*Chem. Eng. Comm.*, 1999, Vol. 172, pp. 151 - 169
Reprints available directly from the publisher
Photocopying permitted by license only



# SIMULATION AND EXPERIMENTAL IMPLEMENTATION OF A NEURAL-NETWORK-BASED INTERNAL-MODEL CONTROL STRATEGY ON A REACTOR SYSTEM

MOHAMED AZLAN HUSSAIN[a],* and L. S. KERSHENBAUM[b]

[a] *Dept. of Chemical Eng., University Malaya, Kuala Lumpur 50603;*
[b] *Center for Process Systems Engineering, Imperial College, London SW7 2BY*

*(Received 29 April 1997; In final form 5 November 1998)*

The use of inverse-model-based control strategy for nonlinear system has been increasing lately. However it is hampered by the difficulty in obtaining the inverse of nonlinear systems analytically. Since neural networks has the ability to model such inverses, it has become a viable alternative. Although many simulations using neural network inverse models for controls have been reported recently, no actual experimental application has been reported on a reactor system. In this paper we describe a novel experimental application of a neural network inverse-model based control method on a partially simulated pilot plant reactor, exhibiting steady state parametric sensitivity and designed to test the use of such nonlinear algorithms. The implementation involved the control of the reactor temperature under set point changes, disturbance rejection and set point regulation with plant/model mismatches. Simulation tests on the model of the system were also carried out to enable better design of the neural network models and to highlight the differences between simulation and actual online results. The online implementation results obtained were sufficient to demonstrate the capability of applying these neural-network-based control methods in real systems.

*Keywords:* Inverse model; neural networks; nonlinear control; pilot plant; reactor

## INTRODUCTION

Although the use of linear control methods is prevalent in the chemical process industries, they have their limitations especially when dealing with

*Corresponding author. e-mail: azlan@fk.um.edu.my

nonlinear plants in wide operating regions as commonly found in these industries. However with the progress in nonlinear control theory and the advancement in computer technology, nonlinear control strategies are coming to the forefront (Bequette, 1991). One such technique is the nonlinear based inverse-model control strategy. The ease and speed of applying this method relative to other possible methods (such as the predictive schemes) for many applications is clearly evident. However this method relies heavily on the availability of the inverse of the system's model, which acts as the controller in this scheme. Unfortunately the inverse of a system may be difficult to obtain analytically for many nonlinear systems, which is one reason why its use is not widespread in the control of such systems. Since neural networks have the potential to model any system, modeling these inverses with them in these inverse-model-based strategies has become attractive. These connectionist models also have the ability to learn the frequently complex dynamic behavior of a physical system. In fact many researches *e.g.* (Cybenko, 1989; Hornik, 1989) have recently proven that any continuous functions can be approximated to an arbitrary degree of exactness on a compact set by a feedforward neural network comprising two hidden layers and a fixed, continuous non-linearity.

In recent years, much simulation work involving neural networks in the inverse-model control scheme have been done (Hunt, 1992a; Willis, 1992; Hussain, 1996). These simulations were useful in obtaining a preliminary analysis of the stability of these algorithms as well as enabling us to test quickly the implementation and effectiveness of these methods. However many of these simulations were done with the system model under ideal conditions, without experimental verification of the efficacy of the proposed control strategies. Since the real plants do not behave in exactly the same manner as their models, the real performance and stability tests can be efficiently done by subjecting these algorithms and methods to an actual plant. Plant/model mismatches and disturbances are also inherently present in these real systems. In fact, these control algorithms would only be useful for industrial applications if proven successful in pilot plants, which is the common, safe and economical approach for testing new and advanced methods such as these neural-network based methods. This paper reports a novel study involving the implementation of this neural-network-inverse-model based control strategy on a partially simulated pilot plant reactor system, designed to test such nonlinear methods. To date, no other application of these neural-network inverse-model-based control method on an actual pilot plant has been reported. The reactor condition chosen for this work also represents an interesting case study since it exhibits steady state parametric sensitivity *i.e.*, introduction of disturbance either by changing the inlet conditions or by introducing plant/model mismatches drastically change the system's temperature and concentration. Another important result shown in this work is the comparison between the simulation results and the experimental results under the same imposed conditions for set point changes, disturbances and mismatches, which is rarely highlighted in other works.

## NEURAL-NETWORK INVERSE MODEL CONTROL STRATEGY

### Internal Model Control (IMC) Scheme and the Neural Network Models

The method utilized in our implementation here is the nonlinear internal model control technique, which is basically an extension of the linear IMC method (Economou *et al.*, 1984). In this scheme, both the neural network forward and the inverse models acting as the controller, are used directly as elements within the feedback loop (as seen in Fig. 1). The procedure of training a neural net to represent the forward dynamics of a system (*i.e.*, obtain outputs given the inputs) is referred to as forward modeling and the models obtained from this procedure are called the forward models while

