International Conference on Control, Instrumentation and Mechatronics Engineering (CIM'07),
Johor Bahru, Johor, Malaysia, May 28-29,2007

Modeling of real pH Neutralization Process using Multiple Neural Networks (MNN) Combination Technique

Zainal Ahmad* and Fairuoze Roslin
Process Control Group,
School of Chemical Engineering, Engineering Campus, University Sains Malaysia (USM), Seri Ampangan, 14300, Nibong Tebal, Penang, Malaysia
Tel : +604-593 7788 ext 6462, Fax : +604 594 1013
Email: chzahmad@eng.usm.my, fairuo_ze@yahoo.com

Abstract

Combining multiple neural networks appears to be a very promising approach in improving neural network generalisation since it is very difficult, if not impossible, to develop a perfect single neural network (SNN) especially when dealing with a real time data. Therefore, in this paper, two feedforward neural networks model technique are developed to predict the performance of a pH neutralization process, which uses a sulphuric acid as the acidic stream and sodium hydroxide aques as the bes stream. The technique involves combining multiple neural networks (MNN) and single neural network (SNN). The Levenberg–Marquardt (LM) optimization technique was employed for training the NN for both techniques. Application results demonstrate that the proposed multiple neural networks (MNN) combination techniques significantly improve model generalisation compared to single neural network (SNN) models.

Keywords: Neural networks, Multiple Neural Networks, simple averaging, nonlinear process modeling.

1. Introduction

The pH control is very important in many processes. For examples, in wastewater treatment plant, the cell growth rate and the accurate stabilization of pH at an optimal level often determines the efficiency of the bioprocess. The regulation and control of a pH process is a typical problem found in a variety of industries including wastewater treatment, pharmaceuticals, biotechnology and chemical processing. It is a nontrivial task arising from the nonlinearity of the titration process. Therefore, controlling the pH at certain region or set point is very important. On the other hand, in chemical processes, pH neutralization is not easy to control due to the fast and quite complicated reaction [1,2]. In terms of modeling, one of the disadvantages of pH neutralization is the difficulty of obtaining a rigorous mechanistic model of the process, which accounts for several important operating factors such as the flow rate of the influent stream, the flow rate of the titrating stream, the concentration of the influent stream, the concentration of the titrating stream, the concentration of the acid solution, and the volume of the mixture in the CSTR [3]. This is particularly true when knowledge about the process is initially vague or if the process is so complex that the resulting equations cannot be solved. Therefore modeling the pH is very challenging and a neural network is one of the options.

Process modeling is an area where neural networks configurations and structures have been considered as alternative modeling techniques, particularly in cases where reliable mechanistic models cannot be obtained [4–9] where this is due to the complexity and difficulty in control, the model based control is come to the picture. As mention in [1], to be successful in implementing the control strategy for this system, the pH control system must contain two main features: (i) reliable estimation of the process nonlinearity and (ii) a nonlinear compensation and control. In this aspect the neural networks capabilities are utilized.

Why neural network? Artificial neural networks have been shown to be able to approximate any continuous non-linear functions and have been used to build data base empirical models for non-linear processes [10]. Hence what is a neural network? According to [11].

* corresponding authors

'A neural network is a massive parallel-distributed processor that has a natural capability for storing experiential knowledge and making it available for use. It resembles the brain in two respects knowledge is acquired by the networks through a learning process. Interneuron connection strengths known as synaptic weights are used to store the knowledge'

Furthermore, the main advantage of neural network based process models is that they are easy to build. This feature is particularly useful when modelling complicated processes where detailed mechanistic models are difficult to develop. However a critical shortcoming of neural networks is that they often lack robustness unless a proper network training and validation procedure is used. Robustness of the model can be defined as one of the baseline to judge the performance of the neural network models and it is really related to the learning or training classes as what Bishop [12] described:

'The importance of neural networks in this context is that they offer very special powerful and very general framework for representing non-linear mappings from several input variables to several output variables, where the form of the mapping is governed by a number of adjustable parameters.'

Therefore a lot of techniques have been introduced to improve the generalisation capability of neural network models like regularisation techniques [e.g.13,14,15] Bayesian Learning [e.g. 16,17] and also by using the parsimonious networks structure [18]. The most exceptional model for this approach is network pruning techniques and sequential orthogonal training techniques. A sequential orthogonal training techniques gradually builds up a neural network model and avoids unnecessarily large networks structure [19,20].

