INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
Application of back-propagation neural networks to system identification and process control

Broussard, Mark Randall, Ph.D.
Rice University, 1991
RICE UNIVERSITY

APPLICATION OF BACK-PROPAGATION NEURAL NETWORKS TO SYSTEM IDENTIFICATION AND PROCESS CONTROL

by

Mark Randall Broussard

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE DOCTOR OF PHILOSOPHY

APPROVED, THESIS COMMITTEE

Ka-Yiu San,
Associate Professor of Chemical Engineering, Chair

Sam H. Davis,
Professor of Chemical Engineering

John W. Clark Jr.
Professor of Electrical Engineering

Kyriacos Zygourakis,
Associate Professor of Chemical Engineering

Houston, Texas
April, 1991
Application of Back-propagation Neural Networks to System Identification and Process Control

Mark Randall Broussard

ABSTRACT

Certain properties of the back-propagation neural network have been found to be potentially useful in structuring models for process control applications. The network’s relative simplicity and its ability to learn by example are potentially important in the effort to develop automated continuous on-line system identification. The capacity of the network to form nonlinear mappings enhances research designed to advance nonlinear system identification techniques. Since most real processes are nonlinear, this prospect can have wide impact.

The unstructured nature of the network model was found to be controllable by techniques developed in the study. Care must be taken to identify and train the network with consistent data that contains sufficient dynamical information.
Model-based fine tuning of a controller using a network model that was identified with closed-loop data was successful for the linear and nonlinear systems examined. The utility of the model is a function of the dynamical history of the process. When the information content of the data is sufficient, the network can capture the most important features of system behavior so that fine tuning can be based on optimal parameters such as integral absolute error. This method offers a more complete picture of tuning options than that of other fine tuning techniques such as trial and error, which are not based on a system model.

The techniques developed in the tuning effort may be extended to closed-loop model identification for the purpose of controller redesign. In this case, successful identification probably depends on the continuous on-line identification to correct for modeling error.
To my wife,

May we always be free in each others embrace.
ACKNOWLEDGEMENTS

The contributions of the following persons and organizations are gratefully acknowledged:

Professor Ka-Yiu San, my thesis advisor, for his guidance and friendship and unselfish approach to his students.

Professors John Clark, Sam Davis and Kyriacos Zygourakis for their suggestions and criticisms and valuable time as members of my thesis committee.

Department of Chemical Engineering at Rice University, professors and fellow students, who are the type of people that make Rice the wonderful institution that it is.
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of Figures</td>
<td>x</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xvii</td>
</tr>
<tr>
<td>Preface</td>
<td>xx</td>
</tr>
<tr>
<td>PART 1 SYSTEM IDENTIFICATION USING BACK-PROPAGATION</td>
<td></td>
</tr>
<tr>
<td>I. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>II. Back-Propagation Neural Networks</td>
<td>5</td>
</tr>
<tr>
<td>A. Introduction</td>
<td>5</td>
</tr>
<tr>
<td>B. Properties</td>
<td>5</td>
</tr>
<tr>
<td>C. Internal Workings</td>
<td>7</td>
</tr>
<tr>
<td>D. Operation and Application</td>
<td>15</td>
</tr>
<tr>
<td>III. Practical Limitations of Conventional Linear Identification</td>
<td>17</td>
</tr>
<tr>
<td>IV. Advantages of BP Network Identification</td>
<td>20</td>
</tr>
<tr>
<td>V. Disadvantages of BP Network Identification</td>
<td>24</td>
</tr>
<tr>
<td>VI. BP Networks and Control</td>
<td>28</td>
</tr>
<tr>
<td>VII. BP Identification Basics: Model System</td>
<td>31</td>
</tr>
<tr>
<td>VIII. Back-Propagation Model</td>
<td>37</td>
</tr>
<tr>
<td>A. Closed-Loop Identification</td>
<td>37</td>
</tr>
<tr>
<td>B. Identification Tasks</td>
<td>39</td>
</tr>
<tr>
<td>1. Variable Selection</td>
<td>41</td>
</tr>
<tr>
<td>2. Data Manipulation</td>
<td>43</td>
</tr>
<tr>
<td>a. Vector Pair Formation</td>
<td>45</td>
</tr>
<tr>
<td>b. Normalization</td>
<td>47</td>
</tr>
<tr>
<td>3. Time History Length</td>
<td>47</td>
</tr>
</tbody>
</table>
PART 1. VIII. (Continued)

4. Training Set Construction 50
5. Screening 51
6. Training and Updating 56
7. Depth of Learning and Tested Training 65
8. Training Vector Removal 74

IX. Simulation With the Network Model 76
   A. Introduction 76
   B. Simulation Results and Discussions 81
      1. Linear System with Inlet Measured 81
      2. Nonlinear System with Measured Inlet Flow Rate. 86
      3. Linear and Nonlinear Systems with Unmeasured Inlet Flow Rate 94
   4. First Order System 101
      a. Open-Loop Mapping 102
      b. 1st Order Closed-Loop Mapping 105
   C. Discussions 107

Part 2. MODEL APPLICATION
I. Introduction 109
II. Model-Based Tuning 110
   A. Background 110
PART 2. II (Continued)

B. Simulated Systems

1. Linear Simulated System
   a. Grid Search
   b. Disturbance Tests
   c. Comments and Conclusions

2. Nonlinear Simulated Systems
   a. Grid Search
   b. Second Training Cycle
      1. Grid Search
      2. Disturbance Tests
      3. Conclusion and Summary

C. Experimental Systems

1. Linear Level Control Experiment
   a. Controller Variable Squared Error (CVSE)
      179
   b. Total Error (TE)
      185

2. Nonlinear Level Control Experiment
   a. Upper Disturbance
      1. Experimental Results
      2. Network Results
      3. Tuned Responses
   b. Lower Disturbance
      1. Experimental IAE and CVSE
      2. Network IAE and CVSE

PART 2. II. C. 2. b. (Continued)
3. Nonlinear TE
   c. Conclusions

II. Controller Optimization

PART 3. THESIS REVIEW
   I. Summary
   II. Conclusion

REFERENCES
LIST OF FIGURES

Figure 1  
Schematic of Back-Propagation Neuron  

Figure 2  
Back-Propagation Neural Network Schematic  

Figure 3  
Schematic of PI Liquid Level Control  

Figure 4  
Closed-Loop Identification Variables  
Training System  

Figure 5  
Network Identification  

Figure 6  
Data Manipulation  
Training System  

Figure 7  
Input/Output Vector Pair  

Figure 8.  
Screening Example  
Cylindrical Tank System  

Figure 9  
Information Flow For Tuning Application  

Figure 10  
Network Representation of Step in Set Point  
vs. Number of Vectors in Training Set  

Figure 11  
Idealized Training Set Error (TSE) Behavior vs.  
Number of Vectors in Training Set  

Figure 12  
Test Set Total Error (TSTE) vs. Number of Testing Cycles  

Figure 13  
Network Training - Logic Diagram  

Figure 14  
Effect of Tested Training on Response
Figure 15
One Step Ahead Behavior

Figure 16
Recursive Simulation Using BP Network

Figure 17
Response of Linear System to Step Down in Set Point
Inlet Flow Rate Measured

Figure 18
Response of Linear System to Step in Inlet Flow Rate
Inlet Flow Rate Measured

Figure 19
Comparison of Responses of Linear and Nonlinear Systems
Step Down in Set Point in Upper Portion of Tank
Inlet Flow Rate Measured

Figure 20
Comparison of Responses of Linear & Nonlinear Systems
Step Down in Set Point in Lower Portion of Tank
Inlet Flow Rate Measured

Figure 21
Response of Nonlinear System
Step Down in Set Point in Middle of Tank
Inlet Flow Rate Measured

Figure 22
Response of Nonlinear System
Step Down in Inlet Flow Rate in Lower Portion of Tank
Inlet Flow Rate Measured

Figure 23
Response of Nonlinear System
Step Down in Set Point in Upper Portion of Tank

Figure 24
Response on Nonlinear System
Step Down in Set Point in Lower Portion of Tank
Inlet Flow Rate Measured

Figure 25
Response of Linear System
Step Down in Set Point in Middle of Tank
Unmeasured Disturbances Not Included in Model
Inlet Flow Rate Unmeasured
Figure 26  98
Response of Linear System
Unmeasured Disturbances Included in Model Identification
Step Down in Set Point in Middle of Tank
Inlet Flow Rate Unmeasured

Figure 27  99
Response of Nonlinear System
Step Down in Set Point in Middle of Tank
Inlet Flow Rate Not Measured

Figure 28  100
Response of Nonlinear System
Step Up in Set Point in Lower Tank
Inlet Flow Rate Not Measured

Figure 29  103
Open-loop Response to Step in Controlled Outlet
Cylindrical Tank - Controlled and Uncontrolled Outlets
Inlet Flow Rate Unmeasured

Figure 30  106
Response of 1st Order Linear System
Inlet Flow Rate Unmeasured

Figure 31  114
Information Flow
Network Model Reference Tuning

Figure 32  115
Network Model Reference Tuning
Procedure

Figure 33  122
Integral Absolute Error (IAE) vs. $k_c$ and $\tau_i$
Linear Level Control in Cylindrical Tank
Disturbance - Step Down in Set Point From 0.75 to 0.4
Discretized Differential Equation Model

Figure 34  125
Integral Absolute Error (IAE) vs. $k_c$ and $\tau_i$
Linear Level Control in Cylindrical Tank
Disturbance - Step Down in Set Point From 0.75 to 0.4
Neural Network Model

Figure 35  128
Comparison of Responses for Different Tunings
Step Up in Set Point
Simulation of Linear System
Figure 36
Comparison of Responses for Different Tunings
Step Down in Inlet Flow Rate
Simulation of Linear System

Figure 37
Integral Absolute Error (IAE) vs. $k_p$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Discretized Differential Equation Model

Figure 38
Integral Absolute Error (IAE) vs. $k_p$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Neural Network Model

Figure 39
Integral Absolute Error (IAE) vs. $k_p$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Discretized Differential Equation Model

Figure 40
Integral Absolute Error (IAE) vs. $k_p$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Network Model

Figure 41
Integral Absolute Error (IAE) vs. $k_p$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Sum of IAE for Upper and Lower Tank Disturbance
Network Model

Figure 42
Integral Absolute Error (IAE) vs. $k_p$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Network Model - Second Identification

Figure 43
IAE vs. $k_p$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Network Model - Second Identification

Figure 44
Comparison of Responses For Different Tunings
Step Down in Inlet in Lower Tank
Simulated Nonlinear System
Figure 45
Comparison of Responses For Different Tunings
Step Up in Inlet in Upper Tank
Simulated Nonlinear System

Figure 46
Comparison of Responses For Different Tunings
Step Down in Set Point in Lower Tank
Simulated Nonlinear System

Figure 47
Comparison of Responses For Different Tunings
Step Up in Set Point in Upper Tank
Simulated Nonlinear System

Figure 48
Sensitivity of Response Near IAE Global Minimum Tuning
Step Up in Set Point in Upper Tank
Simulated Nonlinear System

Figure 49
Schematic of Experiment

Figure 50
Experimental vs. Simulated Response
Step Up In Set Point
Linear Level Control Experiment

Figure 51
Integral Absolute Error (IAE) vs. $k_o$ and $\tau_u$
Experiment - Linear Level Control in Cylindrical Tank
Set Point Step From 0.5 to 0.75

Figure 52
IAE vs. $k_o$ and $\tau_u$
Experiment - Linear Level Control in Cylindrical Tank
Set Point Step From 0.5 to 0.75

Figure 53
Comparison of Responses to Step Up in Set Point
Initial vs. Experimental and Network Optimal Tunings
Linear Level Control Experiment

Figure 54
Comparison of Responses to Step Up in Inlet Flow Rate
Initial vs. Experimental and Network Optimal Tunings
Linear Level Control Experiment
Figure 55 181
CVSE vs. $k_C$ and $\tau_u$
Experiment - Linear Level Control in Cylindrical Tank

Figure 56 183
CVSE vs. $k_C$ and $\tau_u$
Simulation - Linear Level Control in Cylindrical Tank
Network Trained With Experimental Data

Figure 57 187
Optimal Tuning Parameters vs. Weighting Factor

Figure 58 190
Comparison of Liquid Level Responses for Different Tunings
Linear Level Control Experiment

Figure 59 192
Comparison of Outlet Flow Rate Responses for Different Tunings
Linear Level Control Experiment

Figure 60 196
IAE vs. $k_C$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Experimental Results

Figure 61 198
CVSE vs. $k_C$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Experimental Results

Figure 62 201
IAE vs. $k_C$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Network Simulation

Figure 63 203
CVSE vs. $k_C$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Network Simulation

Figure 64 206
Comparison of Responses to Step Up in Set Point
Initial vs. Optimum Tuning
Nonlinear Experiment
Figure 65
Comparison of Responses to Step Up in Inlet Flow Rate
Initial vs. Optimum Tuning
Nonlinear Experiment

Figure 66
IAE vs. $k_c$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Experimental Results

Figure 67
CVSE vs. $k_c$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Experimental Results

Figure 68
IAE vs. $k_c$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Network Simulation (net trained with experimental data)

Figure 69
CVSE vs. $k_c$ and $\tau_u$
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Network Simulation (net trained with experimental data)

Figure 70
Comparison of Responses to Step Up in Set Point
IAE Optimum vs. TE Optimum (Upper Tank)
Nonlinear Experiment

Figure 71
Comparison of Responses to Step Up in Set Point
IAE Optimum vs. TE Optimum (Lower Tank)
Nonlinear Experiment

Figure 72
Optimal Network Compensator
LIST OF TABLES

Table 1.
Neuron Transfer Functions 11

Table 2.
PI Control of Liquid Level - Simulation Specifications 34

Table 3.
Training Disturbances - Computer Simulated Systems 83

Table 4
IAE of Linear Discretized Computer Model
Step Down in Set Point from 0.75 to 0.4
Inlet Flow Rate, 0.5
Global Minimum at $k_c$, 6.5 and $\tau_u$, 0.5 123

Table 5
IAE of Network Model
Step Down in Set Point from 0.75 to 0.4
Inlet Flow Rate, 0.5
Global Minimum at $k_c$, 4.5 and $\tau_u$, 0.5 126

Table 6
IAE Nonlinear Tank
Discretized Differential Equation Model
Upper Tank Disturbance 134

Table 7
IAE Nonlinear Tank
Network Simulation
Upper Disturbance 136

Table 8
IAE Nonlinear Tank
Discretized Differential Equation Model
Lower Disturbance 139

Table 9
IAE Nonlinear Tank
Network Simulation
Lower Disturbance 141

Table 10
Combined IAE Nonlinear Tank
Sum of Upper and Lower Disturbances
Network Simulation 144
Table 11  
IAE Nonlinear Tank  
Upper Disturbance  
Network Simulation - Second Identification

Table 12  
IAE Nonlinear Tank  
Lower Tank Disturbance  
Network Simulation - Second Identification

Table 13  
Experimental Equipment Specifications

Table 14  
Training Disturbances - Experimental Systems

Table 15  
Experimental IAE Values for Linear Experiment

Table 16  
Simulated IAE values for Linear Experiment  
Network Trained with Experimental Data

Table 17  
CVSE Values for Linear Experiment  
Experimental Results

Table 18  
CVSE values for Linear Experiment  
Network Simulation

Table 19  
Experimental IAE for Nonlinear Level Control  
Upper Tank Disturbance

Table 20  
Experimental CVSE for Nonlinear Level Control  
Upper Tank Disturbance

Table 21  
Simulated IAE for Nonlinear Level Control  
Upper Tank Disturbance  
Network Trained With Experimental Data

Table 22  
Simulated CVSE for Nonlinear Level Control  
Upper Tank Disturbance  
Network Trained With Experimental Data
Table 23
Experimental IAE for Nonlinear Level Control
Lower Tank Disturbance

Table 24
Experimental CVSE for Nonlinear Level Control
Lower Tank Disturbance

Table 25
Simulated IAE for Nonlinear Level Control
Lower Tank Disturbance
Network Trained With Experimental Data

Table 26
Simulated CVSE for Nonlinear Level Control
Lower Tank Disturbance
Network Trained With Experimental Data

Table 27
Optimum Tuning Parameters vs. Weighting Factor
Disturbance in Upper Tank

Table 28
Optimum Tuning Parameters vs. Weighting Factor
Disturbance in Lower Tank
PREFACE

This thesis discusses the application of back-propagation neural network technology to system identification and process control. The work is divided into three parts. The first part concerns the study of network identification techniques, and the goal is to obtain an accurate process system model. The second part involves applications of the network model, and the aim is to demonstrate uses of the network model which can improve process control. The last part is the general thesis summary and conclusion.
PART 1. SYSTEM IDENTIFICATION USING BACK-PROPAGATION NEURAL NETWORKS

I. Introduction

The back-propagation algorithm belongs to the discipline called artificial intelligence. It is a member of a class of software mapping tools known as neural networks. As of 1988, at least 50 different types of neural network algorithms were employed (Hecht-Nielsen, a, 1988) and more continue to be added. Some of these algorithms, like adaptive resonance, classify incoming data without the need to specify class divisions (Carpenter and Grossberg, 1988). Others such as bidirectional associative memory reconstitute an entire remembrance from pieces of the memory (Kosko, 1987) and still others such as counter propagation are pattern matchers which map a set of inputs to a set of outputs (Hecht-Nielsen, a & b, 1987). Back-propagation (BP) belongs the latter group. The most important features are its ability to form nonlinear mappings and to learn by example. It is the most popular neural network and has been employed in a multitude of applications. A few diverse examples are recognition of sonar targets (Liou, 1988), conversion of written text to sounds (Sejnowski, 1987), and prediction of natural gas market behavior
(Werbos, 1988). In this work, back-propagation is used to identify a model which will match a pattern of past input variables from a system to a prediction of future system behavior.