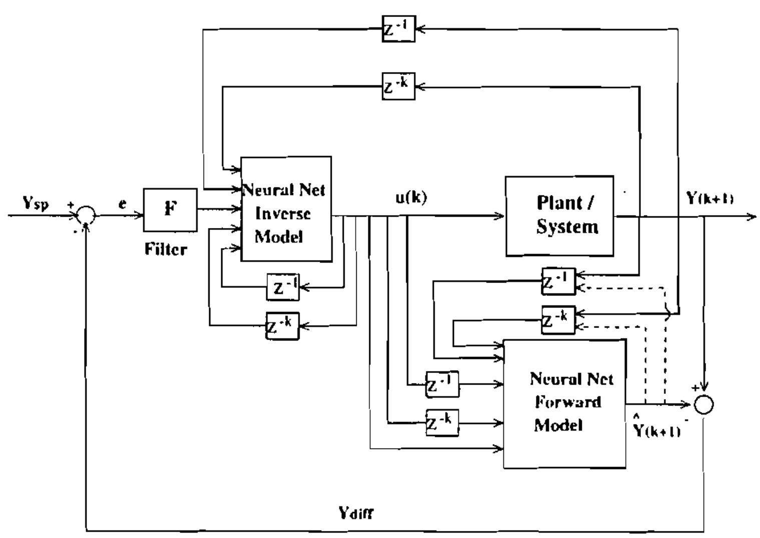


FIGURE 1 Neural network in IMC strategy.

the inverse models are the neural net models representing the inverse of the systems dynamic. Details of the control method and the training of the models can be found in various references (Hunt, 1992b; Antsaklis, 1994). However one detail of this work worth mentioning here is concerning the choice of the input size to the network. There are many alternatives to the choice of the number of past inputs, $m$ and number of past outputs, $n$ *i.e.*, to the choice of the input node size assignment of the network model (which equals $n+m$). Various model validation techniques have been proposed as guidelines to check on the adequate assignment of these inputs (Billings *et al.*, 1992) but the final choice still remains a case-to-case decision. In this work, we assume that the input and output structure is equivalent to the plants' (discrete-time linearised) structure, and hence our initial choice of the input size is dependent on the order of the plant. If they are inadequate, they are subsequently revised until the training achieves satisfactory convergence. From our experience in training various neural network models, only slight modifications to the size are normally necessary from these initial estimates. This method of deciding the input and output pattern has various advantages. One is that it preserves the parsimonious character of the network. Second is that it does not result in a total black-box identification but one that is partly related to the plant model order. Another advantage is that it avoids the use of a large sequence size of past inputs, $m$ and past outputs, $n$ (normally done in practice) resulting in big networks, longer training period and inconsistency in the quality of the output predictions with the risk of over-parameterisation.

## CASE STUDY – PARTIALLY SIMULATED REACTOR SYSTEM

The reactor system used in this study is shown in Figure 2. This pilot plant system, called PARSEX (Partially simulated exothermic) reactor, has been devised for testing the performance of various estimation and control algorithms (Kershenbaum *et al.*, 1994). It basically consists of two main units: a continuous well-stirred reactor of approximate volume of 0.1 $m^3$ and a separate cooler section with approximately 0.7 $m^2$ of heat transfer area. The reactor is charged with water which represents the liquid reactants in this system. Heat within these reactants is exchanged with the cooling medium by pumping the reactants through the external cooler before being recycled into the reactor again. The cooler is provided with good circulation of cooling water and fresh make up water from the main water supply system. The feed to the reactor is pumped into the reactor from the feed

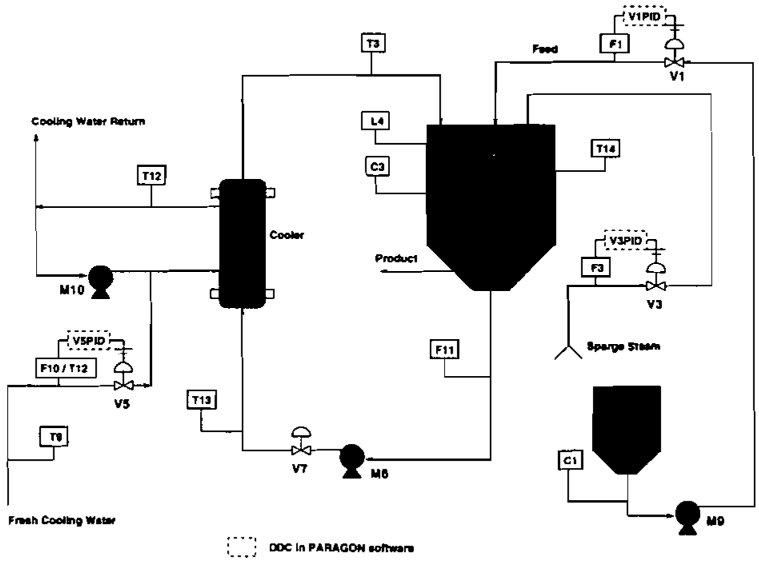


FIGURE 2 Diagram of the PARSEX reactor system.

tank and this feed flow controlled by the control valve V1. The temperature of the reactor, measured by detector T14 is regulated by the cooling water temperature, T12. This cooling water temperature is manipulated by controlling the fresh make-up cooling water flow through control valve V5. The "reaction" in the reactor is simulated by solving the relevant dynamic mass and heat balance equations modeled for the system and the equivalent amount of heat released in the reactor is produced by injecting the required amount of steam into the system. As it is notoriously difficult to test the performance of control algorithms experimentally on reactive systems for reasons of safety as well as economics, many exothermic reactions can be experimentally simulated quite realistically in this way and the effect of this operation is to achieve close resemblance in the pilot plant to the real reactor with normal reactions.

