'94 KACC (1994. 10. 17 ~ 20)

Experimental Study of Neural Linearizing Control Scheme Using A Radial Basis Function Network

Suk-Joon Kim° and Sunwon Park[1]

Dept. of Chem. Eng., Korea Advanced Institute of Science and Technology, 373-1, Kusong-dong, Yusong-ku, Daejon, Korea

Tel)82-042-869-3920, FAX)82-042-869-3910 ,E-mail)swpark@convex.kaist.ac.kr

Abstract - Experiment on a lab-scale pH process is carried out to evaluate the control performance of the neural linearizing control scheme(NLCS) using a radial basis function(RBF) network which was previously proposed by Kim and Park. NLCS was developed to overcome the difficulties of the conventional neural controllers which occur when they are applied to chemical processes. Since NLCS is applicable for the processes which are already controlled by a linear controller and of which the past operating data are enough, we first control the pH process with PI controller. Using the operating data with PI controller, the linear reference model is determined by optimization. Then, a IMC controller replaces the PI controller as a feedback controller. NLCS consists of the IMC controller and a RBF network. After the learning of the neural network is fully achieved, the dynamics of the process combined with the neural network becomes linear and close to that of the linear reference model and the control performance of the linear control improves. During the training, NLCS maintains the stability and the control performance of the closed loop system. Experimental results show that the NLCS performs better than PI controller and IMC for both the servo and the regulator problems.

Introduction

Recently, the explosive development of microprocessors makes the sophisticated nonlinear control theories realizable in chemical process control. Some techniques of nonlinear model based control have been developed under the assumption that the rigorous process models are initially available and holds their validity as time goes on. For simulation and lab-scale experimental studies, this assumption can be valid even though enormous effort and time are spent.

However, this is not reasonable in the situation of industrial plant. Firstly, since the chemical processes are complexes of many sub-unit processes, it is very costly or rather impossible to identify all devices and units. Secondly, the identification experiments of the real plant require much longer time than those of the small processes do because the time constants of the real systems are frequently long. Therefore, even though the nonlinear control techniques are successful in some ideal situations, they encounters inevitable difficulties in the real processes.

Neural networks, inspired from the human nervous system, hold great promise for solving the current difficulties in modeling and control areas. Some researchers proved that they can be used as a universal function approximator[6]. Neural networks also find the proper parameters, called as the weights, by a learning process. Finally, originated from their highly parallel distributed architecture, several benefits are generated: easy implementation in VLSI hardware, robustness against the imperfection of the input data, input data fusion and etc.

Among the characteristics of neural networks, their learning capability attracts the system engineer's attentions. Just by sequentially applying input and output data of the process, we can construct the neural networks to produce the desired outputs. This property of neural networks spurs the control engineer to apply the neural networks in modeling and control of chemical processes. Therefore, The past plentiful engineering studies, stimulated by the above promise, have been performed to apply the neural networks in chemical process control.

However, when developing a control scheme using neural networks for chemical processes, we must consider the special characteristics of chemical plants. Before the chemical plants are constructed, the real information of them is not known accurately. Furthermore, after being constructed, they must be operated in a relatively narrow range for safety and economic reasons. Therefore, a control system must be developed under the following restrictions:

1. It must handle the regulatory problem as well as the servo one.
2. It must hold the control performance within the acceptable range even during training of neural networks.
3. It must obtain the training data from the process without the serious abnormal field tests.

The neural linearizing control scheme(NLCS) using a radial basis function(RBF) network was originally proposed by Kim and Park[12] in the spring meeting of KIChE conference to overcome the difficulties of the conventional neural

[1]To whom correspondence should be addressed.

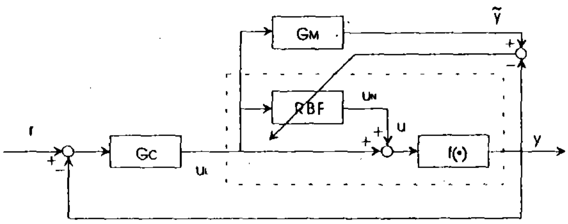


Figure 1. Schematic block diagram of NLCS.

controllers. NLCS showed superior control performance for both the servo and the regulatory problem to that of PI controller in the previous studies simulation.