The techniques for applying the BP network to identification are not well formulated and various techniques are devised. The use of this type of artificial intelligence technology offers the potential of continuous on-line learning about the system to be controlled. Since the network is capable of nonlinear mapping, it offers the potential for nonlinear identification. In particular, back-propagation is used to structure linear and nonlinear closed-loop models that are used to fine tune a proportional integral controller.

Most current identification techniques have the limitations of being linear and time-invariant. Real processes are usually nonlinear and vary with time. Some recent progress has been made in nonlinear system identification, however the problem is still unresolved (Hunt, et. al., 1983; Kravaris and Chang, 1986; Alsop and Edgar, 1986; Kantor, 1986). The foremost technique for identification of nonlinear systems involves approximating the actual system by a linear one. This linear model is valid near the linearization point, however when the system is highly nonlinear, several linearization points
are selected which span the process operating range. The matrices that arise from this identification process are usually determined by analyzing data from an open-loop test. These matrices remain fixed until another test can be performed. If the process changes during this period, the system model remains fixed and is effectively time invariant for the period.

Current identification techniques are manpower intensive, disruptive to process operations and require specialized expertise. Significant amounts of manpower are required to perform the necessary tests and to verify the resulting model. The tests are disruptive to operations, because they are usually open-loop which requires disconnection of the controller. As a result, participation of operations personnel is mandatory to keep the process stable and within test specifications. Finally, the procedures involved in identification are not simple and require specialized expertise.

Application of BP networks to identification does not automatically promise to remove these negative issues, however it does have some powerful capabilities which make it attractive for identification purposes. The BP network has the capacity to learn so that continuous identification from closed-loop data can be arranged. It has the ability to structure nonlinear mappings, so that
the actual system model can be mapped rather than a linearized version. It has an internal mean square filter which allows it to accept unfiltered data directly from the process. It does not require the specification of a functional form for the mapping, since this is handled internally by the network. Fitting of the data is also handled internally. The above properties of the BP network reduce the identification tasks to specification of the pertinent variables and diagnostic checking and verification. A more detailed discussion of the advantages and disadvantages of BP identification will be presented in Part 1 IV and V.
Part 1. II. **Back-Propagation Neural Networks**

Part 1. II A. Introduction

The back-propagation algorithm is computer software that maps a matrix of inputs to a matrix of outputs. It is called a neural network not because it acts like a neuron in a biological brain, but because it was originally created to explore theories concerning the manner in which biological neurons interact (Rumelhart and McClelland, 1987). The name back-propagation originated from the manner in which adjustable parameters in the network are changed (Klimasauskas and Guiver, 1988). Specifically, the error between the actual output and the network’s output is propagated back through the network and used to alter the adjustable parameters in a manner that improves the mapping.

Part 1. II. B. Properties

The attractive properties of back-propagation in terms of system identification are its ability to learn by example and to structure nonlinear mappings. If a BP network is shown a consistent set of examples of input data and corresponding output data, it can be trained to
form a mapping for this set. Afterward, when the input portions of the training set are presented, the network can reproduce the corresponding outputs to an arbitrarily close degree of accuracy. In other words, the BP network is an excellent mimic of the input/output relationship contained in the training set. In addition to this capacity, it is able to interpolate and extrapolate from the examples in this training set. As such it has limited powers of generalization.

In order to apply the BP network's mapping abilities to system identification, a set of examples is needed for training purposes. Data from the process is used to form these examples. As more examples of different dynamic situations are included in the network's training set, the mapping more closely approaches that of the actual system. This is akin to human experience, in that understanding of a system's behavior improves as experience with the system grows.

Another property that makes back-propagation attractive is its ability to form nonlinear mappings. If a function can be normalized in the absolute range of zero to one, then there exists a three-layered back-propagation neural network that can approximate the function to within an arbitrarily small mean-squared-error. This statement is a paraphrase of a proven mathematical theorem which was
introduced by Kolmogorov and refined by Hecht-Nielsen (Kolmogorov, 1958; Hecht-Nielsen, 1988). Mappings encountered in process identification usually can be normalized in this manner, so that back-propagation can be applied. The theorem does not restrict the class of functions. Both linear and nonlinear functions can be represented in this manner. Since most real processes are nonlinear, this particular property introduces the prospect of unstructured nonlinear identification.

As an aside, evidence exists that BP networks may be only a subclass of mathematical transformations that can be described by classical approximation techniques such as generalized splines and regularization theory (Poggio and Girosi, 1990). The anchoring of the algorithm firmly in mathematical theory does not diminish its usefulness in its current form, and will hopefully speed the improvement of its internal workings in the future.

Part 1. II. C. Internal Workings

A BP network consists of neurons, connections and weights. These components can be created in computer hardware as electronic or optical or some other "hard-coded" entity (Eliot, 1987). In order to used the
conventional computer hardware available for this study, they are instead simulated in software.

A schematic of a neuron is shown in Figure 1. A neuron is a node where signals from neighboring neurons are received, summed and processed to yield an output signal. The connections designate which neurons communicate and the order in which they do so. The output from a neuron is sometimes called its activation. Activation signals from a sending neuron travel down the connection to a receiving neuron. Before the activation is received, it is multiplied by the weight associated with the connection. A weight can take on any real value. The product of the output and the weight is the signal transmitted on the connection to a receiving neuron. The sum of the inputs into a neuron is called the net input. If the net input is above an upper threshold value, the neuron fully fires and emits a signal whose magnitude is one. On the other hand, if it is below a lower threshold, the neuron does not fire and has zero output. If it is somewhere in between the thresholds, an intermediate fractional output results.

The strength of the signal emitted by a neuron is determined by its activation function, sometimes called a transfer function. In this study, the transfer function is either a unity function or a particular sigmoidal
Figure 1
Neuron

![Neuron Diagram]

- Inputs
- \( \sum \) Input
- Net
- Transfer Function
- Output
function called a logistic. Equations which characterize these functions are listed in Table 1. If the transfer function is unity, then the net input is passed unchanged to the output. If the transfer function is sigmoidal, then a net input greater than an upper threshold of approximately +3 yields an output of nearly one, and a net input below a lower threshold of -3 results in an output nearly zero. This on/off behavior is inherent to the mapping success of the BP neural net (Lapedes and Farber, 1987). A neuron's output signal is duplicated and disseminated to each connection at its output.

Weights are the parameters adjusted during the learning process. Their collective values actually store the knowledge of the mapping. They can take on any positive or negative value and are analogous to parameters of a curve fitting routine.

In this work, the neuron arrangement is in a standard four layer configuration for back-propagation, namely the input, hidden, output and bias layers. A schematic of this arrangement is shown in Figure 2. A vector or matrix of data which is normalized between zero and one is submitted to the input layer to be processed in succession by the hidden and output layers. For this project, input data is organized in a vector form. Only a single hidden layer is shown in the figure, but extra layers can be
Table 1  
Neuron Transfer Functions

Unity Transfer Function

\[ \text{NEURON OUTPUT} = 1 \times \Sigma \text{NEURON INPUT} \]

Sigmoid Transfer Function

\[ \text{NEURON OUTPUT} = \frac{1}{1 + \exp(-\Sigma \text{NEURON INPUT})} \]
FIGURE 2
Back-Propagation Neural Network

connections with adjustable weights

Input

... Hidden Layer(s)

Output Layer

Output

○ input neuron
○ hidden or output neuron
○ bias neuron
added to increase the degrees of freedom of the mapping. This may be advantageous for applications such as image processing, but for the mappings encountered in process identification for this study, a single layer is found to be sufficient. Here, the output layer is shown as a single neuron, however it can be any number of neurons arranged in either vector or matrix form.

As depicted previously in Figure 2, every input layer neuron is connected to every hidden layer neuron. The two layers are said to be fully connected. Likewise, the hidden layer and output layer are fully connected. The input and output layers are not connected at all. Bias neurons are singly connected to each member of the hidden and output layer, but are not connected to input neurons.

The type of neuron transfer function for each member of the network must be specified. Input neurons and biases have unity transfer functions. They simply pass through the net input and disseminate copies to the connections at their output. Neurons in the hidden and output layer have sigmoidal transfer functions and thus exhibit on/off behavior.

When the input layer receives a normalized vector, each neuron passes the corresponding vector component unchanged to the connection at the output of the neuron. These connections lead to the hidden layer. Before the
signals from the input layer are received by the hidden layer, they are multiplied by the respective weights associated with the connections between the two layers. Each hidden layer neuron also receives input from its associated bias neuron. The combined signal from the input and bias neurons into the hidden layer neuron is its net input. The net input is the variable to be used in the transfer function to determine the neuron's output. This output is duplicated and disseminated to each connection between this neuron and those of the output layer. Before the signal is received by an output layer neuron, it is multiplied by the weights between the two layers.

All layers of the network have been discussed in detail except the bias layer. This layer has one neuron for every associated neuron in the hidden and output layers. Each bias neuron has a single connection to its associated neuron. The input to every bias neuron can be assumed to be one and its transfer function is unity, so that its output is always one. The resulting input into the associated neuron from the bias neuron simply equals the value of weight on the bias connection.

The presence of a bias neuron creates a situation equivalent to that of the associated neuron having an adjustable threshold. This equivalency is shown in
Rumelhart and McClelland (1987 and 1988), but it will be demonstrated by the following example. Suppose that the net input to a neuron with a sigmoid transfer function sums to -3 and further suppose that the bias input is zero, which is equivalent to having no bias layer. Given the nature of the sigmoid transfer function, this net input will not cause the neuron to fire. However, when the bias input to this neuron is made to be 6, the total net input changes from -3 to +3 and the neuron fires. This situation can be thought of in another light, in that without the bias, the threshold for full firing is 3 and with the bias the threshold is lowered to -3. Since the bias weight is fully adjustable, the bias input provides the same effect as an adjustable threshold.

PART 1. II D. Operation and Application

BP networks operate in two distinct modes, learning and reflection. The first mode is a relatively lengthy process while the latter is practically instantaneous. The net learns by sequentially examining a series of inputs and corresponding outputs called a training set. This set is constructed from the experimental data or as in the first section of this study from a physical model of the system to be controlled. During learning, the net
compares desired output called the target with actual output. The weights are adjusted in a manner that allows a gradient descent in an error parameter, the sum of the root-mean-squared error between the network reflections and respective targets. A detailed explanation of this method for adjusting the weights is given in the McClelland and Rumelhart references. The network slowly but eventually steps toward the desired mapping to an extent only limited by the properties of gradient descent and the amount of computer time invested.

In the reflection mode, the weights are fixed. A vector or matrix of normalized inputs is presented to the input layer. The signal is processed, with each neuron behaving in the manner previously described to yield an output. The calculations involve simple summations and transfer function computations, and the results are practically instantaneous. The output is a matrix or vector of normalized values.
PART 1. III. Practical Limitations of Conventional Linear Identification

Most advanced control algorithms depend on a mathematical model of the system to be controlled. Consequently, an algorithms performance heavily relies on model accuracy. Ideally, the desired model is one that mimics the full range of behavior of the real system in the most accurate manner possible.

In theory, this model can be approached to a satisfactory degree. However, manpower considerations usually make the effort impractical. In particular, the manpower necessary to construct and maintain such a model may not be available. The process of structuring and verifying process models is labor intensive. Model development and validation often account for 75% of the expenditures in advanced control projects. (Bhat, et al., 1988). In industry, this type of work may not be as high an economic priority as other engineering activities.

Even if the manpower is available to construct the model, the process may change with time and so must its model. The task of updating a model requires a consistent commitment of manpower. In addition, differences between even similar types of process units may render construction of a generic model impossible. In other
words, each processing unit may need a separate model which further increases the scope of work involved.

Even with these drawbacks, when accurate control is sufficiently important to warrant the effort, models are constructed and model-based control is implemented. The process model is usually not a physical model but a discrete time series model built from process data. Model construction involves statistical analysis of test data which is collected from the process in an open-loop test. One such test involves a pseudo-random binary disturbance that is formulated to perturb the process but keep it stable and near the average set point. The test signal must excite the system in the proper manner to produce data that is rich in the information needed in model construction. Not all process perturbations have this property.

Once the data is collected, it is usually filtered for noise and the trend is removed. It is then statistically analyzed to identify the proper historical window for each process variable. The history of a variable is a collection of measurements of that variable a number of time steps back in time. The historical period or window is the number of steps back in time necessary to be able to predict future process behavior. Once this is determined, model parameters are determined
usually by employing least-squared approach. Finally, the model must then be tested and verified.

The identification process oftentimes can get quite involved. The model's accuracy and usefulness depends on the knowledge and skill of the persons collecting, analyzing and testing the data. Such expertise is probably beyond the scope of the typical chemical or industrial engineer and may require a special consultant.

In summary, even though model-based control can offer improved performance, the expense, effort and complication involved usually limits its application. In fact, less than 10% of all chemical processes were controlled in this manner in 1982 (Garcia and Morari, 1982) and the percentage is still no more than 20% today (Garcia, 1990).
PART 1. IV. Advantages of BP Network Identification

BP neural nets offer a way of addressing some of the limitations of process identification and thus potentially enabling greater use of model-based control techniques. First, networks allow easy automation of model construction. A network can be structured that determines when on-line data from a processing unit is acceptable, incorporates the data into a model and uses the model either directly or indirectly in control. Although application of network modeling to control requires some labor, the effort is small in comparison to that required to implement conventional model-based control.

A second advantage is that networks can be structured which automatically update themselves when a change in the process occurs. Changes are due to such phenomena as fouling or catalyst degradation. When a process changes, so must its control system model. If a network identification system is designed to detect this change, then it can relearn and update the system model. Again, this updating process can be fully automated.

Third, the net's model is robust in the presence of white noise. The term "robust" in this sense means that even though the incoming data is noisy, the model still accurately represents the dynamic responses of the system.
A BP network acts as a least-squares filter to white noise (Klimasauskas, 1989). White noise generates inconsistent mapping relationships. If enough of these relationships are included in the networks training set, then the noisy portion cancels and the useful information remains. A network can screen a considerable amount of white noise, but can not screen biased noise.

