of the input variables have been evaluated. The best solution found is composed by a non-linear autoregressive with exogenous input (NARX) model having three input variables, the pH in raw and coagulated water, and the turbidity in the coagulated water, with coefficient of determination of $R^2 = 0.95$ and $R^2 = 0.91$ for the AS and PAC dosage prediction, respectively.

Keywords: Coagulant dosage; drinking water; dynamic neural networks; NARX model; jar test.

1. Introduction

Drinking water with quality for human consumption is a requirement claimed by society, as well a strategic necessity for the maintenance of public health. The water and sanitation companies (WSC) should provide drinking water within the standards of potability and according to the regulatory constraints. This has driven the

*Corresponding author at: Department of Electrical Engineering, Federal University of Ceará. Tel.: +55-85-3366.9579. Email address: alexandre@dee.ufc.br (Francisco Alexandre A. Souza)

WSC to, more and more, perform the automation of their water treatment plants (WTP).

However, some automation services in WTPs need high financial investments, impeding the automation of some operations in the WTPs. A particular example is the automation of the coagulation-flocculation stage. The Coagulation-Flocculation stage is one of the main processes in a WTP, they are designed to help in the removal of colloidal particles by the addition of coagulants in the raw water. These coagulants are designed to neutralize the electrical charges of the colloidal particles present in the water. The most common coagulants applied in the water are the aluminum sulphate (AS) and poly aluminum chloride (PAC). After Coagulation, the water is gently mixed in the Flocculation, facilitating the agglomeration of the fine particles, so generating the flocks which can be easily removed from water. A correct dosage of coagulants is an important goal by two main reasons: 1) to make the coagulation phase more efficient, and 2) to lower the costs associated with inputs dosages. Excessive coagulant dosage leads to increased treatment costs and public health concerns [19], while underdosing leads to a failure to meet the water quality targets and hence providing a less efficient operation of the water treatment plant [27]. The main difficulty is to determine the optimum coagulant dosage related to the influent of raw water. Actually, both manual and automatic methods are available to determine optimum coagulant dosage rate.

Automatic coagulant control is ensured mainly by streaming current detectors (SCD) [9,5,25]. A SCD is an instrument for measuring the charge that exists on small, suspended particles in liquid. The SCD is the most common instrument used to measure coagulated particle stability for the feedback control of coagulant dosage [33]. However, such equipment has a high cost of investment and operation, making its utilization impracticable in some WTPs.

The coagulant dosage can also be manualy determined by the jar testing [27,5]. The jar test is a test intended to simulate the Coagulation-Flocculation process in a laboratory size scale. Jar testing involves taking a number raw water samples and applying different quantities of coagulant to each sample. After a short period of time each sample is assessed for water quality and the dosage that produces the optimal result is used as a set point. Operators change the dose and make a new jar test if the quality of raw water changes. However, jar tests are relatively expensive and take a long time to conduct [22], and as a result of the long time necessary to obtain the jar results, they cannot cope with the rapid change of raw water quality [16,17].

A common factor among the WTPs is the presence of sensors designed for the supervision of the water treatment. Furthermore, such sensors can be used beyond process monitoring purposes, as for example, as input of a regression model to predict/determine the optimal coagulant dosage to be added to the water, as an alternative to jar tests. Such idea of predicting the optimal coagulant dosage based on the the parameters of water is not new and it has been evaluated by several other authors [12,16,22,31,15] (in all of these works PAC was the coagulant to be pre-

dicted). In [12] a multilayer perceptron (MLP) neural network model was used to predict the coagulant dosage in a WTP at the city of Saint Foy, Canada and the raw water quality parameters (pH, turbidity, temperature, and conductivity) collected from the process sensors were used as input to the MLP model. In [16] several pre-processing techniques, such as outlier removal and down-sampling of data were tested to improve the MLP performance in predicting the coagulant dosage in a WTP at the city of Chungju, South Korea. The inputs used to compose the MLP model in [16] were four raw water quality parameters (collected from the process sensors), the temperature, pH, turbidity, and alkalinity. [22] used a MLP model to predict the coagulant dosage in a laboratory scale (the samples used to build the model were collected from different locations at the state of Victoria and South of Australia), and the inputs used to compose the model were the raw water and treated water quality parameters. The raw water parameters were the pH, turbidity, color, ultraviolet absorbance at a wavelength of 254 [nm] (UVA-254), and the concentration of dissolved organic carbon (DOC). The treated water parameters were the turbidity, the color, and the UVA-254. In [31] an MLP and adaptive neuro-fuzzy inference system (ANFIS) models were evaluated in predicting the coagulant dosage at laboratory scale (water samples were collected from a WTP at Taipei County, Taiwan and laboratory experiments were performed to determine the water quality parameters), the MLP model has shown to provide the best performance results having the turbidity of treated water, and the amount of coagulant dosage on the previous day as inputs to the MLP model. In [15] an ANFIS model was learned to predict the coagulant dosage in a WTP at Boudouaou, Algeria, and six raw water parameters collected from process sensors were considered as input of the model: the turbidity, conductivity, temperature, dissolved oxygen (DO), UVA-254, and pH.

The objective of this work is to establish a regression model which uses a small number of input variables to correctly predict the dosages of PAC and AS in a real WTP in the State of Ceará, Brazil. The motivation behind the use a few number of input variables is to make the implementation of such regression model reproducible in a WTP equipped with a small number of sensors. Moreover, having a lower number of input variables is a positive factor for decreasing implementation costs. In this work, the turbidity and pH in raw and coagulated water and several dynamic neural network models (DNN) were evaluated in different settings (i.e. with different parameters and different combinations of input variables) in order to determine the best regression model to predict the PAC and AS at the WTP of Ceará. To select the regression model which produced the best trade-off between the complexity (number of input variables and parameters) and prediction results, the performance measurement was going to be based on the C_p statistics [30], [23]. The historical data used for modeling comprises 3 years of operational data, and it contains information about the coagulant dosage values (determined by jar tests) and parameters values from raw and coagulated water at every 2 [h], with a total of 13043 data samples. The models were trained using the first 2 years (8723 samples)

of data, and the data of the remaining 1 year (4320 samples) was used as test set. This 1 year test data set includes rainy and dray seasons. The results have shown that having three input variables, the pH in raw and coagulated water, and the turbidity in the coagulated water is sufficient to predict the PAC and AS dosages at the WTP of Ceará.