However, single neural networks sometimes lack robustness when the data is insufficient especially when dealing with real world data due to the fact that the robustness of the network is related to the representativeness of the training data [12]. Single neural networks sometimes suffer badly when applied to unseen data where some neural network might fail to deliver the correct result due to the network training converged to undesired local minima, overfitting or noise in the data [e.g. 21,22]. Therefore the combination of multiple neural networks using simple averaging approach is implemented in this paper with the aim of enhancing the single neural network robustness.

2. Multiple Neural Networks

The idea of multiple neural networks came up from Wolpert [23] where he described about stacked generalisation which is a technique for combining different representations to improve the overall prediction performance. It can also be described as architecture of network consisting of several sub-models and a mechanism which combines the outputs of these sub-models [24]. There are several types of multiple neural networks but the underlying ideas are basically similar and the main difference is on how to create the sub-models as shown in Figure 1 and combined those output to get a single output.

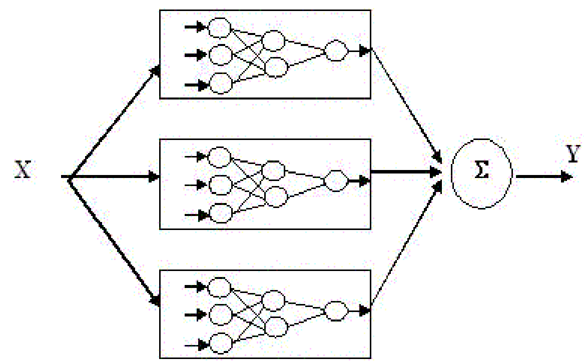

Figure 1. Combining multiple neural networks

Methods of combining multiple networks in current literature can be divided into linear and nonlinear combinations. The common linear combination is averaging and weighted averaging. The linear combination of multiple outputs is to create a single output as a final prediction. In weighted averaging, individual network outputs are multiplied by appropriate weights and then combined to give the final model prediction. Weighted averaging includes PCR and MLR approaches. Zhang [18] used PCR approach to select the combination weights. Another combination scheme is by Wolpert [23] which combines the networks with weights that vary over the feature space. The output from a set of level 0 generaliser are used as the input to level 1 generaliser, which is trained to produce the appropriate output.

Nonlinear combination techniques, include Demspter-Shafers belief based method [25], majority voting [e.g. 26], and also Bayesian model averaging. The Demspter-Shafers belief based method is quite complex and it have to deal with the uncertainty and ignorance of the classifiers. This approach is usually used in model classification or pattern recognition when each network or model represents a character of the image, same as the majority voting combination for example in handwritten recognition [27]. For this paper, simple averaging combination technique is employ to get a final single output. This method is the most common method in combining several model outputs with the weights fixed as shown below:

$$\hat{Y} = w_1\hat{y}_1 + w_2\hat{y}_2 + ... + w_n\hat{y}_n \quad (1)$$

where $\hat{y}_i$ is the network prediction from the *i*th network, n is the number of networks to be combined, $\hat{Y}$ is the final prediction output, and $w_i = 1/n$ is the weight for combining the *i*th network. In this paper the number of network to be combined is 20. In this approach all the networks have the same contribution to the final prediction output even though some of the networks might have better predictions then others.

3. Case study: pH Neutralization Process

The experimental data employed for modeling was obtained from a pH neutralization rig shown in Figure 2. A feed sodium hydroxide (NaOH) solution is fed to the CSTR by a diaphragm pump (metering pump). At the same time, a feed sulphuric acid (H_2SO_4) solution is fed to the CSTR by a diaphragm pump (masterflex pump). A stream leaves the CSTR is called neutralization effluent of the H_2SO_4 and NaOH solution. These NaOH stream and effluent stream passes through a pH sensor to measure its pH values. In this case study, 20 networks with fixed identical structure were developed from boostrap re-samples of the original training and testing data. In re-sampling the training and testing data using bootstrap re-sampling techniques, the training and testing was already in discrete time function, therefore by re-sampling discrete time function it's not effect the sequence of input-output mapping of the prediction.