A fourth advantage is the simplicity of model construction. The decisions necessary to structure a neural network model are relatively few and fairly easy to determine. For example, the width of the time history of the variables in the mapping must be specified. This can be determined from disturbance response data or from assumptions about the order of the closed-loop process and a knowledge of process delay. Another factor is the normalization constants for the appropriate variables. Since every process parameter has a maximum rating which is usually appropriate as the normalization constant. Once the methodology for neural network mapping is structured, a practicing engineer should be able to apply it in a fairly straightforward manner in comparison to conventional identification.

A final advantage is the possibility of nonlinear identification. Current identification techniques are based on linearization of a nonlinear system model. In
particular, as the process set point diverges from the state point of linearization, the discrepancy between the linear model and the true nonlinear system increases. Identification based on the actual nonlinear model is more desirable, but this would require construction of the actual model and all the effort this entails. Automatic construction of the nonlinear model by a BP net would allow model-based nonlinear control.

The capacity of the neural network to accommodate identification of a nonlinear system has been examined and compared to a conventional linear identification method known as Finite Impulse Response or FIR (Haesloop and Holt, 1990). In every case examined, the nonlinear network model performed better than the FIR approach. The degree to which the nonlinear mapping ability of the network is important depends on the degree of nonlinearity of the system.

As a final note, the BP network is not limited to single-input-single output (SISO) type mappings. Any number of inputs and outputs can be selected. If a unique relationship exists between the variables chosen, then the BP network is capable of capturing the representation. This means that not only can the mappings be nonlinear, but they can be multiple-input-multiple output (MIMO), which are the type most often encountered in industrial
practice. Their study may be considered in future efforts, but will not be considered in detail in this work.
PART 1. V. Disadvantages of BP Network Identification

Several potential disadvantages have been identified. First, an ignorant network will yield poor results. When a network is not trained in the proper manner, it does not possess enough knowledge about system behavior and will make identification errors. Proper training entails two aspects, namely that the training set contains a sample of process data which represents the complete dynamic range of the system and that the learning process progresses to a satisfactory point. The first point can be illustrated by considering a system that is near steady state most of the time. Since the data gathered during this period contains little information on system dynamic behavior, a network trained solely on such data will perform poorly in predicting dynamic system responses to disturbances. The quality of the dynamical situations to which the network is exposed determines the quality of the model. This is akin to human experience in that the depth of one's "understanding" of the process depends on one's experience with the process behavior.

Like human experience, the network must be able to at least replay what it is taught. In addition, it is also desirable for the net be able to interpolate and extrapolate using the examples in its training set in
order to encompass situations that it has not encountered. The capacity of a BP network to interpolate and extrapolate is limited. It can usually follow a general trend, but will generate an offset from the correct value if asked to extrapolate too far. When the system enters an operating region on which the network has no knowledge, then the network model will probably perform poorly.

Even if the training set is properly representative, the network model may still be an inadequate if the network is not allowed to learn to the fullest extent that is beneficial. In particular, the depth to which the network is programmed to learn is also an important factor. If the network learns too deeply with a limited amount of information, then the mapping may actually degrade in accuracy. This issue will be addressed in detail later in Part 1 VIII.

A second disadvantage is the computer time necessary for the network to learn. For computers with unenhanced hardware, learning can take days on a micro-computer and hours on a mini-computer. If the neural net is starting from an ignorant state where the weights are randomly set and is expected to learn at approximately the same rate at which new information arises from the process, then current computer technology is too slow even at mini-computer speeds. However, the situation is changing very
rapidly and processing speeds and learning times are diminishing. Computer boards and computer chips which are dedicated to neural network learning and employ parallel processing accelerate learning rates by a factor of 10 to 20 (SAIC, Inc., 1989). In addition, new faster back-propagation algorithms have been devised and are currently being tested and are slowly making their way to greater use (Himmelblau, 1990). The cost of specialized hardware is presently still too high to make the technology practical in most control situations. Yet, hardware and software cost reductions and computer advances in the near future may partially mitigate the speed problem.

A last disadvantage deals with the unconventional nature of the neural net mapping. Conventional model-based control techniques are usually couched in the language of matrices. Control law implementation depends on matrix inversion. Neural net mappings are nonlinear transformations that cannot be restated into matrix format. In addition, they are not yet invertable. For this reason, network mappings do not fit easily into the framework of current advanced control research. The non-invertability of the network means that the exact solution using the network model can not be obtained in a direct manner. Instead, a solution must be proposed, and the
model used to either affirm or dispose of the proposal. This is a "what if" approach to the solution.
PART 1. VI. BP Networks and Control

The history of back-propagation and its application to process control is relatively short. According to Hecht-Nielsen, the algorithm was originally introduced by Bryson and Ho in 1969 (Bryson and Ho, 1969) and independently rediscovered by a number of other investigators including Werbos in 1975 (Werbos, 1974), Parker in the mid to late 1980’s (Parker, 1985; Parker, 1986; Parker, 1987) and Rumelhart, Williams and other members of a group known as Parallel Distributed Processing or PDP in 1985. The reason that the algorithm had so many co-inventors is because these researchers come from diverse backgrounds and communication does not flow naturally between their respective disciplines. The credit for developing back-propagation into a useable tool rests heavily with Rumelhart, Hinton and Williams (Rumelhart, et. al., 1986) and the PDP group (McClelland and Rumelhart, 1988). Before their work, the algorithm was unappreciated and unused. Today, it is the principal choice for neural computing applications.

Shortly after the back-propagation algorithm was publicized, control researchers began considering its application. Leading the way are many robotics researchers (Recent Publications: Liu, et. al., 1989;
Kuperstein and Rubinstein, 1989; Passino, et. al., 1989; Anderson, 1989; Yeung and Bekey, 1990; Iberall, 1990; Bulloch, 1990; Guez and Bar-Kana, 1990; Fadali, et. al., 1990). Robotics is a natural application for neural network control since the relationship of the robotic components are difficult to model exactly. In addition, open-loop data is almost always available with little harm to the system. Robotic applications are usually made based on rudimentary physical models of the system that aide in selecting the variables which are necessary for the mapping. Robotic applications of neural network technology are usually highly tailored to the robotics tasks and are often not directly applicable to the problems encountered in chemical process control.

The philosophy of using networks in control applications is presented by several researchers in the control area (Psaltis, et. al., 1988; Levin, et. al., 1989). These articles apply well known control learning architectures in outlining possible approaches to network adaptive control. The studies are general in nature, and specifics of network applications are not presented. A more comprehensive analysis of the role of BP networks in system identification, particularly in nonlinear system identification, is given in an article by Narendra and Parthasarathy (Narendra and Parthasarathy, 1990). The
authors examine the use of networks as reference models for controller algorithm synthesis in nonlinear systems that are inherently stable. The major portion of the identification of the network model is accomplished using psuedo-random binary signal as input to the open-loop system. Subsequently, closed-loop identification is used to improve the model. Simulation of the responses of various nonlinear systems to sinusoidal disturbances is examined. Most of the performance results show that this approach is promising for stable systems. Additional research is recommended for unstable system identification.

Applications of back-propagation in chemical process controls have included sensor interpretation (Mc Avoy, et. al., 1989), detection of sensor failure (Naidu and Mc Avoy, 1989), dynamic modeling (Bhat and Mc Avoy, 1989) and in distillation control system design (Birky and Mc Avoy, 1988). Some preliminary examination of the mapping quality of BP networks as applied to optimal predictive control has been discussed (Donat, et al., 1989). Additional examination of the prospect to replace the system model in Internal Model Control has been examined (Hernandez and Arkun, 1990).
PART 1 VII. **BP Identification Basics : Model System**

Several aspects of identification will be addressed in this section. First, to allow for the maximum historical perspective on the system behavior, all measured variables are included in the mapping if possible. Next, a mapping structure must be selected; i.e., the determination of the input and output variable and their arrangement in the mapping. A particularly useful structure employs current and past data to predict the behavior of the controlled variables in the next time step. This type of model will be called a one-step-ahead model. It will be useful in analyzing controller tuning and may be appropriate in the development of self-improving controller strategies. In this work, only controller tuning will be discussed in detail, whereas the second topic will be discussed briefly.

A simple chemical process will be used to explore and illustrate the particulars of network identification and modeling application. The process selected is liquid level control in a tank. The selected control algorithm is proportional integral (PI) control. Two separate tanks are analyzed, one with a constant cross sectional area and the other with a cross sectional area that varies with height. The first tank is a linear system and the second
is nonlinear. For the linear system, the process is first order in the open-loop and second order in the closed-loop.

In practice, level control does not require a neural network technology for good performance. This process is selected, because it is well understood and has many of the features needed to structure a methodology appropriate for more complicated applications.

A schematic of the process is shown in Figure 3. The tank has an uncontrolled inlet and controlled outlet. The inlet flow rate represents a process disturbance. The outlet flow rate is the controlled variable. Specifically, the level is controlled by adjusting the RPM of a variable speed pump at the outlet. The controller is tuned by the quarter decay rule so that the response is slightly underdamped. The computer model is formed by discretizing the appropriate material balance differential equations. Specifics of the computer model and model parameters are presented in Table 2.

Data from the level control process will be used to train a network model. The output variable of the network model output is the liquid height one sample period ahead in time. The input variables are windows of all past and present variables. The model will be derived data from
FIGURE 3

PI Level Control
Training System

Inlet, $F_i$

height, $h$

Set Point, $h_{sp}$

DPC

PI Controller

Outlet, $F_o$

Variable Speed Pump
Table 2.

PI Control of Liquid Level - Simulation Specifications

**Variable Definitions and Specifications**

\( h \) = height of level in tank from tank bottom  
\( D \) = tank diameter = 0.8  
\( A \) = tank cross sectional area = 0.5  
\( V \) = tank volume = 0.75  
\( F^+ \) = maximum volumetric outlet flow rate of pump  
\( = 0.225 \) cubic meters/second  
\( F_i \) = volumetric inlet flow rate to tank  
\( F_o \) = volumetric outlet flow rate from tank  
\( t \) = time  
\( \tau \) = system time constant = \( V/F^+ = 3.0 \) seconds  
\( \Delta t_m \) = modeling period = 0.1 seconds  
\( \Delta t_s \) = sampling period = 0.5 seconds  
\( h_{sp} \) = set point liquid level  
\( h_t \) = normalizing height = height of tank = 1.5 meters  
\( K_c \) = controller gain  
\( K_c \) = normalized controller gain = \( K_c h_o/F^+ = 1.5 \)  
\( \tau_{ci} \) = controller time constant = 2.0 seconds

**Differential Equation**

\[ A \frac{dh}{dt} = F_i - F_o \]

**Nonlinear System**

\( D_T \) = diameter of top of tank = 0.4 meters  
\( D_B \) = diameter of bottom of tank = 0.8 meters  
\( D_h \) = diameter of tank at \( h = D_T + [(h_t - h)/h_t] \times (D_B - D_T) \)

\[ A = \pi D_h^2/4 \]

**System Equation - Digital Approximation**

\[ h(k) = h(k-1) + [F_i(k-1) - F_o(k-1)] \times (\Delta t_m/A) \]

**Control Law - PI Digital Approximation (Velocity Form)**

\[ \varepsilon_k = h(k) - h_{sp}(k) \) (negative of standard form of \( \varepsilon_k \))  
\[ F_o(k) = F_o(k-1) + K_c(l + \Delta t_s/\tau_{ci})\varepsilon_k - K_c \varepsilon(k-1) \]
closed-loop data. A schematic of the level control system and the information flow to the network model is shown in Figure 4. Note that in the control schematic the distinction is made between the variables that actually enter the system and those which are measured or sent to the system. The transfer function labelled with a capital M converts the actual signal to measured signal. The function labelled with a capital S converts the sent signal to the actual received signal. These two transfer functions are part of the unmeasured dynamics of the system. If the system being consider is a computer simulation, then the measurement and sent transfer functions may be set to unity, but for an experimental system the transfer functions are usually not unity.
FIGURE 4
Closed-Loop Identification Variables
Training System

Learning Network
One-Step-Ahead Model

- M - actual to measurement conversion
- S - signal to actual conversion

h  = liquid level
h_m  = measured liquid level
h_sp  = liquid level set point
F_i  = inlet flow rate
F_im  = measured inlet flow rate
F_o  = outlet flow rate
F_om  = measured outlet flow rate
PART 1. VIII. **Back-Propagation Model**

The model is discrete and consists of contiguous time histories of all variables. This type of discrete model is also known as an Auto-Regressive Moving Average or ARMA (Jenkins and Watt, 1968). Data from the test system is assumed to be noiseless and without delay. Noise and delay will be encountered in a discussion of the experiments, and their impact will be assessed at that point.

PART 1. VIII A. Closed-Loop Identification

The identification will be performed closed-loop. A controller, like that of PI, must be operating in order to generate the data necessary to formulate the network's training set. Because the controller suppresses the response to a disturbance, less information about system dynamics is available from closed-loop data in comparison to open-loop data. For this reason, closed-loop identification is usually employed on a limited basis. In adaptive controller studies subsequent to open-loop identification, it is sometimes used to fill in the fine structure of the system model. The limited application of closed-loop identification is unfortunate since the
available data is practically unlimited. As a system operates, it continuously generates closed-loop data that may contain information useful for identification purposes. In current industrial practice, the potential of this information is ignored. One of the goals of this study is to determine if it is possible to use this type of data to improve system performance by developing models for tuning purposes.

Box and MacGregor have analyzed some of the pitfalls of closed-loop identification as it applies to optimal control (Box and MacGregor, 1974 and 1976). If the controller is nearly optimal, then a closed-loop identification effort may not be successful. The PI controller may approach this point if it is successfully fine-tuned, however this is rarely the case. In this study, the initial controller tuning is not optimal during the identification process, and identification efforts were all adequately successful. In order to aid the identification effort, the authors recommend that a dither signal be added to the controller signal to enhance the amount of dynamical data available. Since the controllers studied are all suboptimal, this added signal was not found to be a requirement. The authors point out that the identification content of the system data may be enhanced by a change in the control law and by the presence of
noise. Both play roles similar to the dither signal in that they generate additional dynamical information. In the computer simulations which follow, the control law is changed during the identification process by implementing new tuning parameters. This action improves the quality of the identified model. The effect of noise is examined in the experimental section. Some information can be derived from the disturbances caused by noise, however it does not appear to be sufficient to completely structure a successful model.

PART 1. VIII B. Identification Tasks

The identification tasks described in this section were specifically developed as part of this study. They can be grouped into four major activities: data manipulation, screening for candidate vectors, screening for training vectors, and network training. A schematic is shown in Figure 5. Each activity will be explained briefly. Data manipulation includes data normalization into the range of zero to one and formation of this data into a vector form acceptable to the network. Once the vectors are formed they are screened. This entails rating the vectors in terms of an error parameter, which is, in this case, the error between the network predicted output
Figure 5
Network Identification

Process
Raw Data

Data Manipulation
- Vector Formation
- Normalization

Vectors

Screen For Candidate Vectors
- Rate Vectors for Error
- Reject Vectors with Error Below Threshold
- Pass Vectors with Error Above Threshold

New Candidate Set Members

Old Candidate Set

Candidate Set

Screen For Training Vectors
- Rate Vectors for Error
- Reject Vectors with Error Below Threshold
- Select Vector with Largest Error

New Training Set Member

Old Training Set

Training Set

Network Training
- Train for n cycles
- Calculate Test Set Total Error (TSTE)
- Re-train when TSTE Slope Above Limit
- Stop when TSTE Slope Below Limit
- Ship new weights

New Weights
and the actual output. Vectors with error below a threshold are rejected. The remaining vectors become the new members of a candidate set and are combined with any existing members to form the complete set. This set is a collection of potentially interesting vectors. It must be screened a second time to select a new training vector member. The candidate set is re-rated for error when new network weights are available, and the screening process is re-instituted. A single vector selected which is combined with existing training vectors to form the new training set. This set is used to train the network for a number of cycles. The progress of learning is checked periodically by a test which will be explained later. If the test determines that learning is sufficient, then the learning process stops and a new set of weights is transferred to the screening network. Each of these activities and their accompanying tasks will be examined in detail.