This paper begins with a brief review of the NLCS and shows the components and characteristics of the experimental apparatus which is a laboratory-scale pH control unit. Finally, we discuss the result of implementation of NLCS to a pH control unit. The experimental results for these systems show the superior control performances of NLCS for simple acid/base systems in a laboratory setting and give its prospect for the real industrial plants.

Summary of Neural Linearizing Control Scheme

NLCS can be applied to the nonlinear processes that are already controlled by the stable linear controllers. In NLCS, a RBF network, a kind of the neural networks, is trained to linearize the relation between the output of the linear controller which can control the process by itself and the process output. Training the RBF network is to minimize the difference between the outputs of the linear reference model(LRM) and the process with the following objective function, E:

$$E = \frac{1}{2}(\tilde{y} - y)^2 \qquad (1)$$

where $\tilde{y}$ and y are the outputs of the LRM and the actual process.

Without the additional test on the process, the LRM can be determined by analyzing the past operation data. The detail procedures for determining the LRM are explained in the results. The training of the neural network is leaded by the pre-defined LRM. Therefore, depending on how to design the LRM, the target of the RBF network can be unique and physically realizable. Since the objective function may vanish to zero and so the weights of the RBF network are not changed any more, the stable learning of the neural network is guaranteed. Additionally, the RBF network is trained by the modified Hierarchically Self-Organizing Learning(HSOL) algorithm which is explained in our previous work[11, 12].

Figure 1 shows the schematic block diagram of NLCS. In that figure, G_M and G_C, RBF and f(•) represent for the LRM, the linear controller, the RBF network and the nonlinear process respectively. Since the neural network is connected to the existing linear controller in parallel, the control performance gradually improves from that of the existing linear controller to that of the proper nonlinear controller. If the RBF network is fully trained, the dynamics of the boxed area, which consists of the RBF network and the process, becomes that of the LRM.

In spite of the high nonlinearity of chemical processes, most industrial plants currently depend on simple linear controllers, for example, PID controller. When the linear controllers are applied to nonlinear systems, the linear controllers are conservatively tuned to insure the stability against the fastest dynamics in the whole operation range. Therefore, if the RBF network linearizes the relation between the linear controller output and the process output, the overall control performance of the linear controller would be improved because the whole dynamics of the process becomes uniform as the fastest dynamics in the operating region. On the other hand, it seems that the process dynamics becomes easy to control from the viewpoint of the linear controller by training of the RBF network.

Table 1. Parameters and steady-state values.

Variable	Description	Value
C_A	concentration of acidic feed	0.005 N
C_B	concentration of basic feed	0.05 N
F_A	flow rate of acidic feed	85 ml/min
F_B	flow rate of basic feed	8.5 ml/min
V	volume of reactor	875 ml
pH	pH	7
Δt	sampling time	5 sec

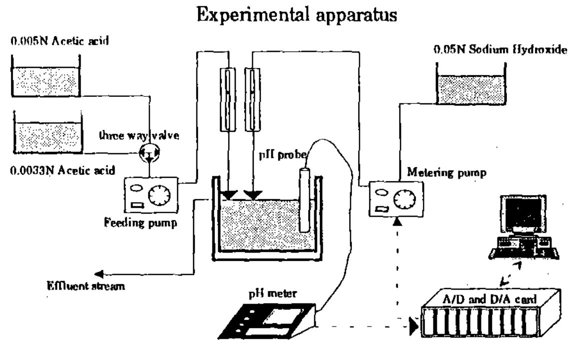


Figure 2. Schematic diagram of the experimental apparatus.

Experimental Apparatus

It is convenient to divide the whole process into four sections for explaining the experimental apparatus: the process feed(acidic) section, the titration feed(basic) section, the reactor section and the control computer section. A schematic diagram of the lab-scale experimental apparatus chosen is shown in Figure 2. Additionally the parameters and the steady-state values used in this experiment are given in Table 1. Some values are not measured exactly or vary to some degree in the normal operation.

Table 2. Parameters used in each experimental step.

Step	Description	Value
	sampling time	Δt=10 sec
PI control	proportional gain	K_C = 5 ml/min
	integral time	T_C = 150 sec
Linear Reference Model	model gain	$K_M = 3\ (\text{ml/min})^{-1}$
	model time constant	T_M = 271 sec
	model time delay	T_D = 10 sec
IMC	filter time	T_F = 10 sec
NLCS	leaning rate	η=3
	initial width of RBF	σ=0.1

PI control → Linear Model → IMC → NLCS

Figure 3. Experimental procedure.