**Reactor Model Equations and Operating Conditions**

In this case, the reactor basically operates as an exothermic stirred tank reactor with first order reactions. The first principles model developed for this system can be found in Limqueco, 1990. These models are utilized in different ways: for assessing the steady state conditions of the reactor under

TABLE I Physical properties and process data for reactor

| *Item* | *Symbol* | *Value* | *Units* |
|---|---|---|---|
| Volume of Reactor | Vr | 0.1 | $m^3$ |
| Heat Transfer Area | Ar | 0.7 | $m^2$ |
| Heat Transfer Coefficient | Ur | 68 | Kcal/(min·$m^2$·°C) |
| Volume of jacket | Vj | 0.012 | $m^3$ |
| Density of reactants | $\rho_r$ | 1000.0 | Kg/$m^3$ |
| Density of jacket fluid | $\rho_j$ | 1000.0 | Kg/$m^3$ |
| Molar heat capacity of reactants | Cpr | 1.0 | Kcal/(Kg·°C) |
| Molar heat capacity of jacket fluid | Cpj | 1.0 | Kcal/(Kg·°C) |

TABLE II Parameter values (dimensionless) for reactor

| | | |
|---|---|---|
| Dimensionless activation energy | $\gamma$ | 20 |
| Damkohler number | $\phi$ | 0.11 |
| Dimensionless heat of reaction | $\beta$ | 7 |
| Dimensionless heat transfer coefficient | $\delta$ | 0.5 |
| Dimensionless Volumetric flowrate | $q$ | 1 |

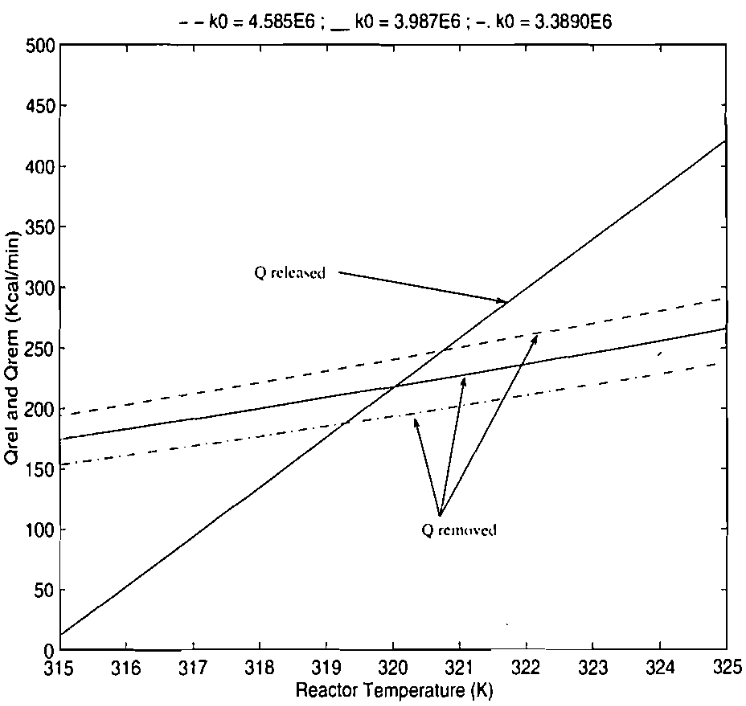


FIGURE 3 Reactor steady state analysis-change in $k_0$.

different conditions, for simulation of the control strategy and also for simulating the concentration of the reactor during online implementation.

The reactor operating values in Table I in conjunction with the parameter values in Table II, are the nominal values utilized in the simulation and online implementation of the neural-network-inverse-model based control strategy. The system with such parameter values exhibit steady state parametric sensitivity, which makes it an interesting case study for studying these neural-network model-based inverse control strategy (Limqueco, 1990). An example of this sensitivity can be clearly seen in Figure 3, where an increase or decrease in the rate constant $k_0$ by 15% changes the nominal steady state reactor temperature to another value immediately. The steady state values are indicated by the intersection between the heat released in reaction, QRrel and the heat removed by the cooling system, QRrem. The purpose of applying these control strategies is to suppress the disturbances caused by the system's sensitiveness to such parametric changes as well as for set point tracking.

## CONTROL IMPLEMENTATION WITH IMC STRATEGY - SIMULATION STUDIES

This section describes simulations, incorporating the actual feasible variables and parameters as in Tables I and II, for the control of the system under servo and regulatory action. Control was implemented in the IMC strategy incorporating the neural network forward and inverse models as mentioned earlier. The system was modeled by a multilayered feedforward neural network for the forward model to predict the one-step ahead temperature and by an inverse model to predict the required control action *i.e.*, the jacket cooling water temperature (assumed equal to the jacket set point temperature in simulation). Details of the training procedure can be found in Kershenbaum, 1995. The final forward model chosen has a configuration of 6 input nodes, 20 hidden nodes and 1 output node and the final inverse model has a configuration of 6 input nodes, 25 hidden nodes and 1 output node. Both of them have a one hidden layer structure with nonlinear output nodes. These forward and inverse models were then incorporated in the IMC configuration for set-point tracking and disturbance-rejection control studies, as discussed below.