Variable selection involves choosing a set of variables that completely characterizes the system. For liquid level control, one such set includes the liquid level, outlet flow rate and set point. If inlet flow rate
is measured, then it can also be included, else its effect will be internally inferred from the measured variables. The complete network input variables are concatenated histories of measured variables which form the input vector. A history of a variable is the past and present values of that variable. The output variable of the mapping is a single item, the height in the next time step. The combination input vector and output variable is known as a vector pair.

Note that the variable set is not unique. In general, any set of variables that completely characterizes the process can be used in the mapping. For example, more fundamental variables can be selected such as the actual voltage measurement from a differential pressure cell which measures liquid height or the actual RPM from the variable speed pump. In some instances, use of such fundamental variables may be a more direct mapping technique.

Another possible variation in the mapping strategy is the use of deviation variables. Since the system is linear, deviation variables can be selected which fully characterize the system. In particular, the input part of the vector pair could consist of histories of error between the set point and height together with a history of the difference of outlet flow rate and some steady-
state reference flow rate. The output portion of the vector pair would be the deviation in the height in the next step. However, this strategy would not have been applicable to nonlinear systems. For such systems, the error history for a particular disturbance is a function of the linearization point so that the mapping is not be consistent over the entire mapping space. Since the ultimate aim is to use neural network mappings in nonlinear systems, this mapping approach is not chosen.

It should be pointed out that additional variables beyond those that minimally characterize the system can be included in the mapping provided these measurements are deemed reliable. These extra variables help to additionally characterize the mapping. In the test example when inlet flow rate is measured, its history can be included as part of the input vector. This is not a necessary part of the mapping for feedback applications, but is required in feed-forward applications.

Part 1. VIII B 2. Data Manipulation

This part of the identification process consists of two steps: vector pair formation and normalization. A schematic of the information flow from process to the two identification tasks is shown in Figure 6. Note that the window in time for each variable is shown as two sample
Figure 6
Data Manipulation
Training System

Process
Sample Time = 1

Process
Sample Time = 2

Process
Sample Time = 3

Vector Pair Formation

Normalization

\[ F_{01}, h_{sp1}, h_1 \]
\[ F_{02}, h_{sp2}, h_2 \]
\[ F_{01}, F_{02}, h_{sp1}, h_{sp2}, h_1, h_2 \]
\[ F_{01}, F_{02}, h_{sp1}, h_{sp2}, h_1, h_2, h_3 \]

\[ h_T \]
\[ F_0 (MAX) \]

\[ F_{01}/F_0 (MAX), F_{02}/F_0 (MAX), h_{sp1}/h_T, h_{sp2}/h_T, h_1/h_T, h_2/h_T, h_3/h_T \]

\( h_i \) = liquid height at sample time i
\( h_{sp} \) = liquid set point at sample time i
\( h_T \) = tank height
\( F_{0i} \) = outlet flow rate of tank at sample time i
\( F_0 (MAX) \) = maximum outlet flow rate of pump
periods for simplicity, but in practice, the window is usually larger.

Part 1. VIII. B. 2. a. Vector Pair Formation

The first step in the data manipulation process is to organize the data into a vector or matrix form. The choice of either form is immaterial to the internal workings of the BP algorithm. For this study, the vector form is selected. An input vector consists of a concatenation of time histories of input variables. The output vector is a single variable, namely the height in the next time step. Note that in multivariable control, the outlet may also be a vector.

As mentioned earlier, the combination of input vector and output vector is known as a vector pair. The order in which the input variables are placed is determined by the user. Once the order is set and learning begins, it must remain in this form throughout. Any rearrangement represents another mapping entirely and requires complete retraining. The exact order selected for the vector pair in this study is shown schematically in Figure 7. The outlet flow rate history is first and is followed by set point and height histories. For feedback mappings, the inlet flow rate time history is not included.
FIGURE 7

Input/Output Vector Pair

Outlet Flow History  Set Point History  Liquid Level History

Liquid Level
One Step into Future

Vector Input  →  Vector Output

Once the vector pairs are formed, each variable must be normalized into a range acceptable for neural network applications. All BP networks operate in an absolute normalization range less than one for convergence reasons (Hecht-Nielsen, 1988). Depending on the neuron transfer function, networks usually have the range of -0.5 to 0.5 (SAIC Inc., 1988) or 0.0 to 1.0. The later is the range used in this study. A simple method for normalization involves division of each variable by its maximum practical value. In level control, the maximum for the liquid level is the tank height. For outlet flow rate, the maximum is the maximum rating of the pump. With these normalization parameters, the mapping input and output is always between zero and one. The key in normalization is to use as much of the normalization range as possible in order to achieve good resolution, but never to exceed the range. A variable out of range causes instability in the BP learning algorithm and results in mapping failure.

Part 1. VIII B 3. Time History Length

Once the mapping format is determined and variables selected and normalized, the length of time history for
each variable must be specified. This is simply the number of measurements back in time which will make up the history of a particular variable. The history lengths for separate variables can be different, however they are all the same length in this work. At a minimum, the history length must be at greater than the closed-loop order of the process. In a real nonlinear process, a psuedo-order can be determined by selecting the order of a linear process that most resembles that of the actual process. The maximum value of time history length is certainly the time necessary for a large disturbance to decay to nearly zero. This value can be quite large. As a result, some value between the maximum and minimum length must be selected.

For PI level control, the order of the closed loop system is two. The time necessary for a large disturbance to decay is dependent on the controller parameters but is between 10 to 20 sample periods. The actual window chosen is five which is slightly more than twice the closed-loop order.

The key in choosing the window is to select the length slightly larger than is necessary, but not too large. If the choice is too small, then the mapping is invalid from fundamental modeling considerations, and any mapping attempt will ultimately fail. When the window is
too large, many elements of the input vector actually have negligible effect on the output. Much of the process data is used to negate the effect of these extra input elements. This requires additional dynamical examples and a bigger training set resulting in a larger network structure. All of these extras will slow network learning.

Note that statistical techniques exist to identify elements in the window that are most correlated (Box and Jenkins, 1976; Jenkins and Watt, 1968). These techniques can and should be used to gauge the size of the window for a system where the process order is not easily identified. They can also be used to select the most effective non-contiguous elements in the history. For example, if the first, third and fifth elements back in time are shown to be highly correlated, then they could be used as the input rather than the first through fifth elements. In this study, all elements of the time history are used, because this approach is more general.
Part 1. VIII B 4. Training Set Construction

With the history length set and data organized into vector pair form, the next step is to construct a training set. As the name implies, the training set is a collection of examples from which the network learns the mapping. Once the sample time is beyond the time history length, a vector pair is generated for each sample period. Eventually, the training set represents a large collection of raw data. The entire set cannot practically be included in the training set, because network learning would slow to a standstill.

In addition, inclusion of every vector pair is not necessary, because many vector pairs are redundant. A redundant pair has the same information content as a pair already in the training set. For example, when the system is in steady state, a single vector pair is representative of all the information content of incoming data. If redundant data pairs enter the training set, their presence increases the calculational load of the learning algorithm which results in longer learning times. Since the redundant pairs offer no new information, the extra learning does not improve the quality of the mapping.
Part 1. VIII B 5. Screening

The admission of redundant vector pairs into the training set can be avoided by a screening procedure. There are many ways of accomplishing this, but one way is to reject all pairs whose output is predictable within a certain error tolerance. This means that each vector pair must be rated in terms of error. The error rating selected is the absolute difference between target and reflected outputs. As a reminder, the reflected output is the output that results when a vector pair input is submitted to a network in the input/output mode. Vector pairs with ratings below 0.02 are considered to be redundant and are rejected. Note that the error threshold of 0.02, 2% of the normalized liquid level range, is not a rigid value, but has been shown by experience to be sufficient to delineate known and unknown information. Vector pairs with higher ratings are considered to be new information and are kept. In this manner, most redundant data pairs are identified as they are generated and eliminated from further consideration.

An example of the screening of a single vector pair is shown in Figure 8. The vector pair input is submitted to the network and the reflected output results. The actual output of the vector pair is subtracted from the
Figure 8
Screening Example
Cylindrical Tank System

Vector_3 (input),

\[
\left( \frac{F_0}{F_0 \text{ (MAX)}} \right| \frac{F_{o2}}{F_0 \text{ (MAX)}} \left| \frac{h_{SP1}}{h_T} \right| \frac{h_{SP2}}{h_T} \left| \frac{h_{1}}{h_T} \right| \frac{h_2}{h_T} \right)
\]

Network Model
Reflection Mode

Weights from
Network in
Learning Mode

\[ h_{n3} \]

\[ e_3 \]

\[ |e_3| \]

\[ \text{Is } |e_3| < 0.02 \]

\[ \text{Reject} \]

\[ \text{Keep} \]

\[ (\text{Vector}_3, |e_3|) \]

\( h_{ni} = \text{network output at sample time i} \)
\( h_i = \text{liquid height at sample time i} \)
\( h_{SPi} = \text{liquid set point at sample time i} \)
\( h_T = \text{tank height} \)
\( F_{oi} = \text{outlet flow rate of tank at sample time i} \)
\( F_0 \text{ (MAX)} = \text{maximum outlet flow rate of pump} \)
reflected output and the absolute value taken to form the error rating. If the error rating is less than 0.02, then the vector pair is considered as known information and is rejected. Alternately, if the error rating is above the threshold, the vector pair is kept as interesting information.

The screening process is not complete yet: it must also be tailored to reject blocks of redundant vector pairs. Suppose a block of identical vectors is generated whose information content is yet unlearned. Such data may arise when the system experiences a new steady state for the first time. Only one vector pair is important to improve the quality of the mapping. The rest are redundant and should not be allowed to enter the training set. With the screening method outlined so far, the entire block of redundant data would be allowed to enter the training set and learning would be slowed unnecessarily.

The screening process is further refined by forming a candidate set of vector pairs. This set contains pairs that have passed the initial screening, but are yet to be accepted into the training set. The candidate set is in effect a pool of potentially interesting information. Based on the error rating, a decision is made as to which pair is the next to enter the training set. Once a vector
pair in the candidate set is selected, it is removed and placed in the training set. It is the only vector to enter the training set. After it enters, the network learns until the information content of the new vector is incorporated. Since the learning state of the network has changed after the incorporation of the new vector, the candidate set is re-rated for error. Subsequently, vector pairs below the error threshold are rejected. These include any redundant pairs. Those that remain go through the selection process again. This type of incremental learning allows only vectors with distinctly new information to enter the training set and effectively avoids inclusion of redundant data pairs into the training set.

An aspect of screening that has not been discussed is the manner in which error rating is used to select vector pairs that enter the training set. One simple method is to choose the pair with the highest rating. For a noise free system, this is the pair with the highest information content. For a noisy system, it may simply be the one with the greatest noise content. Since the assumption has been made that the test system is noiseless, then the pair with the highest error will be selected. When the data is noisy, statistical analysis may be employed using the error distribution in order to select the new training
vectors. For example, vectors beyond a standard deviation could be considered to contain too much noise and be rejected from the candidate set. Vector pairs above a lower threshold but within the standard deviation limit would become candidate set members, and the member nearest the average error rating would enter the training set. This is just one possible scheme, and others can be envisioned.
Part 1. VIII B 6. Training and Updating

After a training set is formed, the network can use this set to learn the mapping. Still unspecified are certain parameters in the BP algorithm, specifically the number of hidden neurons and the network learning parameters. The choice of the number of hidden neurons is important to the success of the mapping. These neurons represent the actual memory capacity of the network. No exact method exists for determining the required number, but some general guidelines result from experience. If the number of hidden neurons is too small and the number of training vectors increases beyond a critical point in the learning process, then the mapping representation begins to degrade. The network starts to forget what it has previously learned and tends to incorporate new information more slowly. In effect, the new information interferes with the old, and the network learning is said to saturate. This situation can be corrected by adding more hidden neurons, but such an action constitutes an entirely different mapping configuration for the network and requires complete retraining.

The number of hidden neurons selected for this study was originally five. The mappings that resulted were useful, but mapping degradation was encountered when the
training set grew beyond 80 vector pairs. As a result, the number of hidden neurons was increased to match the number of input neurons, either 15 or 20 depending on whether the inlet flow rate is measured or not. Degradation problems were not encountered with the new structure, but the training time increased substantially. In general, addition of hidden neurons has the effect of increasing training times, therefore it is desirable to not be excessive about their number.

For system identification as a rule of thumb, when the number of hidden neurons at least equals the number of input neurons, then the mapping seems to be successful provided the mapping itself is valid. A more conservative approach when the computing power is available is to double this number.

Besides specification of the number of hidden neurons, two other network learning parameters must be set: the learning rate and the momentum. The learning rate determines the degree of change in the weights per learning cycle or epoch and determines the speed at which the BP algorithm converges to the solution. The momentum determines whether the direction of learning progresses in the same manner as the previous epoch. Both parameters are fractions. The learning rate should be small in order to obtain convergence. In theory, if the network
optimization algorithm is to truly approach gradient
descent in the error, then the learning rate should be
infinitesimally small (Rumelhart and McClelland, 1986).
This is not practical, because it would require infinite
learning time. Consequently, a small fraction is usually
chosen for the rate. In this study, it is set at 0.15,
and convergence is always successful.

The last learning parameter, momentum, is needed to
maintain a significant learning rate when the learning
algorithm is near the solution. At this point in the
learning process, the error is nearly zero. Since error
is directly related to the rate of convergence to the
solution, convergence is practically halted at this point.
The momentum term in the learning algorithm allows a part
of the change in the weights to be in the same magnitude
and direction as that of the previous change. If the
momentum term is large, this allows the learning process
to continue in a steady direction even when the solution
is near. The maximum value of the momentum is one. The
value selected during learning should be close to the
maximum, and in this study it is maintained at 0.9.

One caution concerning momentum is that it should not
be too close to one, because the learning algorithm may
become unstable. For example, if the algorithm overshoots
the actual solution and the momentum is too close to one,
then learning continues to travel away from the solution. It cannot make a corrective action to return. The value selected for this study was not found to suffer from this problem.

Once the network's learning parameters are set and vectors have been incorporated into the training set, learning can begin. The weights are randomized, and learning progresses in epochs. During each epoch, small adjustments in the weights are made in accordance with the learning algorithm. The number of learning epochs must be specified. A large number of epochs will minimize the total error between the targets and actual outputs. Even so, this may not be desirable from either learning time or representational accuracy as will be explained later.

At some sufficient number of learning epochs, the neural network model is satisfactory for prediction. A system that provides for network use and continuous network updating must be instituted. Simultaneous prediction and learning functions can be accommodated by employing two networks: a learning network and a simulation network. The weights of the learning network are transported to the simulation network in the input/output mode which operates quickly to make the required predictions. Meanwhile the learning network can continue processing raw data. A schematic of this
information exchange is shown in Figure 9. In this manner, prediction improves in discrete steps, and use of the model is not hampered by the lengthy learning times.

The learning network continually screens both process data and its candidate set for new interesting vector pairs. The network model tries to keep pace with changes in the process or in the control system. These changes may result from physical changes in the process and readjustments in the control system. In general, physical changes in a process arise from events like heat exchanger fouling, vessel plugging, catalyst degradation. Control system changes result from operator actions such as readjustment of measurement devices or change in controller tuning parameters or alteration of the sampling time. Both process and control changes are usually on the time scale of hours, days, and weeks and learning should be able to keep pace.

As mentioned earlier, system identification in this paper is performed solely on closed-loop data which are the type encountered in day-to-day operations. Open-loop dynamics normally occur only as a result of an identification test. The later is disruptive to operations, but allows quick access to the information needed in identification. Closed-loop dynamics contain less information, because the controller tends to mute the
FIGURE 9
Information Flow

Process

Controller

Controller Parameters

Simulation Net

Neural Net Weights

Inlet Vector Construction and Screening

Updated Screening Criteria

Learning Net

Normalized Vector for Training Set

Measured Variables
relationship between input disturbances and output responses in accordance with its function. Closed loop identification is not appropriate for every type of application, but it is found to be adequate for tuning purposes provided the system undergoes a sufficient range of disturbances. In this situation, relative relationships of the dynamic behavior are required, and the closed-loop data appears to contain the needed information.