In summary, the contributions of the this paper are: 1) evaluate several dynamic models instead of static models in the prediction of coagulant dosage; 2) evaluate all the models in a 1 year data set, which includes rainy and dray seasons; 3) establish a regression model which uses three sensors to correctly predict the dosages of PAC and AS in a real WTP in the State of Ceará, Brazil.

The paper is organized as follows. In Section 2 the water treatment plant configuration is described. An overview about the DNN models used in the modelling phase is done in Section 3. Section 4 describes the methodology used to set the parameters of DNN models and the methodology used to evaluate the different configurations of the DNN models. Section 5 is dedicated to the experimental results. Finally, Section 6 gives concluding remarks

2. Water Treatment Plant

The WTP at Ceará takes the following steps to treat the water: Pre-treatment, Coagulation, Flocculation, Sedimentation, Filtration and Disinfection. A short summary about the process is given as follows. Raw water is pre-treated prior to the main processes within the WTP. The Pre-treatment done in the plant are the algae control, and treatment to remove metals such as manganese and iron, where the latter is done through the addition of chlorine. The Coagulation and Flocculation stages are designed to help the removal of dissolved and suspended particles. During Coagulation, the chemical coagulants PAC and AS are added to raw water aiming to neutralize the electrical charges of the fine particles present in the water. After Coagulation, the water is gentle mixed during the Flocculation stage, facilitating the agglomeration of fine particles, so generating flocks which can then be easily removed in the subsequent stages. The Sedimentation stage prepares the water for effective filtration, by allowing the flocks to settle by gravity. After sedimentation, only small unsettled particles remain in the water, and they are removed in the Filtration stage. In Filtration, the suspended particles from water, and microorganisms in general, are removed by passing the water through a filter. As the water passes through the filter, flock and impurities get stuck in it and the clean water goes through. The clean water from the Filtration stage is then treated with chloride in the Disinfection stage.

The plant has the capacity of $36.000\,[\mathrm{m}^3/\mathrm{d}]$ and its raw water comes from a reservoir whose water quality is subject to seasonal change. The coagulant dosage control existing at the WTP in Ceará is carried out by jar tests at every $2\,[\mathrm{h}]$ and it is responsible for defining the dosage values of the PAC and AS coagulants. The plant also stores the values of pH and turbidity in raw and coagulated water, and

these variables are used for modeling the PAC and AS dosage values.

3. DNN Models

In this work the following notation will be used: considering a collection of variables $x_1, \ldots, x_n$, such a collection can be collectively represented or organized either as set of variables $X = \{x_1, \ldots, x_n\}$, or equivalently as a vector of variables $\mathbf{x} = [x_1, \ldots, x_n]^T$.

The following four dynamic regression models are discussed in this section: the focused time lagged feed-forward network (FTLFN) [28], the distributed time lagged feed-forward network (DTLFN) [14], the Elman recurrent network (ERN) [11], and the non-linear autoregressive with exogenous inputs (NARX) [20]. All these models are classified as dynamical models. In this work, just models with one hidden layer are considered. Then, with just one hidden layer, these DNN models share the same structure of the traditional multilayer perceptron model (MLP) with one hidden layer and h neurons. The output of the traditional MLP, with one hidden layer is given by:

$$y(k) = \mathbf{z}(k)^T \mathbf{w}_O + b_O, \tag{1}$$

where

$$\mathbf{z}(k) = g\left(\mathbf{u}(k)^T \mathbf{W}_I + \mathbf{b}_I\right), \tag{2}$$

and $y(k) \in \mathbb{R}$ is the model output, $\mathbf{z}(k) \in \mathbb{R}^h$ is the hidden layer output vector, $\mathbf{u}(k) \in \mathbb{R}^D$ is the k-th sample of input vector of model (1)-(2), $\mathbf{W}_I \in \mathbb{R}^{D \times h}$ is the matrix of weights connecting the D inputs to h hidden layer nodes, and $\mathbf{b}_I \in \mathbb{R}^h$ is the vector of hidden layer nodes biases. The output weights that connect the hidden neurons to the output neuron and the output bias are represented by $\mathbf{w}_O \in \mathbb{R}^h$ and $b_O \in \mathbb{R}$, respectively. $g(\cdot)$ represents the activation functions of hidden layer nodes. In this work $g(\cdot)$ is tangent sigmoid, and for an input variable a it is given by:

$$g(a) = \frac{2}{1 + e^{2a}} - 1, \tag{3}$$

which is bounded between -1 and $+1$. For a vector $\mathbf{a} = [a_1, \ldots, a_A]^T \in R^A$ the output of the tangent sigmoid is defined as $g(\mathbf{a}) = [g(a_1), \ldots, g(a_A)]^T$.

The FTLFN, DTLFN, ERN, and NARX models, with one hidden layer, have the same format of (1), but they differ in the way which the input vector $\mathbf{u}(k)$ is composed. The input variables (not the input of the model $\mathbf{u}(k)$ in (2)) are defined as $\mathbf{x}(k) \in \mathbb{R}^N$. A more detailed discussion about each DNN model is given as follows.

3.1. *FTLFN Model*

The FTLFN model is part of a general class of dynamic networks, called focused networks, in which the dynamics appear only at the input layer. In the FTLFN model, the input vector $\mathbf{u}(k)$ in (1), is given by the actual and delayed components

of input variables $\mathbf{x}(k)$, $\mathbf{u}(k) = [\mathbf{x}(k), \ldots, \mathbf{x}(k - p_x)]^T$, where p_x is the maximum lag of inputs. The use of delayed inputs gives temporal dynamics capability to the FTLFN model, because the current output depends not only on the current input $\mathbf{x}(k)$, but also on collection of its past values. The FTLFN was first proposed for dealing with speech recognition [28], but it is widely applied in many other domains, like the prediction of total suspended solids (TSS) in a water treatment plants [3,8], and forecasting of stock markets [18].