### Set Point Tracking

The first simulation was performed for set point tracking from the steady state temperature of 47°C to a lower (42°C) and a higher (53°C) value. These temperatures are chosen in this simulation since they will be the feasible values expected to be implemented in the actual reactor system later. However incorporation of properly selected filtering action prior to the controller enabled us to obtain smoother responses and control actions, which are all desirable in the real plant. Two values of the tuning filter constants, ($\alpha$) *i.e.*, 0.04 and 0.85 were chosen to demonstrate the effect of these parameters in the simulations. Different transient responses and control actions were obtained at the step changes, reflecting the different tuning constants used. The result with $\alpha = 0.04$ showed fast tracking response but with sudden short spikes in the control action at each of the step changes. This is not feasible for the actual plant implementation as they may exceed the plant limits of operation at times. The better result was obtained with $\alpha = 0.85$ where fast and smooth tracking responses and control actions were obtained, as clearly seen in Figure 4. Consequently all subsequent control implementation for this reactor system, whether in simulation or online implementation, incorporates this value as the filter tuning constant for the controller. Note that TR indicates the reactor temperature and TJ indicates the jacket temperature in all graphs.

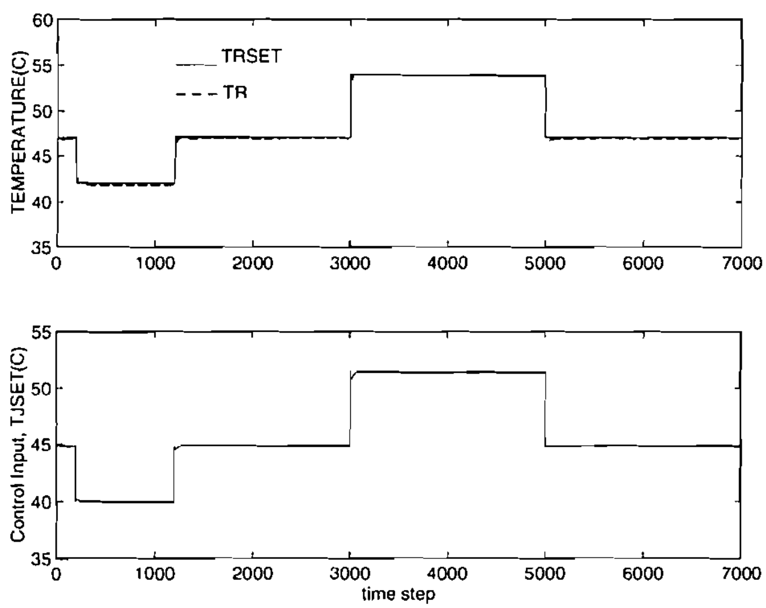


FIGURE 4 Set point tracking with $\alpha$ = 0.85 (simulation).

**Disturbance Rejection - Decrease in Feed Flow Rate**

To test the system under disturbance rejection, the reactor temperature was initially controlled under the neural network IMC control strategy at the nominal steady state temperature of 47°C until the 300th time step. At this instant the feed flow rate was decreased by 50% to 0.00374 $m^3$/min while the controller output was kept to its previous nominal value on open-loop control. At the 1000th time step the neural network controller was initiated again and the reactor temperature, as can be seen in Figure 5 was brought down back close to the nominal steady state value in less than 30 time steps, with slight spikes in the control action.

**Set Point Regulation under Plant/Model Mismatch Cases - Increase in $k_0$ and $\Delta H$ Respectively**

This simulation involves set point regulation of the temperature under various forms of plant/model mismatches such as the increase in the rate constant $k_0$ by 15% to 4.585E6 $\text{min}^{-1}$ and the heat of reaction $\Delta H$ by 10% to 115560 kcal/min respectively. These model mismatches are common in

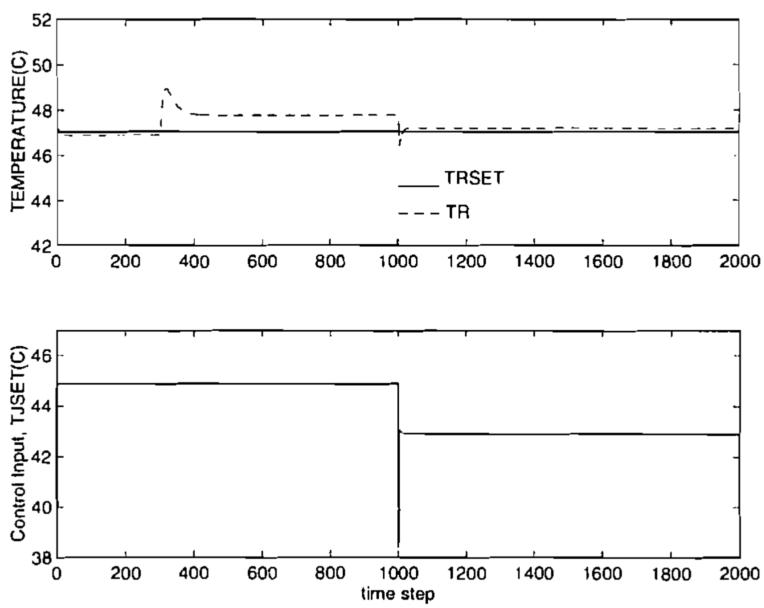


FIGURE 5 Disturbance rejection – change in feed flow rate (simulation).

real life situations due to model uncertainties, side reactions, catalytic deactivation *etc.* They were introduced at the 300th time step with the system under open loop control. When the neural network controller was initiated at the 1000th time step it rejected these mismatches and brought the system back close to its set point (nominal steady state condition) within less than 30 time steps for both cases, with responses similar to that of Figure 5.