The disturbances used in training are meant to be typical of those encountered by a real process. They include steps in set point and inlet flow rate as well as ramps in set point and random inlet flow sequences. Vector pairs generated by these disturbances are screened and included in the candidate set. From the candidate set, data pairs are selected one at a time to form the training set.

During the learning process, a mapping develops gradually as illustrated by the set of response curves shown in Figure 10. The neural network representation in response to a step in set point from 0.75 to 0.4 is compared to those generated from the discrete computer model, which is labeled as the actual response. It should be pointed out that the step disturbance is not a member of the training set. In other words, the responses
FIGURE 10

Neural Network Representation of Step in Set Point

vs.

Number of Vectors in Training Set

- Set Point
- 5 Vectors
- 50 Vectors
- Actual

Sample Time vs. Normalized Liquid Level
generated by the network are all inferred values. The first curve is the response with just 5 training vectors in the training set and the second one with 50 training vectors. Notice that the representation with only 5 training vectors is poor. The jerky nature of the curve is due to gaps in knowledge. Even though the mapping is not complete, the general trend is developing. The case with 50 training vectors is a much better representation and the jerky behavior is not as pronounced. It is expected that continued addition of training vectors should allow the network response to approach more closely that of the actual curve.

The alignment of the actual and network responses is one indicator of the quality of the mapping, but it is not a guarantee of a fully developed mapping. The network is an excellent imitator. If a disturbance is encountered early in the training process that is similar to that used to gauge the quality of the mapping, then the network reproduction of the test disturbance will be close to the actual. In reality, the network is only imitating behavior of a single disturbance in its training set and is still unable to extrapolate beyond its limited knowledge. The ability to extrapolate correctly is critical to success of applications like controller tuning. Disturbances generated by network simulations are
unlikely to be exactly like those contained in the training set, as such the network must have the ability to extrapolate.

For these reasons, the accuracy to which the network approaches the actual response should not be used as an absolute test of the quality of a model. Instead, experience shows that the degree to which the screening process rejects vectors constructed from dynamic data is a better measure. If for example 98% of raw vector pairs from new disturbances are rejected, then one can conclude that the network model is able to adequately simulate the predictive behavior.

Part 1. VIII B 7. Depth of Learning and Tested Training

Another variable that affects the network representation is the degree of learning, specifically the number of learning epochs. As this number increases, the reflected outputs of the training set become closer to their respective target values. For this reason, the sum of the squared error between output and target for the training set, the Total Squared Error (TSE), is one measure of the degree of convergence of the mapping. Typically, the TSE is checked after each learning epoch and if it is below a prescribed value then learning stops,
otherwise it continues. TSE behaves in a predictable manner as illustrated in Figure 11. When a new vector pair is added to the training set and learning commences, its value begins high and decreases monotonically to an asymptote after many learning cycles. When another, vector pair is added, the TSE jumps to a new initial value and then decays to another asymptotic value as learning progresses. In general, as the number of training set members increases, the initial value of the TSE becomes lower and the asymptotic value becomes higher.

The last point indicates that problems may arise if TSE is used as a benchmark to stop learning, as is often recommended in the literature. The recommended use of TSE may be appropriate for a situation where the number of members in the training set is fixed and may not be applicable to one where the number changes continually like the one at hand. Suppose a TSE benchmark is set so that learning stops when the calculated TSE is below a critical value. Since the TSE asymptote increases as the training set becomes larger, it can rise above the critical value. At this point, no amount of training will reduce TSE below the benchmark, and learning does not automatically stop. As a result, this method becomes wholly unreliable as a means of truncating the learning process.
FIGURE 11
Idealized Training Set Error (TSE) Behavior
vs.
Number of Vectors in Training Set

TSE (Arbitrary Scale)

--- n vectors
--- n+1 vectors
--- n+2 vectors

Training Epochs
In order to correct the situation, another method is used. It is called tested training (Pollard, et. al., 1990) and is similar to a technique in the statistical literature called cross-validation (Wold, 1978). This procedure consists of training the network for a short number of learning cycles then performing a test to check if learning is adequate. The test consists of measuring the error between target and network output for a test set of vector pairs. This error will be called the test set total error (TSTE). The test set is composed of members selected at random from the candidate set. The point at which the TSTE achieves a minimum has been found by experience to be the state where the mapping best resembles the ultimate mapping. Figure 12 shows an example of TSTE behavior. Note that it usually rises initially due to misdirection of the BP learning algorithm but eventually falls when learning is in the correct direction. It goes through a minimum and will slowly rise if learning continues.

Tested training is instituted when a new training vector is added to the training set. The process of tested training is shown in the logic diagram in Figure 13. The network is then trained for a small number of epoches, in this case 50. Once this is complete, a new TSTE is calculated. When the absolute value of the
FIGURE 12
Test Set Total Error (TSTE) vs. Number of Testing Cycles
50 Epochs of Learning per Cycle

TSTE

0.75 0.80 0.85 0.90 0.95 1.00 1.05 1.10

Testing Cycles

0 5 10 15 20 25
FIGURE 13
Network Training
Logic Diagram

START

$\text{i = 0}$

Training Set → Train Network for One Epoch

Is Number of Training Epochs 50?

YES

$i = 1, \text{TST}_E = 0$

NO

Reset Number of Training Epochs to 0

Test Set Vector$_i$ (input) → Reflect Network

Test Set Vector$_i$ (output) → $e_i$

$|e_i|$

$\text{TST}_E = \text{TST}_E(i-1) + |e_i|$

Last Test Set Vector?

NO → Increment $i$

YES → Increment $j$

$\text{TST}_E = \text{TST}_E_i$

Is $|\text{TST}_E(i)-\text{TST}_E(i-1)| < 0.001$?

NO

Stop Training
normalized slope of the TSTE is above a certain threshold, the learning cycle continues until it falls below the threshold. For this study the threshold is set at 0.001. Once again, this threshold is not an absolute, but has been found by experience to be accurate enough to adequately identify the point where TSTE is minimized.

Not only is this method fairly reliable at stopping learning, but it has the added benefit of preventing overlearning. Any attempt to reach the asymptote of the TSE by an inordinate number of learning cycles may actually degrade the representation. When the number of training vector pairs are few, interpolation between the pairs is highly sensitive to the inherent properties of the network. The network interpolates like a high order polynomial curve fitting program, and may lead to unpredictable results. This problem is demonstrated in Figure 14. The actual response to a step change is given along with two network responses. The first network response occurs when the network is exposed to a large number of learning cycles and the TSE value is small. The second network response is when the learning process has been stopped at an earlier point using the TSTE test. For this case, even though the TSE value for the training set is relatively large, the representation is much better.
FIGURE 14
Effect of Tested Training on Response Representation
Step Down in Set Point

- Set Point
- Actual
- Deep Learning
- Tested Training

Normalized Liquid Level

Sample Time
One theory of why the TSTE is useful pertains to limiting the pseudo-order of the network's curve fitting capacity. Imagine the network behaves like a high order polynomial curve fitting program. When it learns a new mapping, its curve fitting order always begins at one and progresses to higher order until every mappable training set member is accurately represented. Suppose the order of curve that is being fitted is really one, and the data is either noisy or inconsistent. Since, the network has no such apriori knowledge, it will incorrectly fit this data with a higher order curve. This is a direct result of the excess degrees of freedom and over-parameterization in the network structure.

The testing method is one way of limiting this pseudo-order. If the test set is chosen at random, then its information content should represent a sample of the ultimate mapping. This set can also be thought of as a preview of vector pairs that may be accepted into the training set in the future. Since the ultimate mapping must reflect the commonalities of both the test and training set, then as learning progresses both the TSE of the training set and TSTE of the test set should decrease. If the TSTE increases, learning is tending away from the test set information and away from the proper mapping. Further learning is unproductive and should be stopped.
For these reasons, the TSTE parameter is a way of tailoring the pseudo-order of the curve fitting process and thereby controlling the manner of interpolation between training set members.

Part 1. VIII B 8. Training Vector Removal

Up to this point, discussion has been focused on the manner of incorporating vector pairs into the training set. However, a training pair must also be removed from the training set for two reasons: first, the set becomes full and second, the dynamics of the process change. The training set can become full if a limit is placed on the number of members of the set. This limit is based on learning time considerations. If the upper limit is reached, then one of the data vectors must be extracted for another is to enter. The second reason for removing vector pairs is changing process dynamics. Pairs that represent the old model must be replaced so that a new model can develop.

One simple removal method is to extract a data pair at random from the training set. If the number of training vectors is small, then each vector can represent a significant block of information, and its removal and replacement can adversely affect the mapping. On the
other hand, when the number of data pairs in the training set is large, a single pair will be of small consequence to the total mapping. Exchange of one small bit of information for another more current piece is one way of smoothly adapting the BP mapping.

Age of a vector can be another criterion for removal based on the assumption that newer data is more representative of current process dynamics than older data. However, when a process is not changing rapidly or when certain dynamical situations occur rarely, then old data pairs may have value equivalent to that new data. In this case, age may not be an appropriate criterion.
Part 1. IX. Simulation With the Network Model

Part 1. IX. A. Introduction

Once the mapping is available, it can be used to simulate system responses to disturbances. The results from simulations can be applied to controller tuning or to redesign of the control algorithm. In this section, the process of network simulation will be discussed. The accuracy of network prediction will also be evaluated by comparing with those of the discretized differential equation model, referred to later as the actual model.

Two types of mappings will be discussed for the model system. In the first type, the inlet flow rate measured so that the mapping has four input variables: inlet and outlet flow, set point and liquid level. This type is more typical of feed forward applications. In the other type of mapping, the inlet flow rate is unmeasured so that the mapping has three input variables. This type is typically used in feedback applications. In industrial practice, feedforward control is less prevalent than feedback control, therefore the first type of mapping is of less practical importance than the second.

Network modeling can be of two classes, one-step-ahead prediction and recursive closed-loop response. One-
step-ahead modeling employs past and present data from the actual system as input to the network to predict the system behavior one step into the future. Recursive modeling employs a one-step-ahead prediction, but this prediction is used to construct the input data for the next step. In this manner, the network model is used to walk the prediction forward in time. In the one-step-ahead model, modeling error is not included as part of the input to the network, whereas in the recursive model it is.

Consider, a one-step-ahead model of the test system where the inlet flow rate is measured. In this situation, the input vector is derived from the actual system at each sample time. No network model prediction is included as part of the input vector. A comparison of the response curves between the one-step-ahead network mapping and those of the actual model is shown in Figure 15. In this case, the accuracy of the network mapping is excellent. The network's one-step-ahead prediction is almost indistinguishable from that of the actual.

Even though an accurate one-step-ahead model is important to recursive modeling, the former by itself is not as useful as the later. A knowledge of the system behavior only one step into the future may not be sufficient to make a decision on changes in the controller
Figure 15
One Step Ahead Behavior
Comparison of Actual and Network Responses
Step Up in Set Point

Normalized Liquid Level

Set Point
Actual
Network

Sample Time
behavior that will lead to significant improvements in performance. Instead, a view of many steps into the future is more informative. For this reason, recursive modeling is the simulation technique emphasized in this study.

The process of recursive modeling is shown in Figure 16 for the case where the inlet flow rate to the tank is not measured. It begins by forming a vector from the initial conditions of the input variables. This input vector is submitted to the network, and the liquid height in the next time step is predicted. This value is used together with the desired set point for the next time step to calculate the error between set point and liquid height. The error is in turn used in the controller algorithm, the velocity form of the PI equation, to compute the control action, the outlet flow rate, in the next time step. All this information incorporated into the next network input vector and the height one step ahead is predicted again. In this manner, the closed-loop response of the controlled variable moves forward in time.

As mentioned earlier, any modeling errors in predicting the liquid level are included in the calculation of the controller action. This magnifies the error and can significantly change the controller variable behavior. Since, controller behavior is part of the input
Recursive Simulation Using BP Network

Set Point Specification

\( (h_{sp4}, h_{sp3}) \)

Initial Conditions

\( (F_{o1}, F_{o2}, h_{sp1}, h_{sp2}, h_1, h_2) \)

\( h_i = \text{liquid height} \)

\( h_{sp_i} = \text{set point} \)

\( F_{o_i} = \text{outlet flow rate} \)

\( i = 1 \text{ to } 5 \)
vector to the network, modeling error can alter the manner in which the network responds as the simulation proceeds into the future.

Recursive modeling is applied to the control of liquid level system previously described in VII. There are five examples. The inlet flow rate is measured in the first two, but not in the remaining three. The first example is a linear system where the cross sectional area of the tank is constant with height. The second example is a nonlinear system where the cross sectional area of the tank decreases with increasing height. In the third and fourth examples where the inlet flow rate is not measured, the systems are linear and nonlinear as before. The final example is a first order level control system where an uncontrolled outlet is added to the cylindrical tank. The flow rate from this outlet is proportional to the liquid level height.

Part 1. IX. B. Simulation Results and Discussions

Part 1. IX. B. 1. Linear System with Inlet Measured

In the first case, the inlet flow rate is measured. The network is trained with a series of 17 different step changes in set point and 7 different step changes in inlet
flow rate that are listed in Table 3. The disturbances in this table will be used in all future simulations to train the network. From the disturbances, approximately 1,020 candidate vectors are generated. The screening process identifies 50 training vectors as necessary for a complete mapping.

The quality of the mapping is tested with a step down in the set point from 0.75 to 0.4. A comparison of the responses to this disturbance for the network model and the actual is shown in Figure 17. The upper panel contains the liquid level responses, the controlled variables, and the lower panel contains the outlet flow rate responses, the manipulated variables. Notice that the prediction by the network is excellent: the two liquid level responses are very close. The responses of outlet flow rates are also in good agreement. Another test is performed with a step in inlet flow rate and the results are shown in Figure 18. Again the network's representation is very similar to that of the actual model.

The above studies indicate that an accurate model can be constructed using a BP network for a simple integrating process where the inlet flow rate is measured. The model is obtained by training with examples formulated from closed-loop disturbances, which are similar to the type
<table>
<thead>
<tr>
<th>Disturbance</th>
<th>From</th>
<th>To</th>
<th>Inlet Flow Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step Down in Set Point</td>
<td>0.87</td>
<td>0.50</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>0.47</td>
<td>0.13</td>
<td>0.70</td>
</tr>
<tr>
<td></td>
<td>0.87</td>
<td>0.23</td>
<td>0.70</td>
</tr>
<tr>
<td></td>
<td>0.80</td>
<td>0.44</td>
<td>0.66</td>
</tr>
<tr>
<td></td>
<td>0.52</td>
<td>0.15</td>
<td>0.20</td>
</tr>
<tr>
<td></td>
<td>0.80</td>
<td>0.53</td>
<td>0.30</td>
</tr>
<tr>
<td></td>
<td>0.27</td>
<td>0.13</td>
<td>0.70</td>
</tr>
<tr>
<td></td>
<td>0.57</td>
<td>0.33</td>
<td>0.30</td>
</tr>
<tr>
<td>Step Up in Set Point</td>
<td>0.13</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td></td>
<td>0.40</td>
<td>0.80</td>
<td>0.30</td>
</tr>
<tr>
<td></td>
<td>0.27</td>
<td>0.68</td>
<td>0.45</td>
</tr>
<tr>
<td></td>
<td>0.53</td>
<td>0.80</td>
<td>0.77</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td>0.50</td>
<td>0.20</td>
</tr>
<tr>
<td></td>
<td>0.73</td>
<td>0.30</td>
<td>0.30</td>
</tr>
<tr>
<td></td>
<td>0.33</td>
<td>0.53</td>
<td>0.75</td>
</tr>
<tr>
<td></td>
<td>0.33</td>
<td>0.67</td>
<td>0.80</td>
</tr>
<tr>
<td></td>
<td>0.13</td>
<td>0.63</td>
<td>0.80</td>
</tr>
<tr>
<td>Step Down in Inlet Flow Rate</td>
<td>0.35</td>
<td>0.15</td>
<td>0.47</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.30</td>
<td>0.20</td>
</tr>
<tr>
<td></td>
<td>0.80</td>
<td>0.55</td>
<td>0.40</td>
</tr>
<tr>
<td></td>
<td>0.60</td>
<td>0.30</td>
<td>0.44</td>
</tr>
<tr>
<td>Step Up in Inlet Flow Rate</td>
<td>0.20</td>
<td>0.60</td>
<td>0.27</td>
</tr>
<tr>
<td></td>
<td>0.50</td>
<td>0.75</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>0.75</td>
<td>0.90</td>
<td>0.45</td>
</tr>
</tbody>
</table>
FIGURE 17
Response of Linear System to Step Down in Set Point
Inlet Flow Rate Measured

Normalized Liquid Level
- Set Point
- Actual
- Network

Normalized Flow Rate
- Inlet
- Outlet Actual
- Outlet Network

Sample Time
0  5  10  15  20  25  30  35
FIGURE 18
Response of Linear System to Step Down in Inlet Flow Rate
Inlet Flow Rate Measured

Set Point
Actual
Network

Normalized Liquid Level

Normalized Flow Rate

Inlet Flow Rate
Actual Outlet
Network Outlet

Sample Time
encountered in industrial practice. More significantly, the resulting mapping yields accurate simulated responses to set point and load disturbances that were not part of the training process.

