OBSERVING OF pH FOR TITRATION PROCESS WITH HYBRID NEURAL NETWORK STRUCTURE

Shebel Asad
Assistant professor, Mechatronics Engineering Division, Faculty of Engineering Technology,
Al-Balqa Applied University, P.O Box 15008, Amman (11134), Jordan
Tel.: +962 777745553, +962 64790333
E-mail: shebel_asad@hotmail.com , www.fet.edu.jo

ABSTRACT
This paper presents the application of a numerical pH observer integrated into titration process as an industrial replacement of real hardware electrodes to measure pH. The proposed observer is designed with LabView and MatLab. First, two kinds of neural networks NN - multilayer perceptron network MLP and radial basis function network RBF- are used, separately, to design pH observers, then to ensure the accuracy and modify the response, a hybrid neural network is developed, it accomplishes the best features found with both MLPNN and RBFNN. The split-sample method is implemented to select the optimal NN structure. Results are presented and compared in presence of measurement noise (uncertainties in base flow in and temperature variation).

Keywords: *Hybrid neural network, pH measurement, RBFNN, MLPNN, LabView.*

1. INTRODUCTION
This work aims to design, numerically, a hybrid-based neural network pH observer to be used in titration processes. This approach enables the industrial replacement of the real hardware measurement pH electrodes, which is a conventional method to measure pH, with a hybrid neural network-based pH observer. The proposed intelligent-based observing method add more and more accuracy to the measurement techniques and add also more ranges to the control systems.

The hybrid NN has been developed by switching between both multi layers perceptron MLPNN and radial basis function RBFNN. This approach could invest the points of strength with each (MLPNN and RBFNN). The work is designed with LabView and MatLab. The proposed hybrid model could justify the higher accuracy in observing the pH values and the speed enquiries for processing. An experimental data base has been used to train the nets and find out the optimal NN structure using split-sample method. Numerical models and results have been obtained and discussed.

2. PROBLEM DESCRIPTION AND FORWARD PROBLEM
The geometry of the problem is summarized in figures 1 and 2. pH measurement is unlike most of the on-line measurements in the aspect that it cannot be “installed and forgotten”. It requires constant maintenance including cleaning, calibration and fault diagnosis and even if the maintenance is performed to the last detail, the pH probe has a process dependent life-span after which it has to be replaced [1, 2].

A pH measurement loop is made up of three components, the pH sensor, which includes a measuring electrode, a reference electrode, and a temperature sensor; a preamplifier; and an analyzer or transmitter. A pH measurement loop is essentially, as represented in Fig.1. A battery where the positive terminal is the measuring electrode and the negative terminal is the reference electrode. The measuring electrode, which is sensitive to the hydrogen ion, develops a potential (voltage) directly related to the hydrogen ion concentration of the solution. The reference electrode provides a stable potential against which the measuring electrode can be compared.

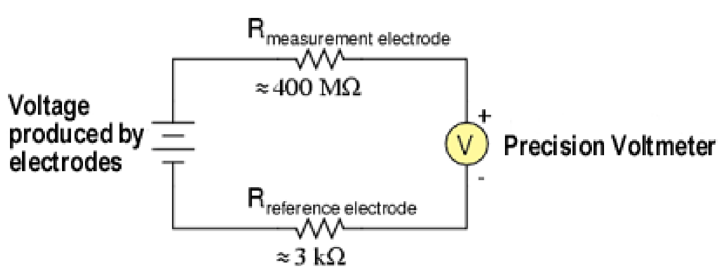

Figure 1. The equivalent circuit of a pH measurement loop

Because pH measurement is a logarithmic representation of ion concentration, there is an incredible range of process conditions represented in the seemingly simple 0-14 pH scale. Also, due to the nonlinear nature of the logarithmic scale, a change of 1 pH at the top end (say, from 12 to 13 pH) does not represent the same quantity of chemical activity change as a change of 1 pH at the bottom end (say, from 2 to 3 pH). Control system engineers and technicians must be aware of this dynamic if there is to be any hope of controlling process pH at a stable value.
Keep in mind, application requirements should be carefully considered when choosing a pH electrode. Accurate pH measurement and the resulting precise control that it can allow, can go a long way toward process optimization and result in increased product quality and consistency. Accurate, stable pH measurement also controls and often lowers chemical usage, minimizing system maintenance and expense [3, 4].