The process feed section consists of two feed tanks, a three-way valve and a feed pump. The feed tanks are made of polyethylene and produced by Nalge company. The capacity of them is 20L. They contain 0.005N and 0.0033N CH_3COOH (acetic acid) respectively. The three-way valve provides the way to select the feed from two feed tanks. In this way it was possible to introduce disturbances in the concentration of the feed stream in a manner as close to a step change as possible. In this experiment, the concentration of the normal feed stream is 0.005N CH_3COOH. The feed pump is a peristaltic Masterflux pump produced by Cole-Parmer International. It is used to maintain a constant flow rate of the acidic feed.

In the titration feed section, there are a feed tank and a metering pump. The feed tank is the same type as the feed tank in the process feed section but contains 0.05N NaOH(sodium hydroxide). The base flow is controlled by a metering pump which is also a peristaltic Masterflux pump and receives a control signal from the computer. The RS232 is used in this signal communication.

The reactor section consists of a polyethylene vessel and a pH measuring system. When we imagine a regular triangle on the cross sectional view of the vessel, the inlet tubes of the acidic and the basic streams and the pH probe located at vertexes of the regular triangle. The inlet tubes go down to the bottom of the vessel and the pH probe is located at the effluent stream. The effluent stream tube is located at 10 cm high from the bottom and so the liquid over this height overflows without pumping. In this way, the volume of the vessel could be maintained constant. A magnetic agitator is also used to ensure proper mixing. Finally, the pH measuring system consists of a pH probe and a pH meter. The pH probe is the general-purpose pH electrode produced by Cole-Parmer International and the pH meter is the model 720 produced by Orion Research Incorporated.

In the control computer section, there are an IBM PC 386SX and a set of A/D and D/A converter made by Jietae Lee, Kyungpook Nat. Unv., Korea. The measuring signal from the pH meter is 1-5V and the actuating signal to the metering pump is 4-20 mA. An on-line data acquisition and control algorithm is coded in MicroSoft C.

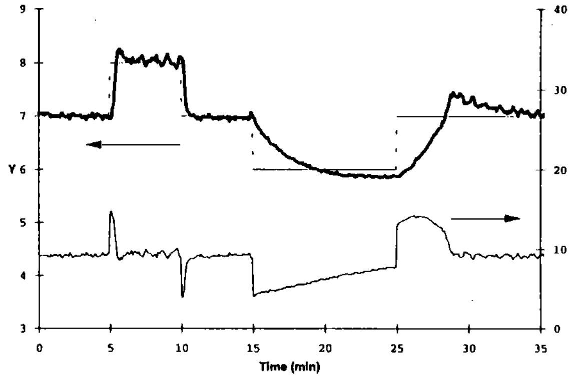


Figure 4. Closed-loop response and the corresponding control action of PI control (setpoint ······· , process output ▬▬ and control action ——).

Results

Figure 3 shows the experimental procedures and the parameters of each step are summarized in Table 2. NLCS can be implemented to the process which would be controlled by a linear controller and of which the past operation data are sufficient. We assume that this lab-scale pH process would be controlled by a simple PI controller, and so we carry out a pre-experiment to construct the control loop using PI controller. The tuning parameters of the PI controller are roughly determined by on-line tuning because it is difficult to clearly find the optimum tuning parameters due to the severe process nonlinearity. Figure 4 shows the process responses and the corresponding control actions for various setpoint changes using PI controller. The control performance for the setpoints between 7 to 8 is acceptable but that between 6 and 7 is very poor.

The above assumption is quite reasonable for industrial situation in chemical industry. Then the following determination of the LRM may be the first step in implementation of NLCS to the industrial processes. Among the historical data as shown in the historical data of PI control, we choose the transient responses of the process with respect to setpoint change from 7 to 8 to determine the LRM, because they represent the fastest dynamics among the past data. After choosing 2 sampling steps as the time delay, $3(\text{ml/min})^{-1}$ as the model gain and the first order autoregressive exogeneous (ARX) model as the base structure of the LRM, we find the model time constant by optimization. The results of this optimization are shown in Figure 5 and the resulting LRM is

$$\tilde{y}(k+1) = e^{-271/5}y(k) + 3\{1 - e^{-271/5}\}u(k-2) \qquad (2)$$

Using this LRM, we construct and implement IMC to the process. The reason for introducing IMC is that IMC can