## CONTROL IMPLEMENTATION - EXPERIMENTAL STUDIES

The objective of this novel experimental study is to implement the neural network based internal model control strategy on the real plant as well as to investigate the correspondences between the previous simulation tests and that of the experiment. Here the reactor temperature is controlled under set point tracking and disturbances rejection studies involving external disturbances and plant/model mismatches, as per the simulation studies. However before actual implementation, we describe the experimental computer control system and the important steps towards online data gathering in obtaining the relevant neural network models.

### Computer Control System in the Experimental Study

The PARAGON 550 software, running on a NAGA 486 (33 MHz) personal computer and interfaced with the control simulation program (in FORTRAN), manages the overall data acquisition and control of the various control loops. At high frequency (typically every 2 secs) the reactor simulation program reads in, through the PARAGON system, the pertinent measurements: reactor temperature and feed rate. These are used to calculate the concentration of the reactants in the reactor by solving the CSTR model equation in the reactor simulation program. Knowledge of the heat of reaction and rate constants of the simulated reactions allows one to calculate the rate of heat evolution, which is then converted to the equivalent rate of steam needed to be sparged into the system. The desired flow rate of steam is then passed back as set point to the controller V3PID in the PARAGON system, at every 2 secs interval. This steam flow rate is controlled by the steam control valve, V3 under the action of the PI loop in the controller V3PID (refer to Fig. 2).

At a lower frequency (every 6 secs), the advanced control algorithms employed *i.e.*, the neural-network-based IMC method determines the desired control input *i.e.*, set point of the cooling water jacket temperature which is then relayed back to the PARAGON system. Here the neural network controller acts as the master controller and the controller V5PID (in PARAGON) as the slave controller. This slave controller manipulates, under a conventional PID loop, the fresh cooling water flow rate, F10 through the control valve V5. This flow rate directly affects the jacket cooling water temperature, T12 and subsequently the reactor temperature, T14. The set up of this cascade control loop with the neural network control action as the master controller can be seen in Figure 6.

**Data Gathering from Reactor - Offline Training**

In order to implement the neural-net inverse control methods on-line, open loop experiments were performed on the plant itself to generate the relevant data for training the neural-net forward and inverse models of the real system. The open loop data were generated by varying the control input signal *i.e.*, the cooling water jacket set point temperature, in various step sequences, sufficient to excite the system adequately and extract its dynamic behavior. The input excitation signals were not chosen randomly as normally done in simulation but were basically determined by various factors such as; the intended region of operation, the speed of response of the system at each step change and the period available for performing a single open-loop experiment continuously in real time. Under the operating condition of Tables I and II, the initial operation of the PARSEX reactor was targeted at a cooling water temperature of 45°C with a steady state reactor temperature of 47°C and dimensionless concentration of around

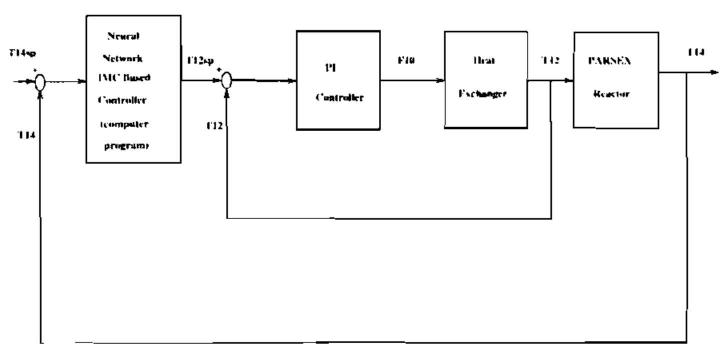


FIGURE 6 Control loop (cascade) for neural network implementation.

0.72. The control input *i.e.*, cooling water set point temperature, was varied to higher and lower values in multistep sequences, with varying frequencies, from this initial steady state value. An example of this can be seen in Figure 7. The data were sampled every 6 secs so that a good representation of the system dynamics could be acquired for training the forward and inverse model; the input/output pattern of which are shown in Figures 8 and 9 respectively. The data collected included the present and past values of the control input (*i.e.*, jacket set point temperature), reactor temperature and concentration respectively, in accordance to the input and output pattern of these forward and inverse models.

Both the forward and inverse models were trained using the normal backpropagation method with momentum term. An adaptive learning rate was used in the initial stages of learning to speed up the convergence rate due to the large amount of data being processed. The final neural net forward model chosen, has 6 input nodes, 25 hidden nodes and 1 output node with sigmoidal activation function and a nonlinear output. A similar configuration was obtained for the inverse model.

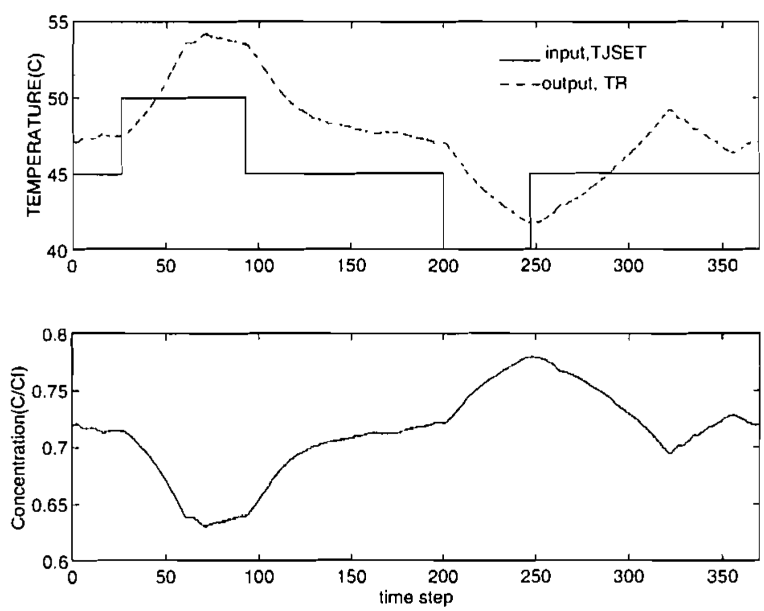


FIGURE 7 Open loop data – training data set (experimental).