2.1 Evaluation of pH value with Titration Processes

While pH can be measured by color changes in certain chemical powders, continuous process monitoring and control of pH requires a more sophisticated approach. The most common approach is the use of a specially-prepared electrode designed to allow hydrogen ions in the solution to migrate through a selective barrier, producing a measurable potential (voltage) difference proportional to the solution's pH:
What is important to understand is that these two electrodes generate a voltage directly proportional to the pH of the solution. At a pH of 7 (neutral), the electrodes will produce 0 volts between them. At a low pH (acid) a voltage will be developed of one polarity, and at a high pH (caustic) a voltage will be developed of the opposite polarity [1-4].
An unfortunate design constraint of pH electrodes is that one of them (called the measurement electrode, shown in Fig.2) must be constructed of special glass to create the ion-selective barrier needed to screen out hydrogen ions from all the other ions floating around in the solution. This glass is chemically doped with lithium ions, which is what makes it react electrochemically to hydrogen ions. Of course, glass is not exactly what you would call a "conductor;" rather, it is an extremely good insulator. This presents a major problem if our intent is to measure voltage between the two electrodes. The circuit path from one electrode contact, through the glass barrier, through the solution, to the other electrode, and back through the other electrode's contact, is one of extremely high resistance.

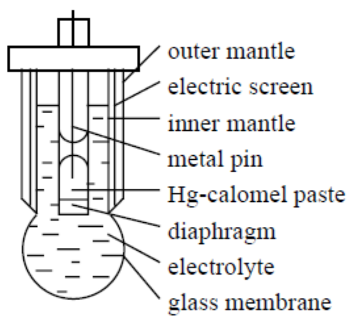

Figure 2. The construction of the glass electrode.

The other electrode (called the reference electrode) is made from a chemical solution of neutral (7) pH buffer solution (usually potassium chloride) allowed to exchange ions with the process solution through a porous separator, forming a relatively low resistance connection to the test liquid. At first, one might be inclined to ask: why not just dip a metal wire into the solution to get an electrical connection to the liquid? The reason this will not work is because metals tend to be highly reactive in ionic solutions and can produce a significant voltage across the interface of metal to-liquid contact. The use of a wet chemical interface with the measured solution is necessary to avoid creating such a voltage, which of course would be falsely interpreted by any measuring device as being indicative of pH.
All pH electrodes have a finite life, and that lifespan depends greatly on the type and severity of service. In some applications, a pH electrode life of one month may be considered long, and in other applications the same electrode(s) may be expected to last for over a year.

2.2 Proposed NN-Based pH Observer

NNs are constituted of interconnected processing elements, called neurons. They can be used for complex and non linear functions modeling. In this paper, two kinds of NN are utilized in the aim to create such observer (MLPNN and RBFNN).
To override the shortages and industrial drawbacks of the hardware pH measurement tools, a numerical pH observer has been proposed and validated. The idea is to design a hybrid ANN-based observer that combines the features of

MLPNN and RBFNN. The two proposed nets have been designed with MatLab as M-files and then tested. An experimental data base (200 samples) has been created to train the nets. Then, finally, a hybrid structure has been designed with LabView, tested and validated, as will be seen later.