There are a number of key decisions required to completely specify a FTLFN model. These include the maximum time lag p_x, the number of hidden neurons h, the activation function of hidden layer nodes $g(\cdot)$, and the algorithm used for learning the model parameters $\mathbf{W}_I$, $\mathbf{b}_I$, $\mathbf{w}_O$, and b_O. The dimension of $\mathbf{u}(k)$ increases proportionally to the maximum lag p_x and it should be selected carefully. The dimensionality of model input is equal to: $D = N \cdot p_x + 1$. Large values of D can increase the training time and even lead to poor generalization in the prediction results [4].

3.2. *DTLFN Model*

The DTLFN model was originally proposed by [29] and has been applied in network traffic prediction [1] and time series prediction [7]. This model has one hidden layer, in the form of (1), and has inputs $u_j(k)$, where each $u_j(k)$ is equal to the output of a finite impulse response filter (FIR) applied to input variable $x_j(k)$. The output of a filter FIR applied on input variable $x_j(k)$, corresponds to a weighted sum of actual and past values of this variable:

$$\varphi(x_j(k), p_x) = \sum_{i=0}^{p_x} c_{ij}\, x_j(k - i), \tag{4}$$

where p_x is the order of filter FIR, and c_{ij} is the coefficient of filter FIR for the input variable x_j and filter line i. The output of the DTLFN model is given by (1), having as input vector $\mathbf{u}(k) = [\varphi(x_1(k), p_x), \ldots, \varphi(x_N(k), p_x)]^T$.

The model parameters which should be defined/tuned for the DTLFN model are the order of filter FIR p_x, the coefficients of filter FIR c_{ij} of each input variable, the number of hidden neurons h, the activation function of the hidden layer nodes $g(\cdot)$, and the network parameters $\mathbf{W}_I$, $\mathbf{b}_I$, $\mathbf{w}_O$, and b_O. The dimensionality of the model input is equal to $D = N$, and it is not dependent of the order p_x of the FIR filters. An example of FIR filter is the moving average filter, where the values of c_{ij} for all input variables $j = 1, \ldots, N$ in equation (4) are set to $c_{ij} = \frac{1}{p_x+1}$.

3.3. *ERN Model*

The ERN model, proposed by [11], is a form of recurrent neural network that includes a set of context units which are responsible for memorizing the past states of hidden units. In ERN, connections comming from the hidden layers through the context

units, are are fed back to the input layer. Thus, in the ERN model, the input layer is constituted by the model input and the feedback from the context units. Therefore, the ERN operates in a way where the output of the network depends on an aggregate of previous states of hidden units and the current input [32]. The ERN model is applied in several contexts, such as fault prediction in transmission lines [10], or chaotic time series prediction [6]. Comparing with other types of models, the most important advantage of the ERN model is to generate time-varying patterns [10,32]. The input $\mathbf{u}(k)$ of the ERN model in (1) is composed by $\mathbf{x}(k)$ and a collection of its past values, and a collection of past values of the hidden layer outputs $\mathbf{z}(k)$ (2), specifically $\mathbf{u}(k) = [\mathbf{x}(k), \ldots, \mathbf{x}(k-p_x), \mathbf{z}(k-1), \ldots, \mathbf{z}(k-p_z)]^T$.

The model parameters which should be defined/tuned for the ERN model are the number of hidden neurons h, the number of past values of input variables p_x, the number of past values of the hidden layer output p_z, as well the activation function of hidden layer nodes $g(\cdot)$, and the network parameters $\mathbf{W}_I$, $\mathbf{b}_I$, $\mathbf{w}_O$, and b_O. The size of the input space in the ERN model increases proportionally to p_x, p_z, and h. The input dimensionality is given by $D = N \cdot p_x + h \cdot p_z + 1$. Parameters, p_x, p_z, and h should be carefully selected to avoid a large value of D; large values of D can increase the training time and even lead to poor generalization in the prediction results [4].

3.4. *NARX Model*

The NARX models [20] are the non-linear generalization of the well-known ARX models, which constitute a standard tool in linear black-box model identification [21]. These models can represent a wide variety of nonlinear dynamic behaviors and are widely used in various applications, such as time series prediction [24], and forecasting of pollution levels [26]. The NARX model is a form of recurrent neural network in which connections from the output layers are fed back to the input layer. In this network, the input layer is constituted by the input variable $\mathbf{x}(k)$ and its past values, and the past values of the output. The NARX model, with one hidden layer, has the form of (1), where the input vector $\mathbf{u}(k)$ is composed by $\mathbf{x}(k)$ and a collection of its past values, as well as a collection of past values of the output $y(k)$, i.e. $\mathbf{u}(k) = [\mathbf{x}(k), \ldots, \mathbf{x}(k-p_x), y(k-1), \ldots, y(k-p_y)]$.

In the NARX model, three parameters should be defined/tuned: the number of hidden neurons h, the number of past values for input variables p_x, and the number of past values of output p_y, as well the activation function of hidden layer nodes $g(\cdot)$, and the network parameters $\mathbf{W}_I, \mathbf{b}_I, \mathbf{w}_O, b_O$. The input dimensionality D of the NARX model increases according with the parameters p_x and p_y, specifically $D = N \cdot p_x + p_y + 1$.

4. Methodology

This section describes the methodology used to set the parameters $\mathbf{W}_I$, $\mathbf{b}_I$, $\mathbf{w}_O$, b_O, and h of the DNN models, and the methodology used to select the best input variables and the respective architecture of each DNN model, which includes the parameters p_x for the FTLFN, the parameter p_x and the values of c_{ij} of the FIR filter for the DTLFN (in all experiments the moving average filter will be used as the FIR filter, where $c_{ij} = \frac{1}{p_x+1}$), the parameters p_x and p_z for the ERN, and p_x and p_y for the NARX model.