Part 1. IX. B. 2.

Nonlinear System with Measured Inlet Flow Rate

The mapping capability of the network is explored further with a nonlinear system. The system is created by replacing the cylindrical tank with a truncated cone placed so that the horizontal cross sectional area decreases with increasing height. The normalized radius the top of the tank in is 0.4 and that of the bottom is 0.8. By comparison, the radius of the linear tank was 0.5. The slope of the tank wall is approximately 15 degrees from vertical. As a result of these changes, this system is no longer a simple integrator.

Comparisons of responses to steps in set point for the linear and nonlinear systems are given in Figures 19 and 20. In the first figure, the step change is in the upper portion of the tank where the cross sectional area is smaller and the response is quicker. In the other figure, the step change is in the lower portion of the tank where the cross sectional area is large and the
FIGURE 19
Comparison of Responses of Linear & Non-linear Systems
Step Down in Set Point in Upper Portion of Tank
Inlet Flow Rate Measured

Normalized Liquid Level

Set Point
Linear
Non-linear

Normalized Flow Rate

Inlet
Linear
Non-linear

Sample Time
FIGURE 20
Comparison of Responses of Linear & Non-linear Systems
Step Down in Set Point in Lower Portion of Tank
Inlet Flow Rate Measured

- Set Point
- Linear
- Non-linear

Normalized Liquid Level

- Inlet
- Linear
- Non-linear

Normalized Flow Rate

Sample Time
response is more sluggish. Note that the differences in responses in both cases is discernable.

The nonlinear system is subjected to the same disturbances that were used in training the network for the linear system. Data from these disturbances are again used to form the examples needed to train the network. Screening and learning processes are repeated. The model representation is complete after selection of 60 training vectors. The model's response to the same step in set point as in the linear case is shown in Figure 21. The response to the same step in inlet is shown in Figure 22. It can be seen that the network's responses do not match those of the actual nonlinear model exactly, but they are reasonably close.

Since the dynamic behavior is different in the upper and lower portions of the tank, additional response tests are needed to encompass both regions. The results are shown in Figures 23 and 24. Note that the network responses are very close to the actual responses.

From these tests, the conclusion is made that the representation of the network appears to be accurate. More specifically, the network not only can use closed-loop data to develop a model of a linear system, but it adequately represents a nonlinear system. Again, this is important since most real systems are nonlinear.
FIGURE 21
Response of Non-linear System
Step Down in Set Point in Middle of Tank
Inlet Flow Rate Measured

Normalized Liquid Level

Set Point
Actual
Network

Normalized Flow Rate

Inlet
Actual Outlet
Network Outlet

Sample Time
FIGURE 22
Response of Non-linear System
Step Down in Inlet Flow Rate in Lower Portion of Tank
Inlet Flow Rate Measured

Normalized Liquid Level

- Set Point
- Actual
- Network

Normalized Flow Rate

- Inlet
- Outlet Actual
- Outlet Network

Sample Time
FIGURE 23
Response of Non-linear System
Step Down in Set Point in Upper Portion of Tank
Inlet Flow Rate Measured

Normalized Liquid Level

- Set Point
- Actual
- Network

Normalized Flow Rate

- Inlet
- Outlet Actual
- Outlet Network

Sample Time
FIGURE 24
Response of Non-Linear System
Step Down in Set Point in Lower Portion of Tank

Inlet Flow Rate Measured

Normalized Liquid Level

□ Set Point
— Actual
— Network

Normalized Flow Rate

○ Inlet
— Outlet Actual
— Outlet Network

Sample Time
Part 1. IX. B. 3.
Linear and Nonlinear Systems – Unmeasured Inlet Flow Rate

Up to this point, the mappings have included the inlet flow rate as an input variable. In the next two examples, the inlet flow rate is unmeasured. As a result, any change in the inlet flow rate is now considered as an unmeasured disturbance. The effects of this disturbance are inferred by the behavior of the measured variables. For example, a step up in inlet flow rate will be detected by a jump in liquid level measurement in subsequent sample periods and by corresponding changes in the behavior of the outlet flow rate.

Because the effect of the unmeasured disturbance is delayed by a sampling period, inconsistent vector pairs are generated. Consider a step in inlet flow rate where the system is initially at steady state. When the step occurs, the input portion of the vector pair exactly matches that of the steady state. The output portion of the vector pair is the liquid level one step ahead. For the steady state case, the output is the same as the period before, but for the unmeasured disturbance the output reflects the effects of the disturbance. These two vectors have the same inputs but different outputs and are therefore inconsistent. The vector pair for the
disturbance is noise and will damage the mapping in terms of performance near steady state conditions. If enough opposing disturbance vectors are included in the training set, their effect will tend to cancel. Vectors much beyond the onset of the unmeasured disturbance are distinct and contribute more useful information to the mapping.

In conventional open-loop identification, unmeasured disturbances are considered as noise and are avoided during an identification test by keeping disturbing elements as constant as possible. In closed-loop identification, unmeasured disturbances are not as easily avoided, however in the identification process, they can be selectively rejected. The rejection process involves ignoring all disturbances except those that start from steady state and involve changes in set point. These changes always produce unique vector pairs, and introduce no inconsistency into the training set.

This type of identification can be accomplished in practice by instituting steady state and set point change detectors. The steady state detector would determine when the system is within a certain tolerance of steady state. If a change in set point occurs above certain threshold to disturb the steady state condition, the data just preceding and following can be used in identification.
Note that this strategy limits the type and number of disturbances that are available for identification. The detection system above cannot practically avoid all unmeasured disturbances, but it can hold them to a minimum.

The identification scheme is tested with all disturbances used in training restricted to those with constant inlet flow rate. The inlet flow rate is assumed to be unmeasured, so it is not a variable used in the mapping. The resulting network model for a linear system produces a response to a set point change as shown in Figure 25. Note that the network response of the liquid level is close to that of the actual response. Figure 26 shows the response curve when the unmeasured disturbances are included in the identification. The accuracy of the response has decreased due to the inconsistencies introduced by the unmeasured disturbances. It will be shown later that this degradation is not harmful to the utility of the model. For this reason, in the remaining sections, unmeasured disturbances are included in the identification process.

The accuracy of the network model in reproducing responses for the nonlinear system is examined in Figure 27 and Figure 28. The results are similar to those of the linear system. The response to set point change is
FIGURE 25
Response of Linear System
Step Down in Set Point in Middle of Tank
Unmeasured Disturbances Not Included in Model Identification
Inlet Flow Rate Unmeasured

Set Point
Actual
Network

Normalized Liquid Level

Normalized Flow Rate

Inlet
Outlet Actual
Outlet Network

Sample Time
FIGURE 26
Response of Linear System
Unmeasured Disturbances Included in Model Identification
Step Down in Set Point in Middle of Tank
Inlet Flow Rate Unmeasured

- Set Point
- Actual
- Network

Normalized Liquid Level

Normalized Flow Rate

Sample Time
FIGURE 27
Response of Nonlinear System
Step Down in Set Point in Middle of Tank
Inlet Flow Rate Not Measured

Normalized Liquid Level

- Set Point
- Actual
- Network

Normalized Flow Rate

- Inlet
- Outlet Actual
- Outlet Network

Sample Time
FIGURE 28
Response of Non-linear System
Step Up in Set Point in Lower Tank
Inlet Flow Rate Not Measured

Normalized Liquid Level

- Set Point
- Actual
- Network

Normalized Flow Rate

- Inlet
- Outlet Actual
- Outlet Network

Sample Time
fairly accurate except that rippling effects develop when the steady state is approached.

Part 1. IX. B. 4. First Order System

The final example involves a linear first order level control on a cylindrical tank with two outlets. One outlet is controlled and the other is not. The flow rate from the uncontrolled outlet is considered to be proportional to the tank height. The inlet flow rate is unmeasured. This system is no longer an integrator and is first order and stable. The modeling parameters are the same as those listed for the linear system. One added parameter is the proportionality constant for the uncontrolled outlet, \( k \), which is calculated by assuming that the maximum flow rate at tank height, \( h_t \) (1.5 meters), is \( 1/4 \) of the maximum flow at the pumped outlet, \( F^+ \) (0.225 cubic meters/second).

\[
1/4 \cdot F^+ = k \cdot h_t
\]

Under this assumption, the \( k \) is 0.0375 square meters per second.
Part 1. IX. B. 4. a. Open-Loop Mapping

Up to this point, all mappings have been closed-loop. The statement was made previously that open-loop mappings allow better representation of the system model than are closed-loop models. This first-order example presents the opportunity of demonstrating that with a proper open-loop identification procedure the system model can be approached quite accurately. The identification procedure involves devising a pseudo-random binary disturbance of the controlled outlet flow rate in such a manner that maintains the liquid level within the limits of the tank.

Such a signal was devised, and the response data was used to train the network. An open-loop test is performed where the system is initially at steady state, and the controlled outlet flow rate is reduced slightly. The inlet flow rate is held constant. The response of the system to these conditions is shown in Figure 29 for both the actual and network simulations. The level in the tank rises and reaches an asymptotic value in both cases. The network simulated response is similar to the actual response. The initial slope of the response determines the first order time constant. The slope for the two responses is nearly identical. The ultimate value of the response determines the system gain. The ultimate value
FIGURE 29
Open-loop Response to Step in Controlled Outlet
Cylindrical Tank with Controlled and Uncontrolled Outlets
Inlet Flow Rate Unmeasured

Normalized Liquid Level

Set Point
Actual
Network

Normalized Flow Rate

Inlet
Controlled Outlet

Sample Time
for the actual system is only 6% higher than that for the network response. Other open-loop tests were performed and the accuracy of the results was similar to the previous case.

The results of this test can be generalized in that a model for the system can generally be obtained by properly designed open-loop identification. This type of identification is the most direct route to obtaining a system model. This study is the opposite end of the spectrum in that it limits the identification to the closed-loop and limits the disturbances to those that are typical in daily operations. This is a more difficult means of identification, and the model that results will probably contain modeling error. The key question is does the modelling error prevent the identification from yielding a useful model when the closed-loop disturbances are relatively rich in dynamic behavior.

One note on the open-loop identification concerns the inclusion of the set point as an input variable to the network mapping. In the closed-loop mapping, the set point is a relevant variable to the mapping, but in the open-loop it is not. In fact, much of the dynamic data is used in negating the effect of set point. A more efficient mapping strategy for the open-loop would have been to omit set point as an input variable.
Part 1. IX. B. 4. b. 1st Order Closed-Loop Mapping

A closed-loop network model was constructed as in the previous examples and tested with a disturbance in set point from 0.75 to 0.4. The inlet flow is held constant at 0.5. The response curves are presented in Figure 30. The one-step-ahead prediction is almost exact. The recursive prediction, labeled as the network prediction, is not as accurate as expected.
FIGURE 30
Response of 1st Order Linear System
Cylindrical Tank with Controlled and Uncontrolled Outlets
Inlet Flow Rate Unmeasured

Normalized Liquid Level

Set Point
Actual
One-Step Ahead
Network

Normalized Flow Rate

Inlet
Actual and One-Step Ahead Outlet
Network Outlet

Sample Time
Part 1. IX. C. Discussions

The simulations presented in the previous sections show that the recursive modeling technique appears to adequately reproduce system responses to standard closed-loop disturbances. This conclusion is valid regardless of whether the inlet flow rate was measured or not. It is valid for the linear and nonlinear integrating systems and for a first order system.

As a final note, the network representation of a closed-loop response curve will never exactly match that of the actual response. The problem is the tiny errors made in prediction at each step. The network may be excellent in predicting the one-step-ahead response, but occasionally will make an error. This error is incorporated into the vector pair used to predict the next time step, where it appears as a perturbation. As a result, the error is magnified by the controller, and is incorporated into the future network input. For this reason, a slight error in the early stages of the prediction process has a cumulative effect, so that the later portion of the simulation may have little resemblance to the actual response. Yet, the network is behaving in a relative sense exactly as the actual model would behave if subjected to the same errors. It is this
relative behavior that probably is important in model applications like controller tuning.
PART 2. MODEL APPLICATION

PART 2. I. Introduction

The efforts in the first part of this study involved the development of techniques for obtaining the process model and the evaluation of the accuracy of the resulting model to reproduce closed-loop responses. In this part of the study, a tuning application is examined in detail and a method of optimal control is outlined. The BP software employed is from McClelland and Rumelhart (1988). For the tuning application, the accuracy of the network model will be assessed in terms of its ability to predict tuning parameters close to those of the actual system.

The network model will be used to determine the system responses to disturbances for the purpose of tuning a PI controller. This will be accomplished by selecting a typical disturbance or, in the case of a nonlinear system, a set of disturbances whose response curve(s) will be modelled. A search will ensue for tuning parameters that yield a reduced integral absolute error between the controlled variable and the set point. In effect, new tuning parameters will be proposed and the network model will be used to determine when to accept or reject the recommendation.
Part 2. II. Model-Based Tuning

Part 2. II. A. Background

Many industrial processes and most chemical reactors are controlled by the PI algorithm (Schnelle and Richards, 1986). The velocity form of the algorithm was shown previously in Table 2. The algorithm is simple, well behaved and easily understood. In addition, a well tuned PI controller can perform at nearly the same level as a model based controller, however rarely are PI controllers tuned to this degree. The algorithm has two adjustable parameters to tune, namely $k_c$ and $tau_i$. If a model of the system exists, then tuning settings can be suggested, and the model is used to gauge the improvement. A closed-loop neural network mapping can serve as a model for tuning purposes. The key issue is whether it is accurate and sensitive enough to properly indicate the direction of tuning that leads to improved performance.

Before entering the subject of tuning using a neural network model, the subject of conventional tuning methods will be discussed. Several methods exist for tuning the PID algorithm. Most methods are meant to supply a first guess of these parameters. For fine tuning, trial-and-error in the field is the principal method. Some of the
gross tuning methods are Process Reaction Curve Method (Ziegler-Nichols, 1942 and Cohen and Coon, 1953), IMC method (Rivera, 1986) and Ziegler-Nichols continuous cycling method (Ziegler and Nichols, 1942; Perry and Green, 1984). The first method requires that a response to a step disturbance be analyzed based on an assumption that the system can be modelled as a first order with delay. Three parameters are determined from the response and are used in tuning rules which are designed to produce a closed loop response with a quarter decay ratio. This method requires that the parameters be fitted using data from an open-loop disturbance test. The IMC method requires that a model of the system be assumed. Usually, a first order system is assumed and then Cohen and Coon techniques are used to define the first order system parameters. The Ziegler-Nichols continuous cycling method is the best known approach and requires measurement of two parameters by a structured trial and error procedure. The method is empirically developed to provide a quarter decay ratio.