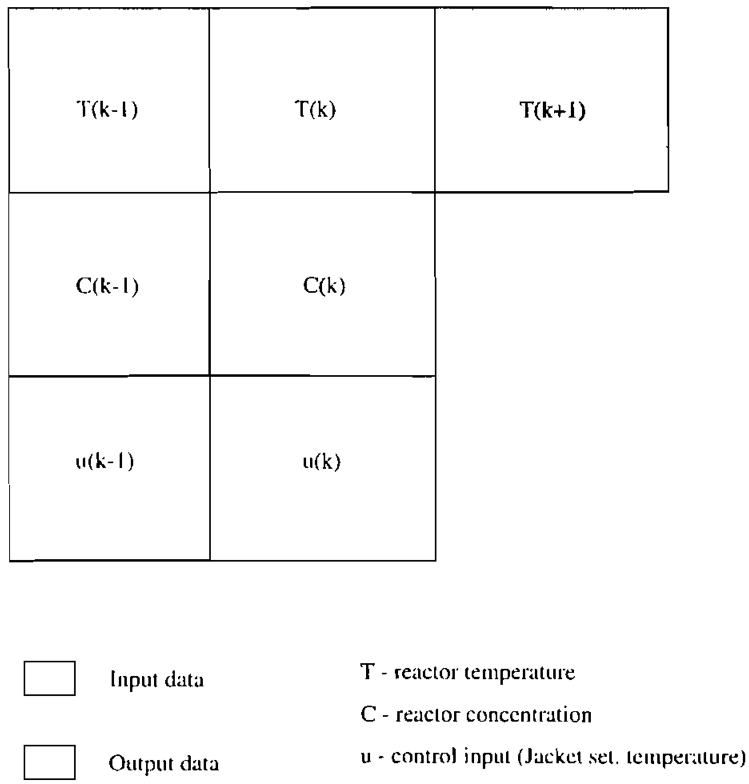


FIGURE 8 Input-output pattern : Forward model.

## Online Control Implementation

Set-point tracking and disturbance-rejection studies, similar to those in simulation, were then implemented on the PARSEX reactor online, as discussed below:-

## Set Point Tracking

The set point tracking experiment was done with step down, to 42°C, and step up, to 53°C from the initial steady state temperature of 47°C. Each time step represents the concurrent data acquisition and control implementation sampling time of 6 secs. The experimental results obtained in this case can be seen in Figure 10. The overall results showed that the reactor temperature could track the set point profile reasonably well with offsets in the range of

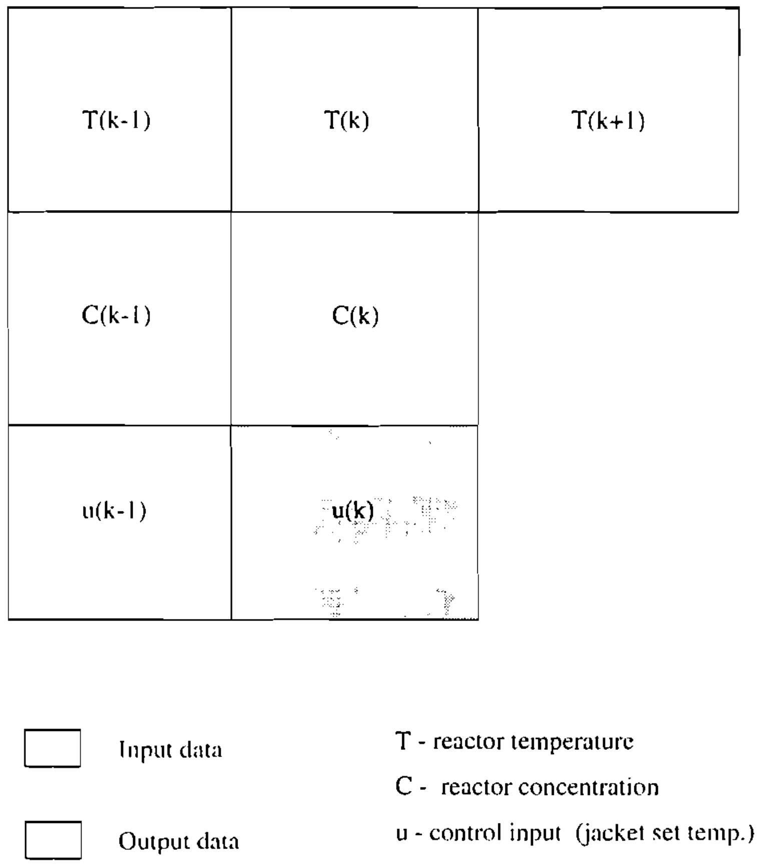


FIGURE 9 Input-output pattern : Inverse model.