4.1. *Tunning of DNN models*

To set the values of the network parameters $\mathbf{W}_I$, $\mathbf{b}_I$, $\mathbf{w}_O$, b_O, each DNN model was trained by minimizing the mean square error (MSE) of all network output samples:

$$E_{mse}(y, y_d) = \frac{1}{M} \sum_{k=1}^{M} [y(k) - y_d(k)]^2 ,$$

where, $y(k)$ and $y_d(k)$ are the predicted and desired output of k-th input data sample. The models were trained using the Levenberg-Marquardt algorithm [13]. Another point is the determination of the number of hidden neurons h, which is essential for a correct application of any neural network model. As an example, a high number of neurons in the hidden layer generally promotes good performance during the training phase, but fails to predict correctly when unseen data is presented to the model, leading to a poor performance in the test phase, a problem known as over-fitting [4]. On the other hand, if an neural network model is trained with a too small number of hidden neurons, in some cases, during the training phase it cannot represent well the system being modeled, a problem which is called under-fitting. Hence, it is necessary to select the number of neurons which can represent well the system during the training phase and at the same time have a good performance for unseen data. In [2] it was suggested an approach to calculate the number of hidden neurons h, and it has shown to provide good results in the experimental part. According to this approach, h given by:

$$h = \frac{\epsilon M - C}{D + C + 1}, \tag{5}$$

where M is the number of training samples, C is the number of network outputs, and D is the number of inputs of the model. The parameter ϵ is the maximum error acceptable for the network test. In this work the value $\epsilon = 0.01$ was considered. As can be noted, the use of this equation does not consider only the structural aspects of network, but also it takes into consideration the maximum admissible error for test examples and the number of available samples.

To avoid overffiting, the early stopping strategy was employed as stop criterion. For this, the training data was divided in 70% for model training and 30% for model validation. The model parameters were learned using the 70% of training data and

the performance of the training was evaluated in the validation data. The training was stopped when the error in the validation data started to increase.

4.2. *Variable and Architecture Selection*

The parameters of each DNN model, refereed here as $\boldsymbol{\kappa}$ (for the FTLFN and DTLFN models $\boldsymbol{\kappa} = p_x$, for the ERN model $\boldsymbol{\kappa} = (p_x, p_z)$, and for the NARX model $\boldsymbol{\kappa} = (p_x, p_y)$) need to be properly selected. The values of $\boldsymbol{\kappa}$ are linked with the input dimensionality of the model: a large input space can increase the training time and even lead to poor generalization in the prediction results [4]. Moreover, as one of the objectives of this work is to select a number of input variables as small as possible, then the best subset of variables to predict the output will be selected together with the parameters $\boldsymbol{\kappa}$.

To evaluate the performance of a DNN model trained with parameter $\boldsymbol{\kappa}$, and with a subset $S \subseteq X$ of the set of input variables X, the C_p statistics [30,23] will be used as the performance criterion. The C_p statistics measures the trade-off between complexity and accuracy. The C_p statistics is defined as:

$$C_p = \frac{(\text{SSE of the DNN model trained with parameter } \boldsymbol{\kappa} \text{ and subset } S)}{(\text{SSE of the model trained with parameter } \boldsymbol{\kappa} \text{ and all available inputs } X)} - (D(X, \boldsymbol{\kappa}) - 2D(S, \boldsymbol{\kappa})), \tag{6}$$

where SSE is the sum of square error, and $D(X, \boldsymbol{\kappa})$ and $D(X, \boldsymbol{\kappa})$ are the input dimensionality of the DNN model trained with the set X and a subset $S \subseteq X$, respectivelly, and they are defined as:

- FTLFN: $D(X, \boldsymbol{\kappa}) = |X| \cdot p_x + 1$, and $D(S, \boldsymbol{\kappa}) = (|S| \cdot p_x + 1)$;
- DTLFN: $D(X, \boldsymbol{\kappa}) = |X|$, and $D(S, \boldsymbol{\kappa}) = |S|$;
- ERN: $D(X, \boldsymbol{\kappa}) = |X| \cdot p_x + h \cdot p_z + 1$, and $D(S, \boldsymbol{\kappa}) = |S| \cdot p_x + h \cdot p_z + 1$;
- NARX: $D(X, \boldsymbol{\kappa}) = |X| \cdot p_x + p_y + 1$, and $D(S, \boldsymbol{\kappa}) = |S| \cdot p_x + p_y + 1$.

According with the C_p statistics, the best models are the ones where C_p is closest to $D(S, \boldsymbol{\kappa})$ (i.e. those subset models that fall near the line $C_p = D(S, \boldsymbol{\kappa})$ in the C_p versus $D(S, \boldsymbol{\kappa})$ plot).

5. Experimental Results

This section provides the experimental results for the PAC and AS dosages prediction at the WTP of Ceará. For this purpose, two individual models were constructed to predict the PAC and AS dosages. The variables used as candidates for modeling the PAC and AS dosages were the pH and turbidity in the raw and coagulated water; Their labels and units used in the remaining of the section are described in Table 1.

Moreover, to select the best architecture of each DNN model and achieve the best solution in terms of model performance with least number of input variables, several DNN models with different combinations of the input variables have been evaluated,

Table 1. Variables and labels description.

Variables	Description
x_1	pH in raw water;
x_2	pH in coagulated water;
x_3	Turbidity in raw water [NTU]
x_4	Turbidity in coagulated water [NTU];
y_1	coagulant dosage of polyaluminum cloride (PAC) [mg/l];
y_2	coagulant dosage of polyaluminum sulfate (AS) [mg/l];

by following the methodology described in Section 4. The best architecture and the respective input variables for each model was selected based on the results of the C_p statistics. The subset S considered in the experimental results was all possible combinations of the input variables X, i.e. $\{x_1\}, \ldots, \{x_4\}, \{x_1, x_2\}, \ldots, \{x_1, x_4\}$, $\ldots, \{x_1, x_2, x_3, x_4\}$. Moreover, for the parameters of the FTLFN, DTLFN, ERN, and NARX models it was considered that $p_x = p_z = p$ and $p_x = p_y = p$, respectively.

The historical data used for modeling comprises 3 years of operational data, and it contains information about the coagulant dosage values (determined by jar tests) and parameters values from raw and coagulated water at every 2 [h], with a total of 13043 data samples. The models were trained using the first 2 years (8723 samples) of data and the remaining 1 year (4320 samples) was used as the test set. This 1 year test data set includes rainy and dray seasons. The performance on the 1 year test set will be measured using the root mean square error (RMSE) and the coefficient of determination R^2 between predicted and desired output.

5.1. *Prediction Results*

The selected variables for each DNN model and their respective parameters, based on the methodology described in Section 4.2, are summarized in Table 2.