Neural network tuning will not replace first guess methods, because the controller must have some initial tuning in order for the system to generate the closed loop disturbance data necessary in training the network. This means that the first guess parameters must be implemented
before network identification can begin. Inaccuracy of gross tuning methods and the trial-and-error fine-tuning method enhance the dynamical behavior of the system. Recall that perfect tuning decreases the dynamical content of the closed-loop data. Consequently, inaccurate tuning is actually beneficial to closed-loop identification.

Several closed-loop tuning methods exit, but all depend on external perturbation of the system for proper identification. Harris and Mellichamp propose a frequency domain identification technique where the system is disturbed by a specially designed pulse in order to test all frequencies of interest (Harris and Mellichamp, 1980). Yuwana and Seborg propose a gross tuning method that uses a first order plus dead time model with a Padé approximation (Yuwana and Seborg, 1982). Fitting of parameters is accomplished by perturbing the process with a test disturbance, usually a step change in set point. Finally, Astrom and Hagglund describe an automatic tuning method that involves a relay controller which causes the system to oscillate with a small amplitude (Astrom and Hagglund, 1988). It also involves a single closed loop experiment to identify the model.

Application of neural network technology to tuning is not widely discussed in the literature. One application concerns the use of back-propagation to aid in Ziegler and
Nichols tuning (Swinarski, 1990). In this tuning method, the process is assumed to be first order and certain auxiliary parameters must be measured in order to ultimately determine the tuning parameters. The neural network mapping is made using open loop data, and is constructed to form a sophisticated look-up table for the ancillary parameters.

This type of network application has its disadvantages. First, Ziegler and Nichols tuning is not very accurate for linear systems and is worse when applied to nonlinear ones. In addition, open-loop data in real processes are not commonly available for tuning purposes. Open loop testing for the purpose of fine tuning is disruptive to the process in comparison to the simple trial-and-error approach.

This study employs a different approach called network model reference tuning. The information flow of the tuning process is shown in Figure 31. A network in the learning mode structures a one-step ahead model from closed-loop information from the process. Periodically, the weights from this model are sent to another network in the reflection mode. This second network is involved in the network model reference tuning procedure. This procedure is listed in detail in Figure 32. Some of the aspects of this procedure will be explained in later
FIGURE 31

Information Flow
Network Model Reference Tuning

- \( k_c \) (optimum)
- \( \tau u \) (optimum)

- \( h_{sp} \)
- \( h_m \)

Controller

System

- \( S \)

- \( M \)

Learning Network
One-Step-Ahead Model

- \( F_i \)
- \( F_{im} \)
- \( F_{om} \)
- \( F_0 \)

- \( h \)

---

- \( M \) actual to measurement conversion
- \( S \) signal to actual conversion

- \( h \) = liquid level
- \( h_m \) = measured liquid level
- \( h_{sp} \) = liquid level set point
- \( F_i \) = inlet flow rate
- \( F_{im} \) = measured inlet flow rate
- \( F_0 \) = outlet flow rate
- \( F_{om} \) = measured outlet flow rate
FIGURE 32
Network Model Reference Tuning Procedure

Start

Select Disturbance

Initialize Tuning Parameters

Change Tuning Parameters \( k_c, \tau_i \)

Get Disturbance Data

Initial Conditions & Set Point Vector

Recursive Network Simulation

IAE & CVSE

\[ TE = \text{IAE} \times WF + (1 - WF) \times CVSE \]

WF

TE

Are all points on tuning grid calculated?

Yes

Have all disturbances been considered?

No

No

Yes

Sum TE From All Disturbances

Form Surface Plot

Select Optimum Tuning Parameters

\( k_c \) (optimum), \( \tau_i \) (optimum)

WF – Weighting Factor
IAE – Integral Absolute Error
CVSE – Controller Variable Squared Error
discussions. The basic idea is that the network is used to simulate a list of test disturbances on which the integral absolute error is calculated and minimized. These calculations are repeated for a grid of tuning parameters. The goal of the tuning process is to determine the tuning parameters within the grid that allow the controller to reject disturbances similar to the test disturbances. In this manner, the search for better tuning parameters can be done off-line and thus is not disruptive to the process. However, such an approach depends both on the accuracy of the model and on the type of test disturbances selected in order for the predicted tuning parameters to be close to the actual optimal parameters. No external disturbance test is designed for identification purposes, rather the model develops as the system is disturbed in normal operations. No functional form of the system model is needed apriori, because this is addressed internally by the network. The tuning methodology is tested using PI control, but can be applied to tuning any controller whose action can be determined analytically.
Part 2. II. B. Simulated Systems

In order to begin to explore this approach, network models are constructed from data generated by the linear and nonlinear level control systems specified earlier in Table 2. The initial settings of \( k_C \) and \( \tau_u \) were determined from quarter decay considerations to be 1.5 and 2.0 respectively. The disturbances used in training the network are those presented earlier in Table 3. Once the training is complete, these models are used to simulate the responses of test disturbances in set point. For these test disturbances, the integral absolute error (IAE) between the network response and the set point is computed. The objective is to find tuning parameters which minimize the IAE.

Other error parameters besides IAE can be used in the minimization, in particular Integral Squared Error (ISE) and Integral Time-weighed Absolute Error (ITAE) are common in control literature. ISE emphasizes moves far from the set point and ITAE emphasizes errors that persist in time. Of the three error criteria, ITAE yields the most conservative settings in terms of overshoot, decay and robustness while ISE gives the least conservative (Seborg, et. al., 1989). ITAE was not selected for this study because the network simulations of disturbances may result in oscillations about the set point due to modeling error.
ITAE will overemphasize these oscillations and skew the determination of the minimum.

The search for minimum IAE can accomplished by a number of existing search algorithms such as the simplex or the bisection method. These types of search methods entail proposing a change in $k_c$ and $\tau_{ai}$ and using the network model to determine the effect of the change on IAE. If IAE is lowered by the change, then the tuning parameters are kept and if not they are rejected and new ones proposed. These methods have the advantage that the search criteria can become quite sophisticated and a minimum found comparatively quickly. They may have the capacity to overcome the local minima that often occur in tuning applications and predict the correct global optimum. However, they do not give a clear picture of the IAE surface. It will be demonstrated later that a global picture of the IAE surface is extremely useful in some cases because, the preferred operating region may be at a suboptimal setting.

The grid search method provides a global picture of the IAE surface. It involves changing both $k_c$ and $\tau_{ai}$ in small steps so that a range of $k_c$’s and a range of $\tau_{ai}$’s are covered. At each step, the IAE is computed until the entire range or grid of interest is covered. The global minimum for the grid is determined by comparing all local
minima and selecting the lowest one. The major disadvantage of this method is that many more calculations are involved and much more computer time is needed. This flaw is not fatal provided the grid is not too fine or a more efficient grid size method is used. An off-line search of this type takes approximately one day of computation on a personal computer. This is not unreasonable since tuning may be required on the time scale of weeks or months, and a day of off-line calculation is within this time frame.

A disturbance must be chosen as a test case which properly represents the type of disturbances encountered in daily operations. Many disturbances fit this requirement, but simple ones are either an impulse displacement of the controlled variable or a step change in set point or in load. The choice of load or set point disturbance both depend on whether the system operator desires improved disturbance rejection for one or the other or both. A step in set point is selected for this study, in particular, a step up in set point in the normalized range of 0.5 to 0.75. The inlet flow rate is held constant at 0.5.

For level control in a linear tank, a single disturbance suffices. Tuning parameters which minimize IAE for set point disturbances theoretically result in a
reduction of IAE for load disturbances. In this case, the load and process transfer functions are the same. The load and process transfer functions may not be the same for more complicated systems, and both load and set point changes can be included as test disturbances in the tuning these systems.

The mapping where the inlet flow rate is not measured is chosen as the sample system. The one where it is measured could have been selected, however this model is more typical of feed-forward/feed-back control applications. In these applications, the tuning of the feed-back portion is of lesser importance than for strict feed-back control. In addition, the mapping with the inlet flow rate measured was shown to be more accurate than the one with no measurement. For this reason, if the tuning approach performs adequately for the case with no inlet flow measurement, then it probably will function as well for the case where it is measured.

Part 2 II B. 1. Linear Simulated System

Two different identification techniques are applied, one with only set point disturbances and the other with both set point and inlet flow rate disturbances. The first identification yields very accurate response curves,
because there is no unmeasured disturbance to contribute noise to the mapping. The second yields noisy response curves which behave very poorly near the steady state. The first type of identification requires a detector to identify when a set point change occurs. The second type is more general since unmeasured disturbances occur at any time and can occur often in a real process. It was found that the second type of identification, with the unmeasured disturbances, was adequate for tuning purposes even though the response behavior was degraded.

PART 2. II. B. 1. a. Grid Search

A grid search is initiated where $k_c$ ranged from 0.5 to 10 and $\tau_i$ ranged from 0.5 to 5 in increments of 0.5. At each increment, the IAE is measured for the test disturbance.

Results of the search for the Discretized Difference Equation (DDE) model is shown in Figure 33 as a surface plot with IAE as one axis and $k_c$ and $\tau_i$ as the other two. The numerical values used in constructing this surface plot is given in Table 4. Note that even for this simple linear system, an area with a local minimum appears as the upper valley in the right-hand part of the graph with $k_c$ at 2 and $\tau_i$ from 3 to 5. In the narrow floor
FIGURE 33
Integral Absolute Error (IAE) vs. kc and taul
Linear Level Control in Cylindrical Tank
Step in Set Point (0.75 to 0.4); Inlet Flow Rate, 0.5
Discretized Difference Equation Model
Global Minimum at kc, 6.5 and taul, 0.5
<table>
<thead>
<tr>
<th>$k_c$</th>
<th>0.5</th>
<th>1.0</th>
<th>1.5</th>
<th>2.0</th>
<th>2.5</th>
<th>3.0</th>
<th>3.5</th>
<th>4.0</th>
<th>4.5</th>
<th>5.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2.253</td>
<td>2.634</td>
<td>2.830</td>
<td>2.864</td>
<td>2.834</td>
<td>2.818</td>
<td>2.841</td>
<td>2.887</td>
<td>2.926</td>
<td>2.947</td>
</tr>
<tr>
<td>1.5</td>
<td>1.653</td>
<td>1.813</td>
<td>1.878</td>
<td>1.910</td>
<td>1.908</td>
<td>1.913</td>
<td>1.923</td>
<td>1.942</td>
<td>1.950</td>
<td>1.928</td>
</tr>
<tr>
<td>2.0</td>
<td>1.385</td>
<td>1.451</td>
<td>1.474</td>
<td>1.467</td>
<td>1.473</td>
<td>1.476</td>
<td>1.481</td>
<td>1.490</td>
<td>1.499</td>
<td>1.511</td>
</tr>
<tr>
<td>2.5</td>
<td>1.246</td>
<td>1.266</td>
<td>1.275</td>
<td>1.282</td>
<td>1.317</td>
<td>1.368</td>
<td>1.441</td>
<td>1.552</td>
<td>1.698</td>
<td>1.843</td>
</tr>
<tr>
<td>3.0</td>
<td>1.159</td>
<td>1.166</td>
<td>1.167</td>
<td>1.209</td>
<td>1.307</td>
<td>1.452</td>
<td>1.633</td>
<td>1.814</td>
<td>1.991</td>
<td>2.162</td>
</tr>
<tr>
<td>3.5</td>
<td>1.111</td>
<td>1.108</td>
<td>1.122</td>
<td>1.205</td>
<td>1.386</td>
<td>1.593</td>
<td>1.798</td>
<td>2.001</td>
<td>2.198</td>
<td>2.388</td>
</tr>
<tr>
<td>4.0</td>
<td>1.075</td>
<td>1.073</td>
<td>1.108</td>
<td>1.250</td>
<td>1.475</td>
<td>1.699</td>
<td>1.922</td>
<td>2.140</td>
<td>2.352</td>
<td>2.557</td>
</tr>
<tr>
<td>4.5</td>
<td>1.049</td>
<td>1.050</td>
<td>1.120</td>
<td>1.306</td>
<td>1.544</td>
<td>1.782</td>
<td>2.017</td>
<td>2.248</td>
<td>2.471</td>
<td>2.687</td>
</tr>
<tr>
<td>5.0</td>
<td>1.032</td>
<td>1.042</td>
<td>1.150</td>
<td>1.350</td>
<td>1.600</td>
<td>1.848</td>
<td>2.094</td>
<td>2.334</td>
<td>2.567</td>
<td>2.791</td>
</tr>
<tr>
<td>5.5</td>
<td>1.019</td>
<td>1.045</td>
<td>1.177</td>
<td>1.386</td>
<td>1.645</td>
<td>1.902</td>
<td>2.156</td>
<td>2.404</td>
<td>2.644</td>
<td>2.876</td>
</tr>
<tr>
<td>6.0</td>
<td>1.009</td>
<td>1.058</td>
<td>1.200</td>
<td>1.417</td>
<td>1.683</td>
<td>1.947</td>
<td>2.208</td>
<td>2.462</td>
<td>2.709</td>
<td>2.946</td>
</tr>
<tr>
<td>6.5</td>
<td>1.002</td>
<td>1.071</td>
<td>1.219</td>
<td>1.442</td>
<td>1.715</td>
<td>1.985</td>
<td>2.252</td>
<td>2.512</td>
<td>2.764</td>
<td>3.006</td>
</tr>
<tr>
<td>7.0</td>
<td>1.004</td>
<td>1.082</td>
<td>1.236</td>
<td>1.464</td>
<td>1.742</td>
<td>2.018</td>
<td>2.289</td>
<td>2.554</td>
<td>2.810</td>
<td>3.057</td>
</tr>
<tr>
<td>7.5</td>
<td>1.011</td>
<td>1.092</td>
<td>1.250</td>
<td>1.483</td>
<td>1.766</td>
<td>2.046</td>
<td>2.322</td>
<td>2.591</td>
<td>2.851</td>
<td>3.101</td>
</tr>
<tr>
<td>8.0</td>
<td>1.023</td>
<td>1.100</td>
<td>1.262</td>
<td>1.500</td>
<td>1.786</td>
<td>2.071</td>
<td>2.350</td>
<td>2.623</td>
<td>2.886</td>
<td>3.139</td>
</tr>
<tr>
<td>8.5</td>
<td>1.052</td>
<td>1.107</td>
<td>1.274</td>
<td>1.515</td>
<td>1.805</td>
<td>2.092</td>
<td>2.373</td>
<td>2.651</td>
<td>2.917</td>
<td>3.173</td>
</tr>
<tr>
<td>9.0</td>
<td>1.335</td>
<td>1.114</td>
<td>1.283</td>
<td>1.528</td>
<td>1.821</td>
<td>2.112</td>
<td>2.397</td>
<td>2.676</td>
<td>2.945</td>
<td>3.203</td>
</tr>
<tr>
<td>9.5</td>
<td>1.903</td>
<td>1.120</td>
<td>1.292</td>
<td>1.539</td>
<td>1.836</td>
<td>2.129</td>
<td>2.417</td>
<td>2.698</td>
<td>2.970</td>
<td>3.230</td>
</tr>
<tr>
<td>10.0</td>
<td>1.977</td>
<td>1.128</td>
<td>1.300</td>
<td>1.550</td>
<td>1.849</td>
<td>2.145</td>
<td>2.435</td>
<td>2.718</td>
<td>2.992</td>
<td>3.255</td>
</tr>
<tr>
<td>10.5</td>
<td>1.995</td>
<td>1.162</td>
<td>1.307</td>
<td>1.559</td>
<td>1.860</td>
<td>2.159</td>
<td>2.451</td>
<td>2.737</td>
<td>3.012</td>
<td>3.277</td>
</tr>
</tbody>
</table>
of this valley, the IAE is nearly constant at about 1.48. The global minimum is in the lower valley toward the front of the plot at \( k_C \) at 6.5 and \( \tau_I \) 0.5 where the IAE is 1.01. Note that as a result of the topology of the IAE surface, if initial tuning parameters are chosen near the local minimum, then a trial-and-error search with small search step sizes may get "stuck" in the upper valley and not be able to find a path to the global minimum.