Then the individual networks were trained by the Levenberg-Marquardt optimisation algorithm with regularisation and "early stopping". All weights and biases were randomly initialised in the range from –0.1 to 0.1. The individual networks are single hidden layer feed forward neural networks. Hidden neurons use the logarithmic sigmoid activation function whereas output layer neurons use the linear activation function. Instead of selecting a single neural network model, a combination of several neural network models is implemented to improve the accuracy and robustness of the prediction models.

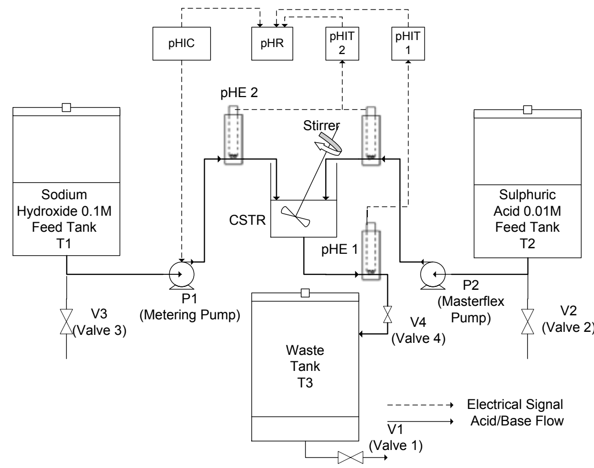

Figure 2. Advanced pH control schematic diagram

There were fours strokes percent for the metering pump during the data generation. The stroke percentages are 40, 60, 90 and 100 respectively. While the stroke length percent and stroke per minute for the masterflex pump are constant at 20 percent during the experiment. The other parameter such as NaOH concentration, H_2SO_4 concentration and H_2SO_4 stream stroke are also remain constant. The duration of each manipulated variables percent changes was 4 min. The process was allowed to reach steady state for perfect mixing during the first three minutes followed by pH evaluation of the effluent for the next one minutes. The pH value for the effluent was then obtained automatically through the pH sensor and the signals transmitted by the pH transmitter to the recorder and it have recorded in every two seconds. Then, the data generated from the experimental rig were divided in to training, testing and validation where in this case study the training data is based on the data taken from the strokes of 40 %, testing data from strokes 100 % and the remaining data is for validation.

This case study apply a one-step-ahead predictions approach where , the process output at time (t-1), y(t-1), is used as a model input to predict the process output at time t, y(t), as follows:

$$\hat{y}(t) = f[y(t-1), u1(t-1), u2(t-1), \quad (2)$$

where u1(t-1) and u2(t-1)is the process input at time (t-1) which is the acid flow and the pump strokes, $\hat{y}(t)$ is the predicted process output (pH) at time t , the lags for this model is 1 for both input and output.

4.0 Results and discussion

Initially, the network was trained using all 1166 data points based on the 40 % stroke of the masterflex pump for single and multiple neural networks. By using the LM optimization method, the training stopped after 100 iterations with the sum square error SSE value of 0.0392 and the correlation coefficient R-square equal to 1.00. The trained network was simulated by feeding it with all of the 40 percent stroke data. Then, the model was tested using 100 percent stroke data which contains 1086 data points.

The testing also stopped after 100 iterations with the sum square error SSE value of 0.6935 and the correlation coefficient R-square equal to 0.9994. Figure 3 presents a plot of the pH value for both network outputs (predicted pH value) and the targets (actual pH value) versus the data points for single neural networks and assumption has been made that by duplicating this individual network using bootstrap re-sampling method, the multiple neural networks model will perform as closed as possible to this model or better after combination. In this case, all predicted points are close to the actual, which means that the network has learned the input–output mappings with a good degree of accuracy.