Table 2. Mapping and evaluation of the best pair $(D(S, \boldsymbol{\kappa}), Cp)$ for the PAC (y_1) and AS (y_2) prediction.

	output/target: y_1				output/target: y_2			
Model	p	S	$(D(S, \boldsymbol{\kappa}), C_p)$	h	p	S	$(D(S, \boldsymbol{\kappa}), C_p)$	h
FTLFN	1	(x_1, x_3, x_4)	$(6, 5.04)$	6	3	(x_1, x_3, x_4)	$(12, 11.51)$	4
DTLFN	1	(x_1, x_3, x_4)	$(6, 5.49)$	6	1	(x_2, x_3, x_4)	$(6, 5.72)$	6
ERN	0	(x_1, x_2, x_3)	$(3, 3.08)$	9	0	(x_1, x_2, x_3)	$(3, 2.87)$	9
NARX	1	(x_1, x_2, x_4)	$(7, 6.10)$	5	1	(x_1, x_2, x_4)	$(7, 6.05)$	5

The values of p considered in all experiments ranged from $p = 0, \ldots, 6$. For PAC prediction and from Table 2, it is possible to note that the selected variables are different for each DNN model, and only the pH in raw water (x_1) is present in all models. By evaluating each DNN model on the 1 year test data set, with the

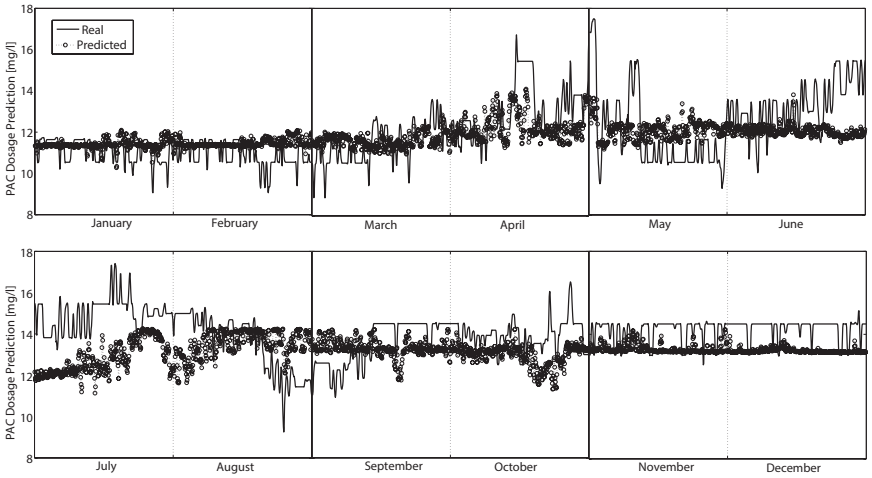

Figure 1. PAC prediction output (1-year dataset) with FTLFN model

parameters described in Table 2, it is possible to see the good performance results of the NARX model, which reached a coefficient of determination of $R^2 = 0.91$, and a RMSE of 0.27. The results of the other DNN models, indicated in Table 3,

Table 3. Results of all models for PAC (y_1) and AS (y_2) prediction on the test set

	y_1		y_2	
Model	RMSE	R^2	RMSE	R^2
FTLFN	1.18	0.51	2.16	0.72
DTLFN	1.26	0.36	1.84	0.48
ERN	0.95	0.50	1.26	0.60
NARX	**0.27**	**0.91**	**0.19**	**0.95**

reinforce that the NARX model provides the best results. From the results, it is possible to see that, to predict the PAC dosage only three sensors are necessary, the pH in the raw and coagulated water and the turbidity in the coagulated water. The input $\mathbf{u}(k)$, to predict the PAC dosage, for the NARX model is then composed by: $\mathbf{u}(k) = [x_1(k), x_2(k), x_4(k), x_1(k-1), x_2(k-1), x_4(k-1), y_1(k-1)]^T$. Figures 1-8, shows the output prediction of the PAC dosage in the 1-year test data set, showing that the NARX model having three input variables, the pH in raw water, pH in coagulated water and the turbidity in coagulated water is a good predictor for the PAC dosage values. The other models (FTLFN, DTLFN and ERN) have acceptable values in some months; However the quality of their results is not comparable with the much better results of the NARX model.