0.5 to 1.1°C. Secondly the set point tracking action was fast when stepping down but sluggish when stepping up. This is basically due to saturation of the control valve, V5 (controlling the temperature of the cooling water) at these lower set-point values, as seen in the valve % opening in Figure 10.

**Disturbance Rejection**

The disturbance rejection study was performed as follows: At the 300th time step interval, the disturbance was introduced into the system by reducing the feed flow rate by about 50% to 0.00374 $m^3/s$. At this instant the controller action *i.e.*, jacket set point temperature, was frozen to its latest value (of over

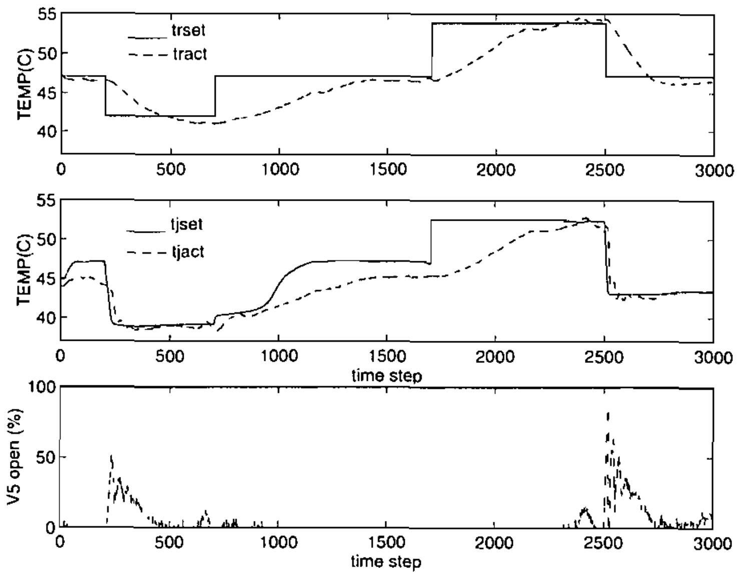


FIGURE 10 Set point tracking (experimental).

47°C), which is the value prior to the introduction of the disturbance. The system was then put under open-loop control until the 1000th time step. The rise in temperature in the real plant (at 55°C) was much higher than that achieved in simulation due to the fact that the jacket temperature, which reached the jacket set point temperature of over 47°C under open loop operation was higher than the ideal temperature of 45°C achieved in simulation (where the jacket set point and actual jacket temperature were assumed equal at all times). When the controller was initiated again at the 1000th time step, the neural network controller immediately acted to reduce the jacket set point temperature and hence the reactor temperature close to its initial nominal value within about 600 time steps. This result can be seen in Figure 11.

### Set Point Regulation under Plant/Model Mismatch Cases – Increase in $k_0$ and $\Delta H$

These case studies involved set point regulation with plant/model mismatches introduced by increasing the rate constant $k_0$ by 15% to 4.585E6 $\text{min}^{-1}$ and the heat of reaction $\Delta H$ by 10% to 115560 kcal/min respectively. These changes at the 300th time step were introduced by

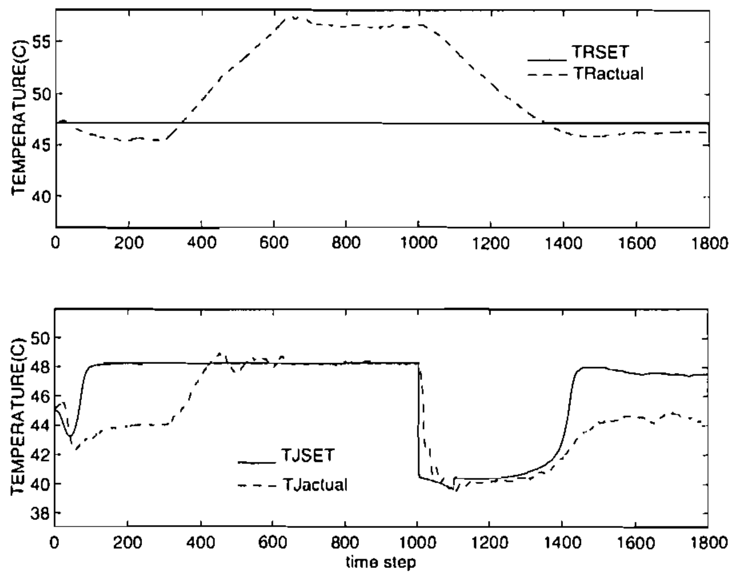


FIGURE 11 Disturbance rejection – change in feed flow rate (experimental).

changing the relevant parameters within the reactor program, which subsequently resimulated the amount of heat generated and hence revised the equivalent amount of steam needed to be injected into the system. At these instant of changes, the controller action *i.e.*, jacket set point temperature, was frozen under open loop control to its value prior to the introduction of these disturbances (*i.e.*, 47.5°C). When the controller was initiated again at the 900th time step, the neural network controller immediately acted to reduce the jacket set point temperature and hence the reactor temperature close to their initial nominal value within the 1330th and 1300th time step for both plant/model mismatch cases respectively, the responses being closely similar to that shown in Figure 11.

## DISCUSSIONS AND CONCLUSIONS

In this work we have successfully demonstrated a novel implementation of the inverse-model neural network controller on a partially simulated pilot plant reactor, which exhibits steady state parametric sensitivity. These experimental investigations also enabled us to test and validate the neural network control strategies on real plant conditions as well as to complement the simulation results. They highlighted some of the features not normally seen in analytical and simulation studies. Simulation studies alone are inadequate to guarantee the successful implementation of such advanced control strategies and a relatively simple low-cost experiments, such as this, is important and useful in testing such online tests.