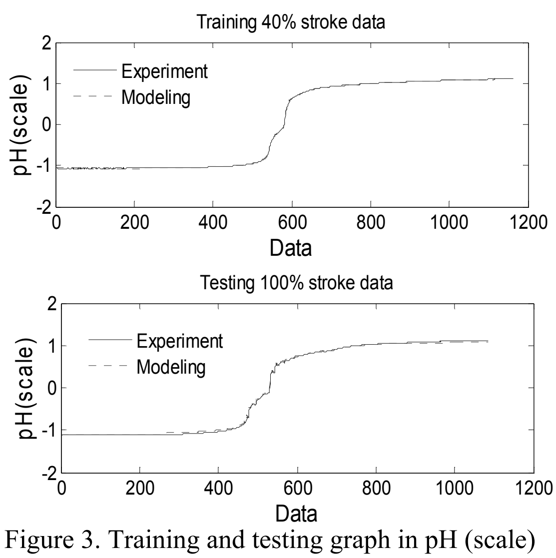

Figure 3. Training and testing graph in pH (scale)

The model has been validated using 60 % and 90 % stroke data which contain around 1000 data points in each set. The validation data will determined whether the generalization capability of the model developed using 40 % and 100 % data for training and testing is acceptable.

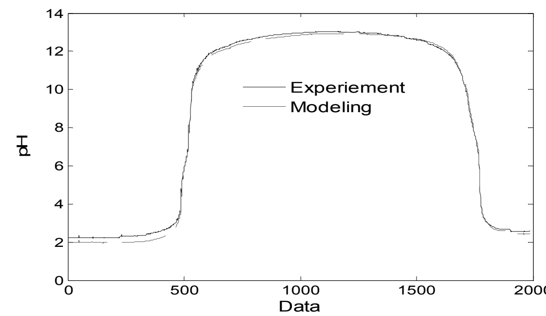

Figure 4. Validation output for 60 % stroke data

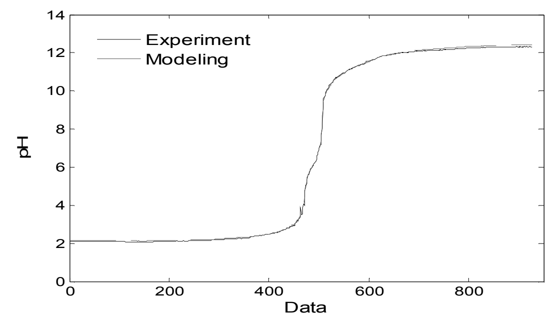

Figure 5. Validation output for 90 % stroke data

Figure 4 and Figure 5 shows the model and actual output in the validation data for single neural networks (SNN). It clearly seen that the single neural networks was performed quite well. The predicted model output showed quite the same as the experiment data, but there is some errors occurred at the low pH region as well as at the end of the high region and also at the transition between the low region and middle region. This might be due to the transition of the pH especially from low region to higher region where the neutralization process was very fast, small changes in the input (acid flow) give a lot of affect to the process.

Then multiple neural networks (MNN) combination approach is applied and the result was shown in Figure 6 and Figure 7 for 60 % and 90 % stroke data respectively. It clearly seen that from Figure 6 and Figure 7, multiple neural networks prediction is significantly better than single neural networked. The predicted and the experiment value can be seen exactly matching for both data.