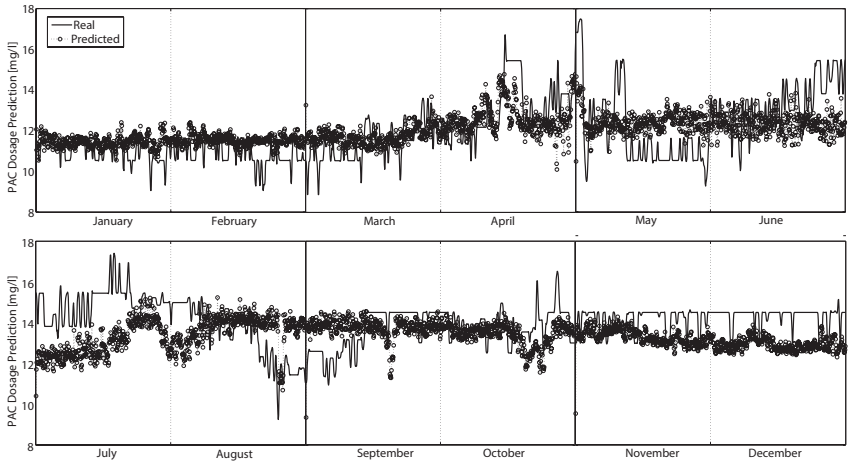

Figure 2. PAC prediction output (1-year dataset) with DTLFN model

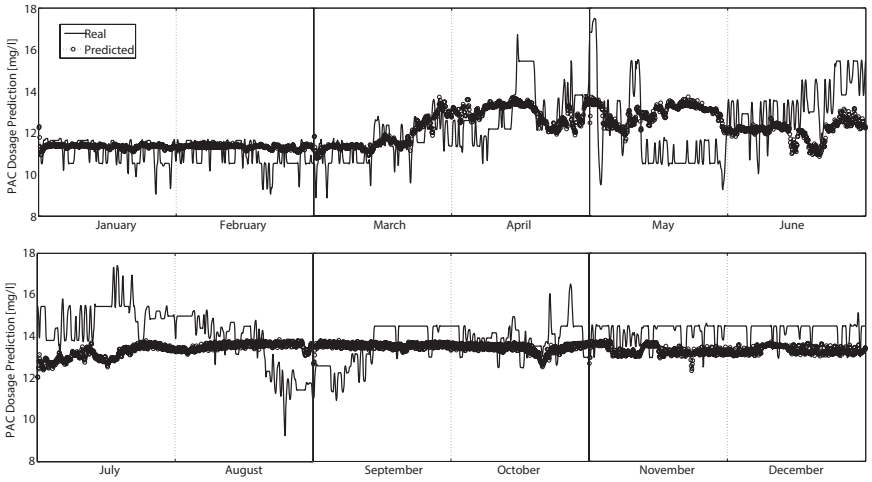

Figure 3. PAC prediction output (1-year dataset) with ERN model

In the case of AS prediction, each DNN model uses a different set of variables to build the model. However, excepted for the DTLFN model, all DNN models use the same variables used for the PAC prediction. According to Table 2, the NARX model exhibited the best prediction performance results in the 1-year test data set, with a large value of coefficient of determination of $R^2 = 0.95$, most notably considering that the test data set has 8192 samples. Figures 5-8, shows the prediction output of all DNN models, as can be noticed the NARX model has the best result.

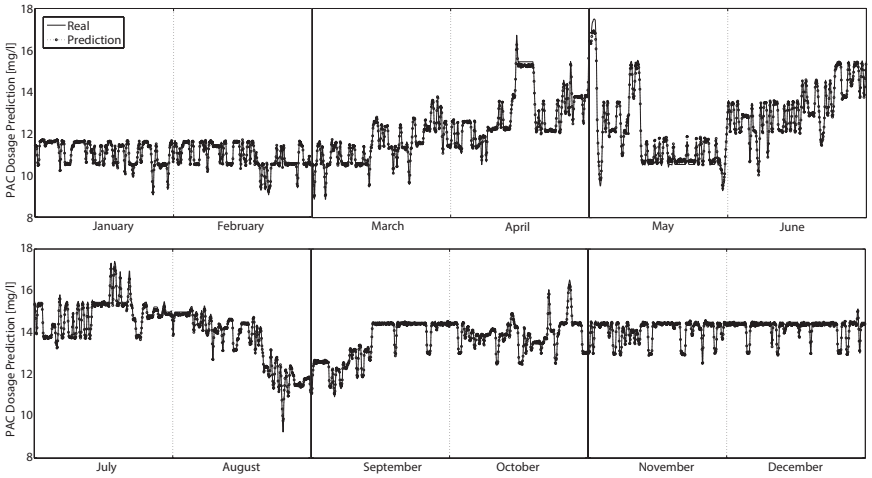

Figure 4. PAC prediction output (1-year dataset)with NARX model

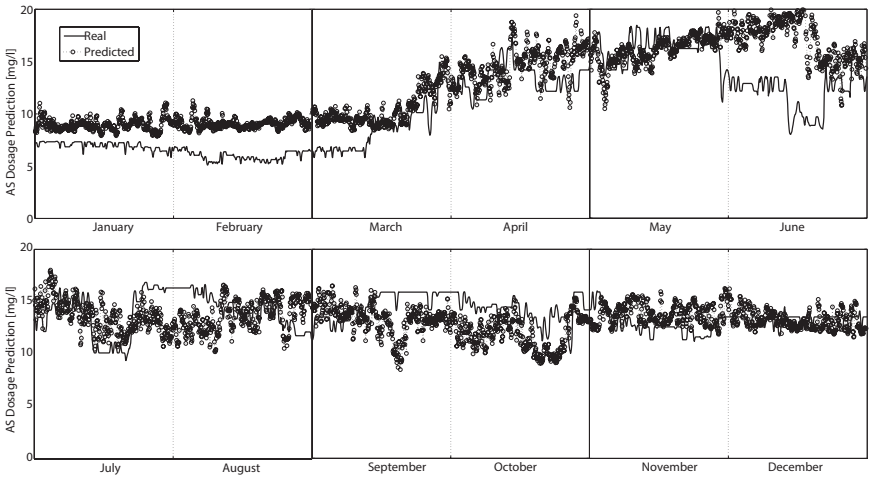

Figure 5. AS prediction output (1-year dataset) with FTLFN model

5.2. *Discussion*

From results, described in Section 5.1, the NARX model reached the best prediction performance for predicting the PAC and AS dosages. In the case of study presented here, it is possible to conclude that three sensors are necessary to provide the PAC and AS prediction. These sensors are the pH in raw and coagulated water and the turbidity in the coagulated water. Using only these sensors, the performance of automatic PAC and AS dosage are satisfactory, as can be seen by the value coefficient of determination ($R^2 = 0.91$ and $R^2 = 0.95$, for PAC and AS prediction,