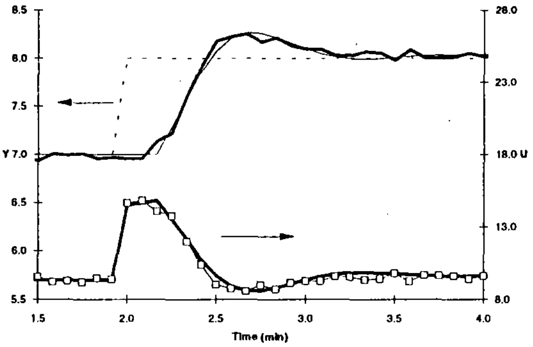


Figure 5. Outputs of the linear reference model after optimization (setpoint······· , process output▬▬▬ , model output —— , control action ▬▬▬ and control action for model —□—).

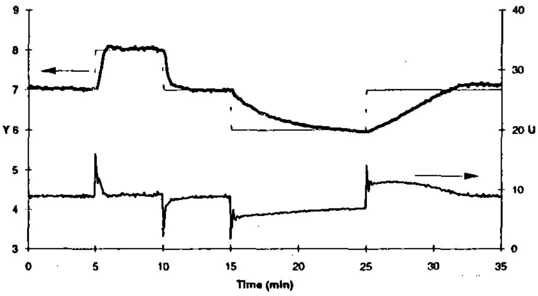


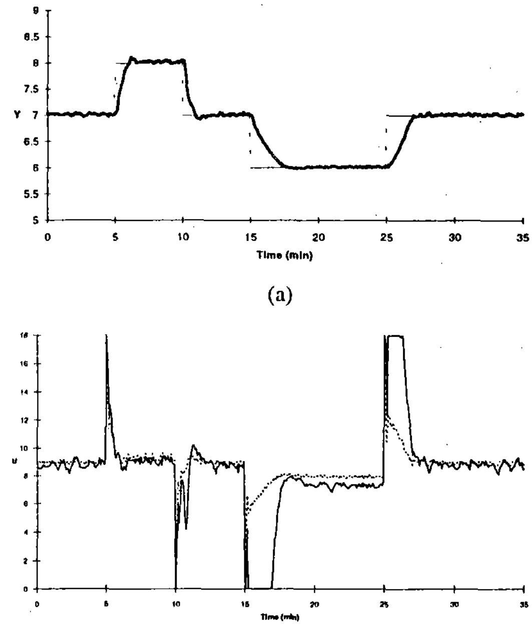
Figure 6. Closed-loop response and the corresponding control action of IMC (setpoint ······· , process output▬▬▬ and control action ——).

effectively handle the time delay and perfectly control the process if the perfect model is given. Figure 6 shows the control performance of the IMC with filter time constant of 10 sec. Since the control performance of IMC severely decreases if the filter time constant is less than 10, we determine that as 10 sec. This phenomena result from the property of the IMC which can not handle the physical bounds of the control action. The control performance of the IMC is improved, comparing with that of the PI controller. However, the intrinsic limitation of the linear controller still appears in that figure.

Based on these preparation, we start to train the RBF network in the NLCS which consists of the IMC as a linear feedback controller and a RBF network. The learning rate is set to

$$\eta = \frac{1}{K_m} \tag{3}$$

The reason for this choice will be explained in the discussion section. After about 10 hours or 20 iterations, the NLCS shows the control performance for servo problem as shown in Figure 7 (a). Since the upper and the lower hard limits of the control action are 0 and 18 respectively, that control performance is optimum for this situation. Additionally, the control



(a)

(b)

Figure 7. Closed-loop response(a) (setpoint ······· and process output▬▬▬) and the corresponding control action (b) (total control action —— and IMC control actions ·······) of NLCS control for the servo problem.

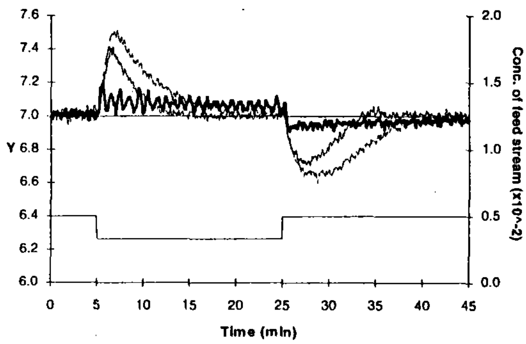


Figure 8. Closed-loop response of NLCS control for the regulatory problem (IMC······· , PI —— and NLCS▬▬▬) and the corresponding concentration of acidic feed stream as the unmeasured disturbance (——).

performance of the NLCS during the training changes from that of the IMC to the final result. Also, there is no stability problem during the training. Figure 7(b) shows the corresponding control action. In that figure, the solid line represents the total control action and the dotted line for the IMC controller. Therefore the difference between the solid and the dotted lines is the control action of the RBF network. This figure shows the superior control performance in spite of the severe nonlinearity of the process.