2.2.1 MLPNN- Based pH Observer

The structure of the MLPNN-based pH observer is as seen in Fig.3,

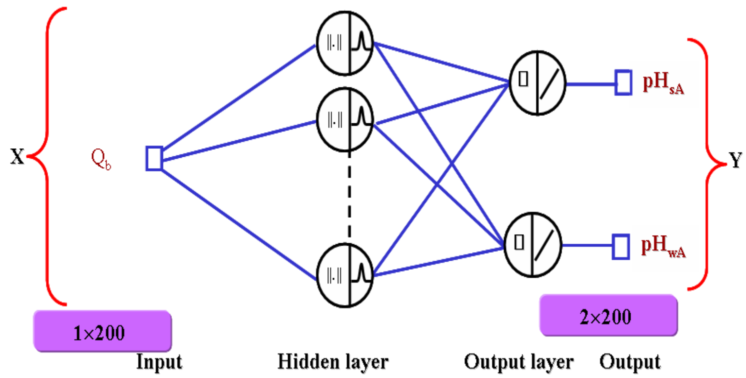

Figure 3. Structure of the proposed MLPNN observer

In Fig. 3, the NN has 1 input ($Q_b(\Delta\theta)$) represents the measured base stream as a function of temperature variation ($\Delta\theta$), and 2 outputs pH_{sA} and pH_{wA}. Where pH_{sA} and pH_{wA} are the pH at strong and weak acid, respectively.
The structure used in this work for this type of application constitutes of one hidden layer with a hyperbolic tangent activation function and output layer with linear function.
For such problems, where the MLP-NN is proposed to observe the pH value, there is no general method to fix the architecture of the network (number of neurons in the hidden layer) [5, 6].

2.2.2 RBFNN-Based pH Observer

RBFNN are generally considered as a smooth transition between Fuzzy inference Systems FIS and Neural Networks NNs. Structurally, a RBFNN is composed of receptive units (neurons) which act as the operators providing the information about the class to which the input signal belongs. If the aggregation method, number of receptive units in the hidden layer and the constant terms are equal to those of a FIS, then there exists a functional equivalence between RBFNN and FIS [7, 8].
The architectural view of the RBFNN-based pH observer is very similar to that of an ordinary feedforward neural network. The neurons in the hidden layer contain Gaussian transfer functions whose outputs are inversely proportional to the distance from the center of the neuron.
The hidden neurons of a RBFNN possess basis functions to characterize the partitions of the input space. Each neuron in the hidden layer provides a degree of membership value for the input pattern with respect to the basis vector of the receptive unit itself. The output layer is comprised of linear neurons. NN interpretation makes RBFNN useful in incorporating the mathematical tractability, especially in the sense of propagating the error back through the network, while the FIS interpretation enables the incorporation of the expert knowledge into the training procedure. The latter is of particular importance in assigning the initial value of the network's adjustable parameter vector to a vector that is to be sought iteratively. Expectedly, this results in faster convergence in parameter space.

2.2.3 Training of MLP and RBF ANNs

The MLP and RBF artificial neural networks ANNs are trained using a supervised training rule which attempts to minimize the error between the network and the target output patterns. If target outputs are not required for training, then the learning rule is unsupervised and the network extracts its own features from the training set. The Kohonen neural network, which is used to classify input vectors, learns using an unsupervised rule. For such applications based on an identified model, the neural network is typically trained using a supervised learning procedure.
The purpose of the training algorithm is to enable the ANN to represent a mapping which describes the I/O behavior of a non-linear system. To achieve this, the algorithm attempts to minimize an objective function by adjusting the ANN weight parameters. The objective function is a measure of how well the ANN fits a set of I/O training data patterns which the system has produced [6-8].
The backpropagation BP algorithm, derived by WERBOS (1974) and rediscovered by RUMELHART *et al.* (1986), was used in is work to train MLPNN and RBFNN. BP was a substantial theoretical advance which fuelled the

resurgence of interest in neural networks. While BP has disadvantages, such as slow convergence, it remains a popular training algorithm.

For simplicity the BP algorithm outlined here is for a three layer neural network as the extension of the algorithm to additional layers is straightforward. Fig.4 shows such a network where xi is the ith network input, hj is the output, or activation, of the jth hidden node, $\hat{y}_k$ is the kth observed network output (pH), w_{1ij} is the weight connecting the i_{th} input node to the j_{th} hidden layer node, w_{2jk} is the weight connecting the j_{th} hidden layer node to the k_{th} output node, n_i is the number of network inputs, n_h is the number of hidden layer nodes, n_o is the number of network outputs.