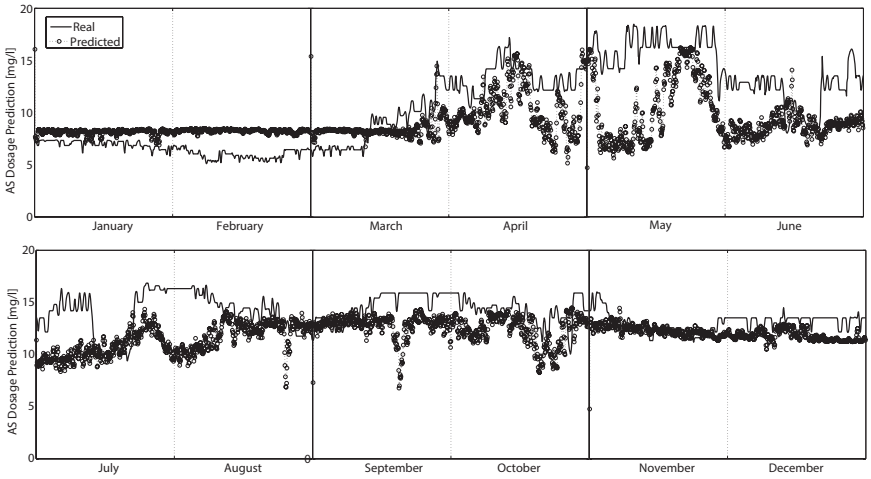

Figure 6. AS prediction output (1-year dataset) with DTLFN model

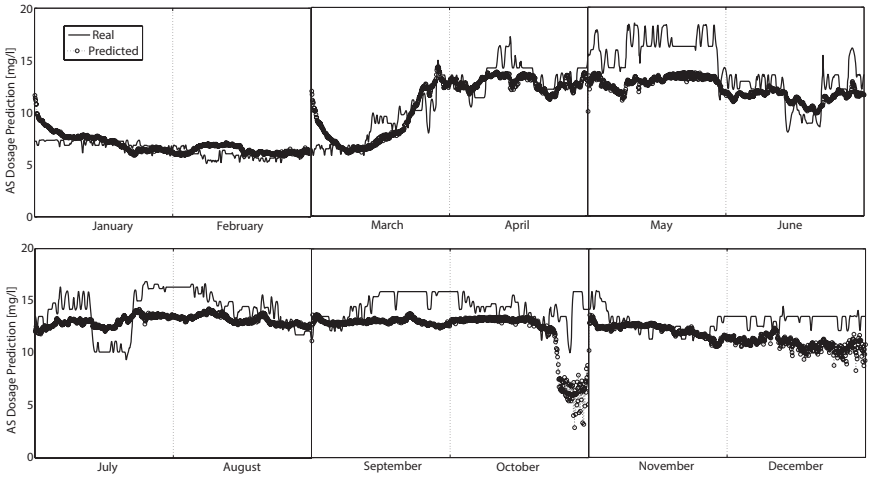

Figure 7. AS prediction output (1-year dataset) with ERN model

respectivelly) and the pictures of prediction in the test-set, that comprises 1-year of real operation.

Regarding the automatic coagulant determination in literature, the PAC prediction is the most studied case, see [12,16,22,31,15]. The comparison of this work with existing literature is done in Table 4. In Table 4, the input variables, the number of samples used for modeling, the data sample time, the model used for modeling, and the R^2 are shown.

As can be seen from Table 4, the work done in Wu et. al.,2008[31], uses only

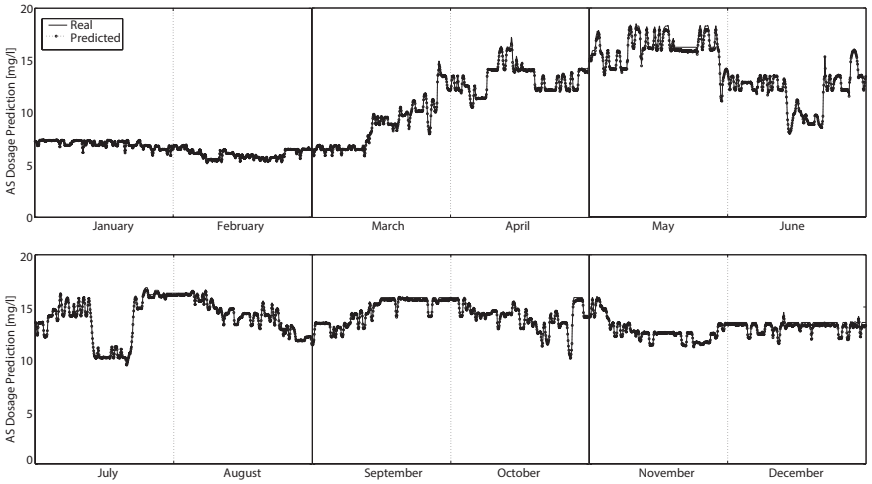

Figure 8. AS prediction output (1-year dataset)with NARX model

one sensor, the turbidity in raw water and the coagulant cosage of previous day, to predict the PAC dosage and it reached a $R^2 = 0.894$. However, the work presented here achieved a $R^2 = 0.91$, with the addition of two sensors, the pH in raw and coagulated water, which in present in most of WTP plants. Other relevant work is done in Maier et. al. 2004[22], which reached a $R^2 = 0.94$, for PAC prediction. However, in Maier et. al. 2004[22], there are a need to use a larger number of sensors when compared with the work proposed in this paper. The other works[12,16,15] have inferior prediction performance or cannot be compared due the lack of data to do it.

Table 4. Summary about the input variables, number of samples (#Samples) and data sampling for PAC prediction described in literature. The terms RW, CW and TW, are the acronym for raw water, coagulated and treated water, respectivelly and ANN and ANFIS are the acronym for artificial neural network and adaptive neuro-fuzzy inference system, respectivelly.

Ref.	Inputs	#Samples	Sampling	Model	R^2
This work	pH-RW, pH-CW, Turbidity-CW	21766	2[h]	NARX	0.92
Gagnon et. al., 1997 [12]	Turbidity-RW, pH-RW, Temperature-RW, Conductivity-RW	332.920	5 [min]	ANN	–
Joo et. al., 2000[16]	Turbidity-RW, pH-RW, Temperature-RW, Alkalinity-RW	731	24 [h]	ANN	–
Maier et. al. 2004[22]	Turbidity-RW, pH-RW, Color-RW, UVA-254-RW, DOC-RW, Turbidity-TW, Color-TW, UVA-254-TW	202	-	ANN	0.94
Wu et. al.,2008[31]	Turbidity-TW, Coagulant Dosage of previous day	819	24 [h]	ANN	0.894
Heddam et. al., 2011[15]	Turbidity-RW, pH-RW, Temperature-RW, UVA-254-RW, DO-RW	725	24 [h]	ANFIS	0.84

From the results of achieved in the last section,

6. Conclusion

The goal of this work was to establish a regression model which uses a small number of input variables to correctly predict the dosages of PAC and AS in a real WTP in the State of Ceará, Brazil. For this purpose, several dynamic neural network models (DNN) were evaluated in different settings (i.e. with different parameters and different combinations of input variables) and tested in a 1-year test data set, which covers summer and drier seasons.

The DNN models evaluated were the FTLFN, DTLFN, ERN and NARX models. The best results for the PAC and AS dosages prediction were reached by the NARX model. The results demonstrated that only three sensors are necessary: the pH in raw and coagulated water, and the turbidity in the coagulated water. The input $\mathbf{u}(k)$, to predict the PAC and AS dosages with the NARX model are composed by: $\mathbf{u}(k) = [x_1(k), x_2(k), x_4(k), x_1(k-1), x_2(k-1), x_4(k-1), y_1(k-1)]^T$. It has been shown that the NARX model has a great capability to predict accurately the PAC and AS dosages values in real time, with coefficients of determination of $R^2 = 0.91$ and $R^2 = 0.95$, respectively.