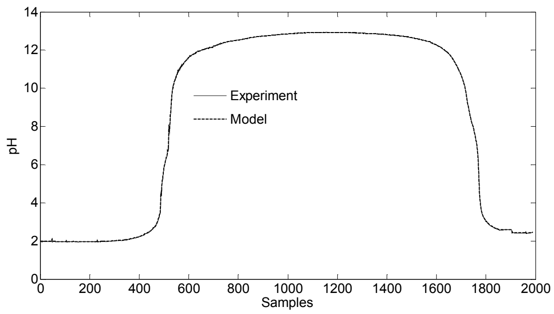

Figure 7. Multiple Neural Networks validation output for 60 % stroke data

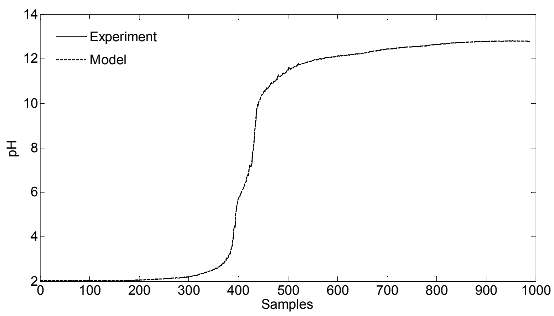

Figure 8. Multiple Neural Networks validation output for 90 % stroke data

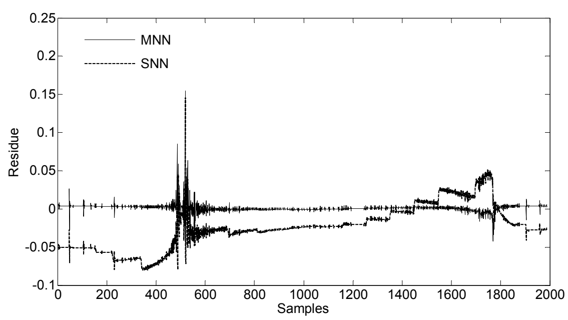

Figure 9. Residue for multiple neural networks (MNN) and single neural network (SNN) prediction for 60 % stroke data

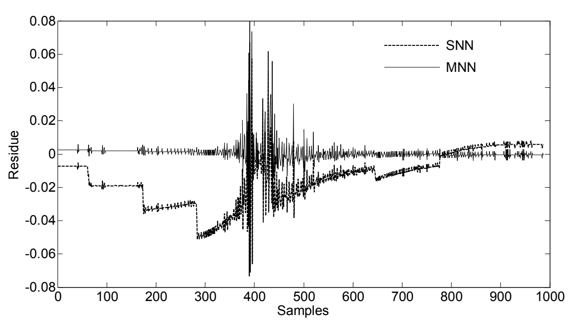

Figure 10. Residue for multiple neural networks (MNN) and single neural network (SNN) prediction for 90 % stroke data

The performance of MNN combination is encouraging especially based on the residue analysis which is shown in Figure 9 and Figure 10. The residue is constant for MNN but for SNN is quite inconsistent especially in the transition of low and upper region. This contributed to the large number of SSE for SNN prediction.

In order to test further the performance of the model, statistical analysis was carried out which is sum square error (SSE), mean square error (MSSE) and relative correlation R-square analysis as well as residue analysis.

The overall statistical analysis result of SSE, MSSE and relative correlation R-square shown in the Table 1 and Table 2. It is clearly shown in Table 1 that the SSE and the MSSE is quite small and in Table 2, the relative correlation (R-square) is nearly to 1 for MNN while in SNN prediction, it's slightly larger for SSE and MSSE. It is shown that the MNN combination model can predict significantly well even though using real process data.

Table 1. Result of the output based on the single and multiple neural networks application on the validation data.

Data	**SSETv**		**MSSETv**	
	SNN	MNN	SNN	MNN
60	2.6757	0.0880	0.0013	4.4234e-005
90	0.4584	0.0458	4.64E-04	4.6383e-005

Table 2. Result of the output based on the single and multiple neural networks application on the validation data for R-square.

Data	**RsquareTv**	
	SNN	MNN
60	0.9977	0.9999
90	0.9996	1.0000

5. Conclusion

A multiple neural network (MNN) was developed to model the performance of a pH neutralization process using experimental data, which was subjected to a series of different stroke percent for sodium hydroxide stream. The inputs to the network were the sodium hydroxide stream flow rate and metering pump percent stroke, and the output was the pH values of the effluent. The Levenberg–Marquardt optimization technique was used together with the 'early stopping' and regularisation methods to improve the robustness of the network.

Application to the real pH neutralization process shows that combining multiple neural networks (MNN) increased the robustness of neural network models compared to single neural network (SNN). The SSE is decreased as well as the increment of R-square analysis compare to single neural networks in all validation data. The result for multiple neural networks combination was consistent especially in residue analysis as well as in R-square and it's concluded that combining multiple neural networks can significantly produced a better models.

Acknowledgement

The authors would like to thank University Sains Malaysia (USM) for financially support this research project using short term grant 6035182.