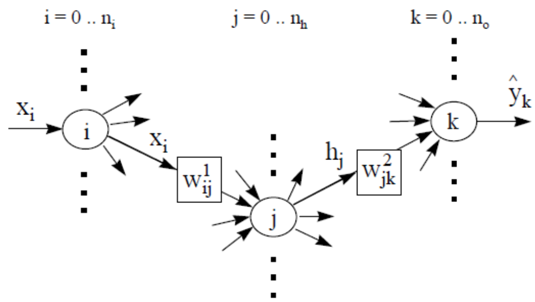

Figure 4. Three layers MLP network

In this case, we are going to study certain number of neuronal architectures. For each architecture, we do different initializations of synaptic parameters to assure that the training of the ANN converges towards the total minimum of the error criterion. For each structure, we calculate the mean square error MSE in the training and validation data bases. Then, the adequate structure that we are concerned is the structure which has the least square error in the validation base (in our case is equal 10^{-06}).

For such application, we aimed to vary the number of neurons in the hidden layer from 1 to 20 neurons, as presented above in the x-axis for the relation between the MSE via the number of neurons in the hidden layer. And for each structure, three different initialisations of synaptic parameters had been carried out. The training had been done using Levenberg-Marquardt algorithm [8].

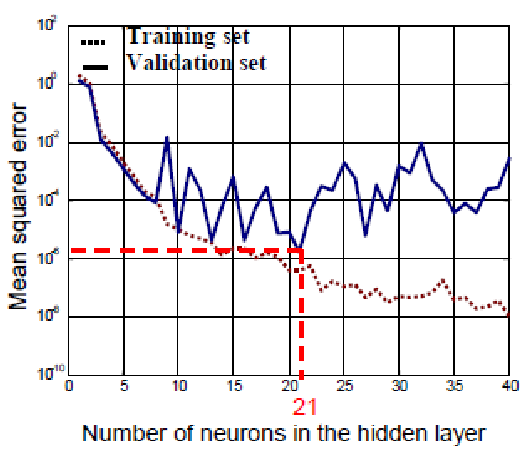

Figure 5. The evaluation of the minimal MSE in the validation data base (blue color) (in our case is equal 21).

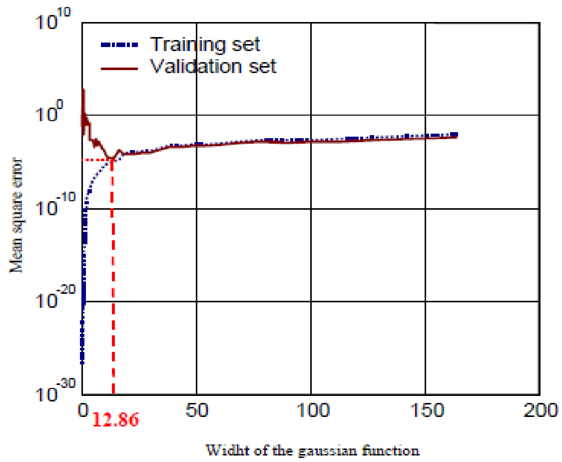

Figure 6. MSE on the training and validation sets (RBFNN)

2.2.4 Comparison of RBF and MLP ANNs

From the obtained results, it's well seen that it is much faster to train an RBFNN than a MLPNN, and thus, it is more convenient to employ the RBFNN to establish an appropriate ANN NARX model structure and data sample time and to investigate methods of dead time compensation. The design of these model attributes should be independent of the type of neural network used to perform the nonlinear mapping, and thus, should be applicable to the MLPNN. To verify this, and to compare the relative performance of the RBF and MLPNNs, the two ANNs and a Spread Encoded MLPNN were trained using the same RAS and the PI was evaluated (Fig.7). Preliminary experiments suggested 30 and 60 were appropriate choices for the number of hidden layer nodes for the MLP and Spread Encoded MLP respectively.

There is little to choose between the RBFNN and conventionally encoded MLPNN, but the Spread Encoded MLP gives more accurate predictions for prediction horizons greater than one sample interval [7, 8].