Bibliography

1. Alarcon-Aquino, V., & Barria, J. A. (2006). Multiresolution fir neural-network-based learning algorithm applied to network traffic prediction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *36*, 208–220.
2. Baum, E. B., & Haussler, D. (1989). What size net gives valid generalization? *Neural Computation*, *1*, 151–160.
3. Belanche, L., Valdés, J. J., Comas, J., Roda, I. R., & Poch, M. (2000). Prediction of the bulking phenomenon in wastewater treatment plants. *Artificial Intelligence in Engineering*, *14*, 307 – 317.
4. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. (1st ed.). Springer.
5. Brabtby, J. (2006). *Coagulation and Flocculation in Water and Wastewater Treatment*. IWA Publishing.
6. Chandra, R., & Zhang, M. (2012). Cooperative coevolution of elman recurrent neural networks for chaotic time series prediction. *Neurocomputing*, *86*, 116–123.
7. Chiewchanwattana, S., Lursinsap, C., & Chu, C. H. H. (2002). Time-series data prediction based on reconstruction of missing samples and selective ensembling of fir neural networks. In *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP '02.* (pp. 2152–2156). volume 5.
8. Dellana, S. A., & West, D. (2009). Predictive modeling for wastewater applications: Linear and nonlinear approaches. *Environmental Modelling & Software*, *24*, 96 – 106.
9. Dentel, S. K., Thomas, A. V., & Kingery, K. M. (1989). Evaluation of the streaming current detector-i. use in jar tests. *Water Research*, *23*, 413–421.
10. Ekici, S., Yildirim, S., & Poyraz, M. (2009). A transmission line fault locator based on elman recurrent networks. *Applied Soft Computing*, *9*, 341 – 347.
11. Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, *5*, 195–225.
12. Gagnon, C., Grandjean, B. P., & Thibault, J. (1997). Modelling of coagulant dosage in a water treatment plant. *Artificial Intelligence in Engineering*, *11*, 401 – 404.

13. Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks*, *5*, 989–993.
14. Haykin, S. (1999). *Neural networks: a comprehensive foundation*. Prentice Hall.
15. Heddam, S., Bermad, A., & Dechemi, N. (2011). Applications of radial-basis function and generalized regression neural networks for modeling of coagulant dosage in a drinking water-treatment plant: Comparative study. *Journal of Environmental Engineering*, *137*, 1–6.
16. Joo, D.-S., Choi, D.-J., & Park, H. (2000). The effects of data preprocessing in the determination of coagulant dosing rate. *Water Research*, *34*, 3295 – 3302.
17. Heddam, S. & Noureddine, D., (2015). A new approach based on the dynamic evolving neural-fuzzy inference system (DENFIS) for modelling coagulant dosage (Dos): case study of water treatment plant of Algeria. *Desalination and Water Treatment* , *53*, 1045 – 1053.
18. Kim, H., & Shin, K. (2007). A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets. *Applied Soft Computing*, *7*, 569 – 576.
19. Kimura, M., Matsui, Y., Kondo, K., Ishikawa, T. B., Matsushita, T., & Shirasaki, N. (2013). Minimizing residual aluminum concentration in treated water by tailoring properties of polyaluminum coagulants. *Water Research*, *47*, 2075–2084.
20. Leontaritis, I. J., & Billings, S. A. (1985). Input-output parametric models for non-linear systems part i: deterministic non-linear systems. *International Journal of Control*, *41*, 303–328.
21. Ljung, L. (1999). *System Identification: Theory for the User*. Prentice Hall.
22. Maier, H. R., Morgan, N., & Chow, C. W. (2004). Use of artificial neural networks for predicting optimal alum doses and treated water quality parameters. *Environmental Modelling & Software*, *19*, 485 – 494.
23. Mallows, C. L. (1973). Comments on cp. *Technometrics*, *15*, 661–675.
24. Menezes, J. M., Jr., & Barreto, G. A. (2008). Long-term time series prediction with the NARX network: An empirical evaluation. *Neurocomputing*, *71*, 3335–3343.
25. Morfesis, A., Jacobson, A. M., Frollini, R., Helgeson, M., Billica, J., & Gertig, K. R. (2009). Role of zeta (ζ) potential in the optimization of water treatment facility operations. *Industrial & Engineering Chemistry Research*, *48*, 2305–2308.
26. Pisoni, E., Farina, M., Carnevale, C., & Piroddi, L. (2009). Forecasting peak air pollution levels using NARX models. *Engineering Applications of Artificial Intelligence*, *22*, 593–602.
27. Valentin, N., Denoeux, T., & Fotoohi, F. (1999). Modelling of coagulant dosage in a water treatment plant.
28. Waibel, A., Hazanawa, T., Hinton, G., Shikano, K., & Lang, K. J. (1989). Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech and Signal Processing*, *37*, 328–339.
29. Wan, E. A. (1990). Temporal backpropagation for fir neural networks. In *IJCNN International Joint Conference on Neural Networks, 1990.* (pp. 575–580).
30. Warne, K., Prasad, G., Rezvani, S., & Maguire, L. (2004). Statistical and computational intelligence techniques for inferential model development: a comparative evaluation and a novel proposition for fusion. *Engineering Applications of Artificial Intelligence*, *17*, 871–885.
31. Wu, G.-D., & Lo, S.-L. (2008). Predicting real-time coagulant dosage in water treatment by artificial neural networks and adaptive network-based fuzzy inference system. *Enginnering Applications of Artificial Intelligence*, *21*, 1189–1195.
32. Şeker, S., Ayaz, E., & Türkcan, E. (2003). Elman's recurrent neural network appli-

cations to condition monitoring in nuclear power plant and rotating machinery. *Engineering Applications of Artificial Intelligence*, *16*, 647 – 656.
33. Sibiya, S. M., (2014). Evaluation of the Streaming Current Detector (SCD) for Coagulation Control. *Procedia Engineering*, *70*, 1211 – 1220.