References

[1]. Chan, H C and C. C. Yu., "Autotuning of gain scheduled pH control: an experimental study", *Ind. Eng. Chem. Res*., vol. 34, 1995.
[2]. Costello, D J., "Evaluation of model based control techniques for a buffered acid-base reaction system", *Trans. IChemI*, vol. 72. Part. A, January 1994.
[3]. Loh, A R, K. O. Looi and K. F. Fong, "Neural network modelling and control strategies for a pH process", *J. Proc. Control*. Vol. 5, 1995, No. 6, pp. 355-362.
[4]. Nascimento, C A O, R. Giudici, R. Guardani, "Neural network based approach for optimization of industrial chemical processes", *Comp. Chem. Eng*. 24 (9–10) (2000) 2303–2314.
[5]. Delegrange-Vincent,N, C. Cabassud, M. Cabassud, L. Durand-Bourlier, J.M. Laine, "Neural networks for long term prediction of fouling and backwash efficiency in ultrafiltration for drinking water prediction", Desalination 131 (1–3) (2000) 353–362.
[6]. Willis, M J, G.A. Montague, C. Di Massimo, M.T. Tham, A.J. Morris, *Artificial neural networks in process estimation and control*, Automatica 28 (6) (1992) 1181–1187.
[7]. Pigram, G.M, T.R. Macdonald, *Use of neural network models to predict industrial bioreactor effluent quality*, Environ. Sci. Technol. 35 (1) (2001) 157–162.
[8]. Mirzai, A.R, J.R. Leigh, "An overview of the applications of neural networks in process engineering", *Comput. Control Eng. J*. 3 (5) (1992) 105–112.
[9]. Karjala, T.W, D.M. Himmelblau, "Dynamic data rectification by recurrent neural networks vs. traditional methods", *AIChE J*. 40 (11) (1994) 1865–1875.
[10]. Hertz J, Krogh A and Palmer R G., *Introduction to the Theory of Neural Computation*, Addison-Wesley, Redwood City, CA, 1991.
[11]. Haykin S. *Neural Networks*, Macmillan College Publicating Company, 1994
[12]. Bishop C., *Neural Networks for Pattern Recognition*, Clarendon Press, Oxford, 1995
[13]. Caruana R, Lawrence S and Lee Giles C, *Overfitting in Neural Networks: Backpropagation, Conjugate Gradient and Early Stopping*. Neural Information Processing System , 2000, 13**:** 402-408.
[14]. Hagiwara K and Kuno K., "Regularisation Learning and Early Stopping in Linear Networks". *International Joint Conference on Neural Networks* (IJCNN 2000)**:** 511-516.
[15]. McLoone S and Irwin G., "Improving Neural Networks Training Solution Using Regularisation. Neurocomnputing", 2001, 37**:** 71-90.
[16]. Crucianu M, Bone R and Asselin de Beauville J P., *Bayesian Learning for Re-current Networks. Neurocomputing* , 2001, 36**:** 235-242.
[17]. Radford M N., *Bayesian Learning for Neural Networks. Statistics*, 1997, 118-183.
[18]. Zhang J, Morris A J, Martin E B and Kiparissides C., "Prediction of Polymer Quality in Batch Polymerisation Reactors Using Robust Neural Networks". *Chemical Engineering Journal*, 1998, 69:135-143.
[19]. Zhang J and Morris A J, *A Sequential Learning Approach for Single Hidden Layer Neural Networks*. Neural Networks, 1998, 11(1): 65-80.
[20]. Zhang J. "Sequential Training of Bootstrap Aggregated Neural Networks for Nonlinear Systems Modelling". In the Proceedings of *American Control Conference* (ACC 2002), 1:531-536.
[21]. Hashem S., *Optimal Linear Combination.* Neural Networks, 1997, 10(4):599-614.
[22]. McLoone S and Irwin G., *Improving Neural Networks Training Solution Using Regularisation. Neurocomnputing* , 2001,37**:** 71-90
[23]. Wolpert D H., *Stacked Generalisation.* Neural networks, 1992,5:241-259.
[24]. Eikens B and Karim M N., *Process* "Identification with Multiple Neural Networks Models". *International Journal of Control* 72(7/8), 1999, 72(7/8):576-590.
[25]. Rogova G., *Combining the Result of Several Neural Networks Classifiers*. Neural Network, 1994, 7(5):777-781.

[26]. Battiti R and Colla M., *Democracy in Neural Net: Voting Schemes for Classification*. Neural Networks, 1997, 7(4):691-707.
[27]. Xu L, Krzyzak A and Suen C Y. “Methods for Combining Multiple Classifiers and their Application to Handwriting Recognition”. *IEEE* Transaction on System, Man, and Cybernetics, 1992, 22(3):418-435.