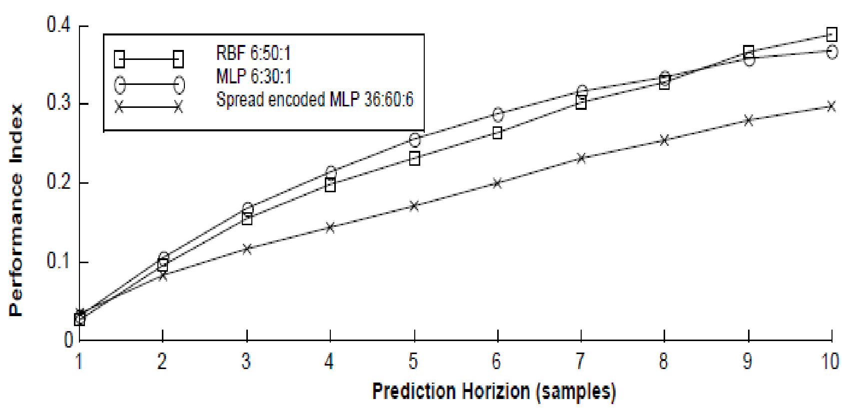

Figure 7. Comparison of the multi-step ahead prediction accuracy of a RBF, a MLP and a SE MLP ANN.

Finally, the capacity of the net will be tested, with respect to extracted experimental data base, to find the observed pH values that corresponded to the least error between the estimated and ideal ones for different 200 values of input vectors. The utilised examples had been subdivided into three stages (training (100), validation (68) and testing (32)) as performed by split-sample method.

2.3 Hybrid NN-Based pH Observer

The created hybrid NN-based pH model is designed with LabView, as shown in Fig. 8

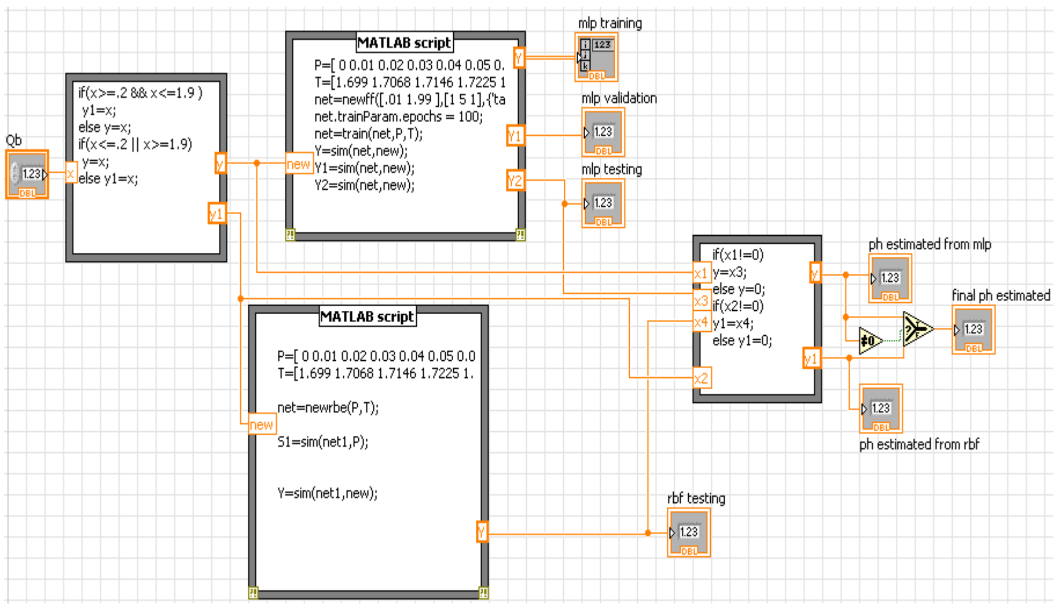

Figure 8. Hybrid structure model

Fig.8 shows the Hybrid observer which is designed with using of different advanced tasks in ANN, as will explained later. The designed observer aims to achieve high accuracy, hardness and treat the nonlinearity of titration curve.
Hybrid observer was created by using two neural network scripted M-files (RBF and MLP), the RBFNN was proposed in when the titration process passes into nonlinear region where the fast variation in pH value occurred (region 2 in Fig.9). While, the MLPNN was selected to match the linear regions proprieties in the titration process (1 and 3, Fig.9).