Finally, we test the control performance of the NLCS for a regulatory problem and compare it with the PI controller and the IMC. Using the three-way valve, we make the step changes in the concentration as an unmeasured disturbance. Figure 8 shows the results of this experiment. Among three controllers NLCS shows the best control performance. For the regulatory problem, NLCS takes longer time to reach a stead state. We will also discuss this phenomenon in the discussion section.

Discussion

First we emphasize the importance of this experiment. There have been many experimental researches on controlling lab-scale pH processes. They have performed the experiments on the simplified process model or ignored some properties of the process. This may cause some problems when implementing their algorithms to large plants, especially with the dynamics of the pH probe and the pH meter. Although the general dynamics of the pH probe is demonstrated in the vendor's manual, there are many factors to influence for modeling it: the degree of mixing reactor, time-varying property of the pH probe, the measuring position of the pH probe and etc. However, NLCS does not use any process model for instruments, devices and units. Therefore, although NLCS is implemented to the lab-scale pH process, implementing the NLCS to an industrial plant is not different from this experiment. Additionally even though we try to maintain the consistency of the operating condition during the training, for example, concentrations of both acidic and the basic feeds, agitation speed, flow rate of the acidic feed, and etc., there exist some variations in these variables. This experiment gives the promise of successful implementation of NLCS even under the more difficult industrial situation.

As mentioned previously, we choose the learning rate, η, as K_M. First, consider the unit of weights and η. From the equation of the output of the RBF network as

$$u_N = \sum_{i=1}^{M} w_i e^{-\|x-c\|} \tag{4}$$

where M, x and c are the number of the RBFs, the input and the center of the *i*th RBF respectively. We know that the weight has the same unit to the control action. Now, from the equation of updating the weights as

$$\Delta w_i = \eta(\tilde{y} - y)e^{-\|x-c\|} \tag{5}$$

we can see that η has the same unit to the inverse of the process gain. Since K_M is set to a largest process gain in the operation range and the linear controller is set to be controllable for K_M, the η set as $1/K_M$ provides the stable training.

Originally, the taining of the neural networks is the optimization problem without constraints. However, there are always physical constraints in the real processes. If these constraints do not take into account of the training, the training goes to the infeasible regions. In this work, we consider the hard limits of the control action among various physical constraints. We make sure that the training does not proceed when the total control action exceeds the physical bounds of the control action. This restriction is expressed as

$$\textit{If}\ u^{old} + \Delta u_N < u_L\ \textit{or}\ u_U < u^{old} + \Delta u_N\ ,$$
$$\textit{then skip the current training.}$$

where u^{old}, u_L and u_U are the control action already applied, the lower and the upper limits of the control action respectively. Additionally Δu_N is the amount of updating the control action and simply approximated as:

$$\Delta u_N \approx \eta(\tilde{y} - y) \tag{6}$$

Using this heuristics, the training of the RBF network does not exceed the hard limits.

For the regulatory problem, NLCS takes longer time to reach a steady state. The RBF network in NLCS does not contain the integral control action to avoid the effect of the unmeasured disturbance as mentioned in reference[11, 12]. Therefore, the change of the steady state due to the unmeasured disturbances is compensated only by the linear controller in the NLCS. Since the effect of the unmeasured disturbance reflected in the process output is relatively small by the tight control of the NLCS, the time to reach the new steady state with the NLCS is longer than that with the linear controller alone.

Conclusions

In this work, the NLCS is implemented on a lab-scale pH process to evaluate its performance and test its possibility. Before introducing the NLCS to the process, we firstly construct a rough PI controller by on-line tuning to evaluate the process data. Using these data, the linear reference model is calculated by optimization. Then, the NLCS is trained to linearize the relation between the outputs of IMC and the process while the IMC with the LRM mainly controls the process. After the learning of the RBF network are fully achieved, the NLCS shows better control performance than PI controller and IMC for both the servo and the regulator control performance. Additionally, during the training, NLCS maintains the stability and the control performance.