As observed for the pure simulation case, good overall set point tracking was achieved with a maximum offset of 0.2°C only. This was expected as the neural network models were well trained offline on the data of the perfect model and also due to the fact that the simulation was performed under various assumed ideal conditions. The very slight offsets were however present as it is difficult for the neural network controller to attain the exact inverse of the plant model and some error of tolerance exists in its output prediction. These offsets were also in line with the theoretical analysis of such inverse-model control strategies mentioned in other references (Nahas, 1992; Kershenbaum, 1995).

Closely similar results were achieved in the actual online implementation but with more offsets (maximum of 1.1°C) at the higher and lower set points. The amount of offsets attained reflected the accuracy of the neural-network inverse models trained on the actual plant data. Sluggish behavior was however observed in this experiment when stepping up from the lower set point to the nominal steady state. These sluggishness is due primarily to the valve saturation in this region of operation (jacket inlet water flow valve, V5 was fully closed at this point) and also to the small temperature gradient that exists between reactor and jacket temperature in this region. Hence the temperature of the jacket could not increase to the required value as quickly as expected.

Table III summarizes the results for the disturbance rejection cases due to the change in feed flow and plant/model mismatch cases in both the simulation and online implementation. For the pure simulation case, the rejection of disturbance (response achieving close to about 1% of the set point value), in all the 3 cases, were achieved within a few time steps (of less than 30) with slight spikes in the control action. The period of disturbance rejection for the online case was however much higher than in the simulation case. This was again to be expected with the higher prediction accuracy of

TABLE III Disturbance rejection – No. of time steps to return to set point

| | *Change in Feed flow* | *Change in $k_0$* | *Change in $\Delta H$* |
|---|---|---|---|
| Simulation | < 30 | < 30 | < 30 |
| Experimental | 600 | 430 | 400 |

the forward and inverse model in the simulation since they were easier and adequately well trained on the perfect model than on the actual plant data (which was slightly noisy in nature) and also to the fact that the magnitude of the disturbance to be rejected was much smaller in simulation than in the online implementation (where its jacket temperature was at a higher temperature during onset of disturbance rejection).

To sum it all up, these experimental results showed the vast potential and capability of implementing the neural-network-based inverse-model-control strategy, in this case the IMC method, in real life situations. This neural network strategy also showed fast implementation (about 2 secs of real time on the Sun Sparc2 machine), which is very much desirable in online applications for controlling systems with fast responses. Furthermore the proper implementation of these neural-network strategies, which lies its main strength, is in the accuracy of the forward and inverse models incorporated in the IMC strategy. The attainment of these models in turn depends a great deal on the quality of the open-loop data used for training them. Simulation studies prior to the actual implementation was also found to be useful in pre-designing these neural network models for the experimental test and for better understanding between the differences of simulation and actual implemention.

## *References*


Bequette, B. W. (1991) Nonlinear control of chemical processes: A review, *Ind. Eng. Chem. Res.*, **30**, 1391–1413.

Cybenko, G. (1989) Approximation by superposition of a sigmoidal function, *Math. Control Signals System*, **2**, 303–314.

Hornik, K., Stinchcombe, M. and White, H. (1989) Multilayer feedforward networks are universal approximators, *Neural Networks*, **2**, 359.

Hunt, K. J. and Sharbaro, D. (1992a) Studies in neural network based control, *IEE Control Engineering Series*, London.

Willis, M., Montague, G. A. and Morris, A. J. *et al.* (1992) Artificial neural networks in process estimation and control, *Automatica*, **28**(6), 1181–1187.

Mohamed Azlan, Hussain (1996) Inverse model control strategies using neural networks: Analysis, simulation and online implementation, *Ph. D. Thesis* (University of London).

Economou, C., Morari, M. and Palsson, B. O. (1986) Internal model control. Extension to nonlinear systems, *Ind. Eng. Chem. Process Des. Dev.*, **25**, 403–411.

Hunt, K. J., Sbarbaro, D., Zbikowski, R. and Gawthrop, P. J. (1992b) Neural networks for control systems – A survey, *Automatica*, **28**(6), 1083–1111.

Antsaklis, P. J. (1993) Control theory approach, Mathematical Approaches to neural networks, Elsevier Science, U.K., pp. 1–23.

Billings, S. and Chen, S. (1992) Neural networks and system identification, *IEE control engineering series no.* 46, *IEE*, U.K.

Kershenbaum, L. S. and Kittisupakorn, P. (1994) The use of a partially simulated exothermic reactor for experimental testing of control algorithms, *Trans. IChemE*, **72**, 55–63.

Limqueco, L. C. and Kantor, J. C. (1990) Nonlinear output feedback control of an exothermic reactor, *Computers Chem. Eng.*, **14**(5), 427–437.

Kershenbaum, L. S., Hussain, M. A. and Allwright, J. C. (1995) Inverse model control strategies based on neural networks – analysis, simulation and online implementation, *AIChE 1995 Annual Meeting*, Florida, 180a.

Nahas, E. P., Henson, M. A. and Seborg, D. E. (1992) Nonlinear internal model control strategy for neural network models, *Computers Chem. Eng.*, **16**(2), 1039–1057.