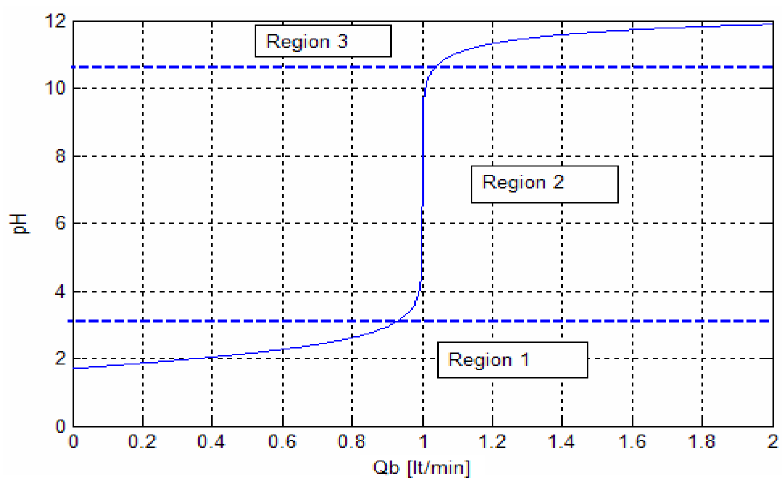

Figure 9. Titration curve regions

The created model with LabView, shown in Fig.8, consists mainly of two stages: the first stage is the logic operation that works to divide the input into three limited regions; the first is nonlinear region [0.8-1.2], the second two regions are linear ([0-0.8] and [1.2- 2]).
The main structural difference between the two proposed nets is the activation function to be used. In MLPNN, the function is a "tansig" while in RBFNN the function is "gauss" in latest layer. This difference makes the RBFNN better than MLPNN in dealing with nonlinear regions in the titration process.

3. RESULTS

The testing data base is of different values than the precedent ones (training and validation data base). The error between the real and observed pH values is defined for each parameter by the relative error RE(pH), illustrated in Fig. 10,

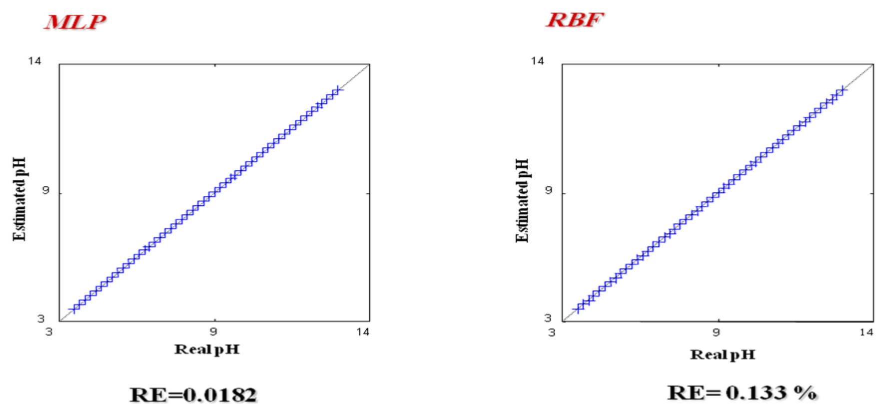

Figure 10. The computed relative error RE for both RBFNN and MLPNN.

Fig.10 shows the computed relative error RE at both nets. It seems that the RE produced by MLPNN is 1.82% which is larger than that of RBFNN (0.133%). That's why, the MLPNN is recommended to be used with linear regions in the titration process, and the RBFNN with nonlinear region, as will be seen later.