The NLCS has been developed to overcome the difficulties of the conventional neural controllers which are encountered when they are applied to industrial processes. The NLCS does not use any process model for instruments, devices and units in this experiment. Therefore, although the NLCS has been implemented to the lab-scale pH process, this experiment gives the promise of successful implementation of the NLCS even under the more difficult industrial conditions.

References

1. Anderson, C. W., "Learning to Control an Inverted Pendulum Using Neural Networks," IEEE Control Systems Magazine, 9, p31-37, 1989.
2. Bhat, N. and T. J. McAvoy, "Use of Neural Nets for Dynamic Modeling and Control of Chemical Process Systems," Computers chem. Engng., 14, 4/5, p573-583, 1990.

3. Chen, F. C., "Back-Propagation Neural Networks for Nonlinear Self-Tuning Adaptive Control," IEEE Control Systems Magazine, 10, p44-48, 1990.
4. Chen, S., S. A. Billings and P. M. Grant, "Non-linear system identification using neural networks," Int. J. Control, 51, 1191-1214, 1990.
5. Guez, A., J. L. Eilbert and M. Kam, "Neural Network Architecture for Control," IEEE Control Systems Magazine, 8, p22-25, 1988.
6. Hornik, K., M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators," Neural Networks, 2, 359-366, 1989.
7. Hunt, K. J. and D. Sbarbaro, "Neural Networks for Nonlinear Internal Model Control," IEE Proceedings-D, 138, 5, p431-438, 1991.
8. Hunt, K. J., D. Sbarbaro, R. Zbikowski and P. J. Gawthrop, "Neural Networks for Control Systems - A Survey," Automatica, 28, 6, p1083-1112, 1992.
9. Jordan, M. I. And D. E. Rumelhart, "Forward models: supervised learning with a distal teacher," Occasional Paper #40, Center for Cognitive Science, Massachusetts Institute of Technology, 1991.
10. Kawato, M., Y. Uno, M. Isobe and R. Suzuki, "Hierarchical Neural Network Model for Voluntary Movement with Application to Robotics," IEEE Control System Magazine, 8, p8-17 1988.
11. Kim, S. J. and S. Park, "A Neural Linearizing Control Scheme for Nonlinear Chemical Processes," submitted to Computers chem. Engng.
12. Kim, S. J. and S. Park, "A Neural Linearizing Control Scheme for Nonlinear Chemical Processes," Spring meeting of KIChE, 1994.
13. Kraft, L. G. And D. P. Campagna, "A comparison between CMAC neural network control and two traditional adaptive control systems," IEEE Control Systems Magzine, 10, 36-43, 1990.
14. Lee, M. and S. Park, "A New Scheme Combining Neural Feedforward Control with Model-Predictive Control," AIChE Journal, 38, ,2, p193-200, 1992.
15. Miller, W. T., R. S. Sutton and P. J. Werbos, "Neural Networks for Control. MIT Press, Cambridge, MA, 1990.
16. Nahas, E. P., M. A. Henson and D. E. Seborg, "Nonlinear Internal Model Control Strategy for Neural Network Models," Computers chem. Engng., 16, 12, p1039-1057, 1992.
17. Narendra, K. S. and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," IEEE Transactions on Neural Networks, 1, 1, p4-27, 1990.
18. Nguyen, D. H. and B. Widrow, "Neural Networks for Self-Learning Control Systems," IEEE Control Systems Magazine, 10, p18-23, 1990.
19. Psaltis, D., A. Sideris and A. A. Yamamura, "A Multilayered Neural Network Controller," IEEE, 8, p17-21, 1988.
20. Rumelhart, D. E. And J. L. McClelland, "Parallel Distributed Processing: Explorations in the Microstuctures of Cognition, Vol. 1: Foundations," MIT Press, Cambridge, MA., 1986.
21. Su, H. T. and T. J. McAvoy, "Integration of Multilayer Perceptron Networks and Linear Dynamic Models: A Hammerstein Modeling Approach," Ind. Eng. Chem. Res., 32, p1927-1936, 1993.
22. Ydstie, B. E., "Forecasting and Control Using Adaptive Connectionist Networks," Computers chem, Engng., 14, 4/5, p583-599, 1990.