Using the compatibility features found in both LabView and MatLab (M-files), the model has been validated and the results have been obtained.

Fig.11 shows the performance of the proposed hybrid net, where it can be seen that the allowance of the sum squared error SSE has been reached (10^{-14}) with 11 epochs. So the both the accuracy and the speed enquiries have been achieved.

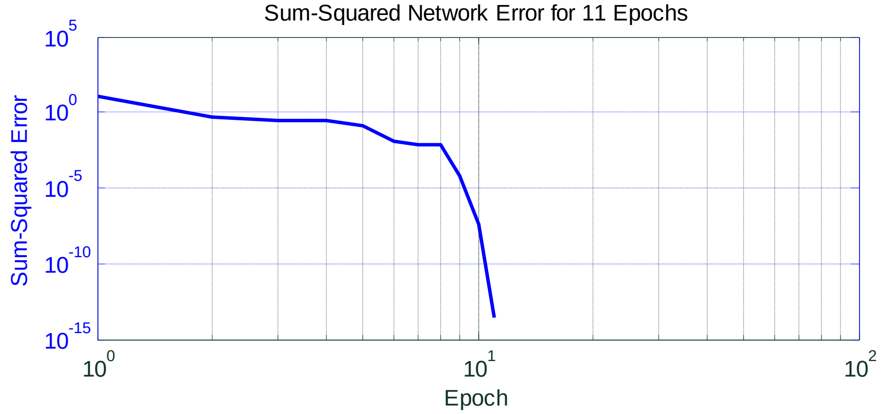

Figure.11 The SSE plot for the performance of the hybrid NN-based observer

4. CONCLUSION

Two kinds of neural inverse models (MLP and RBF) are developed to simultaneously observe the pH value for a titration process. Different input parameters (base flow in) and temperature variation are considered to train, test and validate the designed models. The models are able to generalize and show a good observing accuracy in presence of noise.

In presence of noise, the relative errors of the observed pH values have been computed. Then, a hybrid NN-based structure has been developed. The obtained results ensure the higher accuracy and rapidity to find the optimal structure. The obtained performance is modified when switching between the two nets (MLPNN and RBFNN). These results were obtained for a set of readings containing 200 samples.

With this work, the industrial costs could be reduced when replacing the real hardware with numerical hybrid structure connected to the base stream (flow transmitter), and the size could be also reduced. So, the work could match the commercial benefits, when realized.

5. REFERENCES

[1]. McMillan, Gregory K. (1994). pH Measurement and Control; Second Edition. Instrument Society of America.
[2]. Gadewar, S. B., Doherty, M. F., Malone, M. F., (2001). A systematic method for reaction invariants and mole balances for complex chemistries. Comp. & Chem. Eng. (25):1199-121.
[3]. Wright, R. A., Kravaris, C., (2001). On-line identification and nonlinear control of an industrial pH process. J. Proc. Cont. (11): 361-374.
[4]. Gregory K. McMillan and Cameron (2000). Advanced pH Measurement and Control. 3rd edition. ISA.
[5]. H. Selcuk Noagy, (2009). Prediction of Internal Temperature in Three-Phase Induction Motors with ANN. European Transactions on Electrical Power. Johan Wiley and Sons.
[6]. M. T. Hagan and M. B. Menhaj, (1994). Training Feedforward Networks with the Marquardt Algorithm. IEEE Trans. On Neural Networks. 5(6):989-993.
[7]. Nikhil, Bestamin Özkaya, Ari Visa, Chiu-Yue Lin, Jaakko A. Puhakka, and Olli Yli-Harja, (2008). An Artificial Neural Network Based Model for Predicting H2 Production Rates in a Sucrose-Based Bioreactor System. Proceedings of world academy of science. Engineering and technology. (27). ISSN 1307-6884.
[8]. Evern Guner, (2003). Adaptive Neuro Fuzzy Inference System Applications in Chemical Processes. A Thesis submitted to the graduate school of natural and applied sciences of the Middle East Technical University. In partial fulfillment of the requirement for the degree of Master of Science in the department of Chemical Engineering.