A similar surface plot constructed using the network model is shown in Figure 34. The corresponding values are listed in Table 5. A comparison of this plot with the previous one reveals that the two surfaces have similar topology in that higher IAE values appear at high and low values of \( k_C \) and the lower IAE values occurs for low \( \tau_I \). Some topological features are different, but they are not significant. In particular, this new surface has higher IAE values and appears to have been raised from the previous one.

The most important commonality is the location of the global minimum which occurs at \( k_C \) 4.5 and \( \tau_I \) 0.5 as opposed to 6.5 and 0.5 previously. The \( k_C \) values for the two cases are different by 30%. The actual difference in IAE value is only 5%. This indicates that in this case the neural network model is accurate enough to represent the actual model and can be used to successfully locate the
FIGURE 34
Integral Absolute Error (IAE) vs. kc and tau_i
Linear Level Control in Cylindrical Tank
Step in Set Point (0.75 to 0.4); Inlet Flow Rate, 0.5
Neural Network Model
Global Minimum at kc, 4.5 and tau_i, 0.5
<table>
<thead>
<tr>
<th>$k_c$</th>
<th>0.5</th>
<th>1.0</th>
<th>1.5</th>
<th>2.0</th>
<th>2.5</th>
<th>3.0</th>
<th>3.5</th>
<th>4.0</th>
<th>4.5</th>
<th>5.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>4.060</td>
<td>4.140</td>
<td>3.298</td>
<td>3.580</td>
<td>4.797</td>
<td>5.939</td>
<td>7.103</td>
<td>8.218</td>
<td>8.849</td>
<td>9.306</td>
</tr>
<tr>
<td>1.5</td>
<td>1.793</td>
<td>1.982</td>
<td>2.209</td>
<td>2.220</td>
<td>2.131</td>
<td>2.042</td>
<td>1.979</td>
<td>2.001</td>
<td>2.186</td>
<td>2.408</td>
</tr>
<tr>
<td>2</td>
<td>1.468</td>
<td>1.616</td>
<td>1.712</td>
<td>1.683</td>
<td>1.633</td>
<td>1.826</td>
<td>2.097</td>
<td>2.370</td>
<td>2.641</td>
<td>2.904</td>
</tr>
<tr>
<td>2.5</td>
<td>1.306</td>
<td>1.406</td>
<td>1.447</td>
<td>1.430</td>
<td>1.674</td>
<td>1.965</td>
<td>2.250</td>
<td>2.535</td>
<td>2.805</td>
<td>3.071</td>
</tr>
<tr>
<td>3</td>
<td>1.212</td>
<td>1.276</td>
<td>1.297</td>
<td>1.446</td>
<td>1.746</td>
<td>2.050</td>
<td>2.343</td>
<td>2.630</td>
<td>2.907</td>
<td>3.179</td>
</tr>
<tr>
<td>3.5</td>
<td>1.157</td>
<td>1.204</td>
<td>1.240</td>
<td>1.500</td>
<td>1.806</td>
<td>2.112</td>
<td>2.403</td>
<td>2.696</td>
<td>2.981</td>
<td>3.256</td>
</tr>
<tr>
<td>4</td>
<td>1.117</td>
<td>1.157</td>
<td>1.272</td>
<td>1.534</td>
<td>1.845</td>
<td>2.150</td>
<td>2.454</td>
<td>2.749</td>
<td>3.033</td>
<td>3.311</td>
</tr>
<tr>
<td>4.5</td>
<td>1.086</td>
<td>1.130</td>
<td>1.296</td>
<td>1.559</td>
<td>1.872</td>
<td>2.185</td>
<td>2.484</td>
<td>2.786</td>
<td>3.073</td>
<td>3.354</td>
</tr>
<tr>
<td>5</td>
<td>1.088</td>
<td>1.121</td>
<td>1.313</td>
<td>1.578</td>
<td>1.898</td>
<td>2.207</td>
<td>2.517</td>
<td>2.817</td>
<td>3.109</td>
<td>3.388</td>
</tr>
<tr>
<td>5.5</td>
<td>1.612</td>
<td>1.135</td>
<td>1.329</td>
<td>1.595</td>
<td>1.915</td>
<td>2.231</td>
<td>2.540</td>
<td>2.842</td>
<td>3.134</td>
<td>3.416</td>
</tr>
<tr>
<td>6</td>
<td>2.347</td>
<td>1.143</td>
<td>1.340</td>
<td>1.610</td>
<td>1.932</td>
<td>2.249</td>
<td>2.559</td>
<td>2.864</td>
<td>3.156</td>
<td>3.437</td>
</tr>
<tr>
<td>7.5</td>
<td>2.377</td>
<td>1.250</td>
<td>1.911</td>
<td>2.130</td>
<td>2.088</td>
<td>2.531</td>
<td>2.818</td>
<td>3.105</td>
<td>3.405</td>
<td>3.692</td>
</tr>
<tr>
<td>8.5</td>
<td>2.450</td>
<td>2.541</td>
<td>2.610</td>
<td>2.857</td>
<td>3.136</td>
<td>3.594</td>
<td>3.889</td>
<td>4.221</td>
<td>4.503</td>
<td>4.850</td>
</tr>
</tbody>
</table>
tuning parameters that are very close to the global minimum on the IAE surface.

PART 2. II. B. 1. b. Disturbance Tests

When the predicted global optimal tuning parameters are implemented, it is expected that significant improvements should occur in the responses to disturbances. Figure 35 shows the response of step up in set point from 0.3 to 0.8 for three different tunings, the initial tuning \((k_c, 1.5\) and \(\tau_u, 2.0\)), the global minimum or optimal tuning predicted by the network \((k_c, 4.5\) and \(\tau_u, 0.5\)) and the actual optimal tuning predicted by the DDE model \((k_c, 6.5\) and \(\tau_u, 0.5\)). Note that the liquid level responses of the network and optimal tunings are similar. The response for network tuning is a significant improvement over that of the initial tuning in that it has negligible overshoot. As the lower graph shows, there is a more of a difference in controller variable action, outlet flow rate, for the optimal and network tuning cases.

An additional test is performed for a step down in inlet flow rate from 0.75 to 0.2, and the results are shown in Figure 36. Note that the tuning parameters predicted by the network, which were determined by
FIGURE 35
Comparison of Responses for Different Tunings
Step Up in Set Point
Simulation of Linear System

Normalized Liquid Level

- Set Point
- Initial Tuning
- Network Optimum Tuning
- Actual Optimum Tuning

Normalized Flow Rate

- Inlet
- Outlet (Initial Tuning; kc, 1.5 & tau_i, 2.0)
- Outlet (Network Optimum; kc, 4.5 & tau_i, 0.5)
- Outlet (Actual Optimum; kc, 6.5 & tau_i, 0.5)

Sample Time

0 5 10 15 20 25 30 35
FIGURE 36
Comparison of Responses for Different Tunings
Step Down in Inlet Flow Rate
Simulation of Linear System

Normalized Liquid Level

Set Point
Initial Tuning (kc 1.5, tau 2.0)
Network Optimum Tuning (kc, 4.5 & tau, 0.5)
Actual Optimum Tuning (kc, 6.5 & tau, 0.5)

Normalized Flow Rate

Inlet
Outlet (Initial Tuning)
Outlet (Network Optimum)
Outlet (Actual Optimum)

Sample Time
minimizing the IAE for a set point disturbance, also yield substantial improvement in IAE for load disturbances such as steps in inlet flow rate.

PART 2. II. B. 1. c. Comments and Conclusions

An excellent estimate of the location of the tuning parameters which corresponds to the global minimum in IAE was found, and the new tuning parameters were implemented. The change in $k_c$ from the initial tuning was 600% and the change in $\tau_i$ was -75%. Such a dramatic adjustment in the tuning parameters is probably not attainable for a real process. One must consider modeling error and remain skeptical of predicted tuning parameters which are far from those currently shown by experience to be adequate. Perhaps, the network tuning parameters should be used as pointers to new parameters. The actual implementation may be carried out in a step-wise manner in the direction of the network optimum. New tuning parameters can be selected which are between the existing and network optimum. After the new parameters are implemented, a second identification may be warranted which would add information to the first. If the second identification results in a model which gives a similar recommendation as that of the first, then the operator of the process can
continue to adjust the tuning parameters slowly toward the recommended optimal setting. In this manner, the operator can build confidence in the validity of network model recommendation.

In summary, the neural network model of the simulated linear system is shown to be adequate for determining tuning parameters which are close to the IAE global minimum value. A network model was identified using only disturbance data from the initial tuning, and required only one identification session. The resulting performance of the system with the new tuning parameters was very similar to that using tuning parameters corresponding to the actual global minimum.

PART 2. II. B. 2. Nonlinear Simulated System

The potential of using a neural network for controller tuning is further explored with a nonlinear system. In this case, the IAE of disturbances at various set points are calculated in order to represent the behavior of the system under different operating regimes. Two disturbances are considered, a step down in set point from 0.9 to 0.6 (upper tank disturbance) and a step up in set point from 0.2 to 0.4 (lower tank disturbance). The inlet flow rate for both disturbances is held constant at
0.5. As in the linear case, the same grid of $k_c$ and $\tau_i$ values is used to calculate IAE.

PART 2. II. B. 2. a. Grid Search

A grid of IAE values for the upper disturbance is calculated using the DDE model for the range of $k_c$ from 0.0 to 10.5 and that for $\tau_i$ from 0.0 to 5.0. A surface plot of these values is shown in Figure 37 and the values are listed in Table 6. The principle topology of this surface is similar to that of the linear system except for some distortion. Most notable are for the region of high $k_c$ where the surface slope up instead of down and the higher walls for low $\tau_i$ and high $k_c$. The global minimum for this surface is $k_c$, 5.5 and $\tau_i$, 0.5.

The process is repeated with the network model. The resulting surface plot is shown in Figure 38, and the values are listed in Table 7. Again, the topology is similar to that of the DDE model, but minor features are different due to modeling error. The global minimum for this surface is $k_c$, 4.0 and $\tau_i$, 0.5. This value of $k_c$ is lower than that of the DDE model by 28% and the $\tau_i$ value is exact. The actual global minimum IAE value at $k_c$, 5.5 and $\tau_i$, 0.5 is 0.617, and the actual value at $k_c$, 4.0 and $\tau_i$, 0.5 is 0.644 which is 4% higher. Once
FIGURE 37
Integral Absolute Error (IAE) vs. kc and tau1
Nonlinear Level Control in Conical Tank
Discretized Difference Equation Model - Upper Disturbance
Step in Set Point (0.9 to 0.6); Inlet Flow Rate, 0.5
Global Minimum at kc, 5.5 and tau1, 0.5
TABLE 6
IAE Non-Linear Tank
Discretized Differential Equation Model

Upper Disturbance - Step Up in Set Point (0.9 to 0.6);
Inlet Flow Rate, 0.5;
Forty Time Steps into Future

Global Minimum at $k_c$, 5.5 and $\tau_u$ 0.5

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>------</td>
<td>-----</td>
<td>---</td>
<td>-----</td>
<td>---</td>
<td>-----</td>
<td>---</td>
<td>-----</td>
<td>---</td>
<td>-----</td>
<td>---</td>
</tr>
<tr>
<td>1</td>
<td>1.813</td>
<td>2.067</td>
<td>2.036</td>
<td>2.051</td>
<td>2.040</td>
<td>2.018</td>
<td>2.025</td>
<td>2.042</td>
<td>2.072</td>
<td>2.092</td>
</tr>
<tr>
<td>1.5</td>
<td>1.225</td>
<td>1.331</td>
<td>1.335</td>
<td>1.352</td>
<td>1.378</td>
<td>1.395</td>
<td>1.413</td>
<td>1.438</td>
<td>1.475</td>
<td>1.496</td>
</tr>
<tr>
<td>2</td>
<td>0.963</td>
<td>0.992</td>
<td>0.979</td>
<td>0.967</td>
<td>0.967</td>
<td>0.963</td>
<td>0.958</td>
<td>0.956</td>
<td>0.953</td>
<td>0.949</td>
</tr>
<tr>
<td>2.5</td>
<td>0.826</td>
<td>0.826</td>
<td>0.828</td>
<td>0.848</td>
<td>0.876</td>
<td>0.923</td>
<td>1.001</td>
<td>1.099</td>
<td>1.197</td>
<td>1.292</td>
</tr>
<tr>
<td>3</td>
<td>0.748</td>
<td>0.738</td>
<td>0.783</td>
<td>0.858</td>
<td>0.968</td>
<td>1.100</td>
<td>1.232</td>
<td>1.363</td>
<td>1.490</td>
<td>1.613</td>
</tr>
<tr>
<td>3.5</td>
<td>0.698</td>
<td>0.701</td>
<td>0.792</td>
<td>0.929</td>
<td>1.086</td>
<td>1.242</td>
<td>1.398</td>
<td>1.550</td>
<td>1.698</td>
<td>1.841</td>
</tr>
<tr>
<td>4</td>
<td>0.667</td>
<td>0.682</td>
<td>0.827</td>
<td>1.000</td>
<td>1.175</td>
<td>1.349</td>
<td>1.521</td>
<td>1.690</td>
<td>1.854</td>
<td>2.011</td>
</tr>
<tr>
<td>4.5</td>
<td>0.644</td>
<td>0.686</td>
<td>0.867</td>
<td>1.056</td>
<td>1.244</td>
<td>1.432</td>
<td>1.617</td>
<td>1.798</td>
<td>1.974</td>
<td>2.143</td>
</tr>
<tr>
<td>5</td>
<td>0.628</td>
<td>0.700</td>
<td>0.900</td>
<td>1.100</td>
<td>1.300</td>
<td>1.498</td>
<td>1.694</td>
<td>1.885</td>
<td>2.070</td>
<td>2.248</td>
</tr>
<tr>
<td>5.5</td>
<td>0.617</td>
<td>0.718</td>
<td>0.927</td>
<td>1.136</td>
<td>1.345</td>
<td>1.552</td>
<td>1.756</td>
<td>1.955</td>
<td>2.148</td>
<td>2.334</td>
</tr>
<tr>
<td>6</td>
<td>0.617</td>
<td>0.733</td>
<td>0.950</td>
<td>1.167</td>
<td>1.383</td>
<td>1.597</td>
<td>1.808</td>
<td>2.014</td>
<td>2.213</td>
<td>2.405</td>
</tr>
<tr>
<td>6.5</td>
<td>0.624</td>
<td>0.746</td>
<td>0.969</td>
<td>1.192</td>
<td>1.415</td>
<td>1.635</td>
<td>1.852</td>
<td>2.064</td>
<td>2.269</td>
<td>2.465</td>
</tr>
<tr>
<td>7</td>
<td>0.639</td>
<td>0.757</td>
<td>0.986</td>
<td>1.214</td>
<td>1.442</td>
<td>1.668</td>
<td>1.890</td>
<td>2.106</td>
<td>2.316</td>
<td>2.517</td>
</tr>
<tr>
<td>7.5</td>
<td>0.680</td>
<td>0.767</td>
<td>1.000</td>
<td>1.233</td>
<td>1.466</td>
<td>1.696</td>
<td>1.923</td>
<td>2.143</td>
<td>2.357</td>
<td>2.561</td>
</tr>
<tr>
<td>8</td>
<td>0.987</td>
<td>0.775</td>
<td>1.012</td>
<td>1.250</td>
<td>1.487</td>
<td>1.721</td>
<td>1.951</td>
<td>2.176</td>
<td>2.392</td>
<td>2.600</td>
</tr>
<tr>
<td>8.5</td>
<td>1.840</td>
<td>0.782</td>
<td>1.024</td>
<td>1.265</td>
<td>1.505</td>
<td>1.743</td>
<td>1.976</td>
<td>2.204</td>
<td>2.424</td>
<td>2.635</td>
</tr>
<tr>
<td>9</td>
<td>1.882</td>
<td>0.799</td>
<td>1.033</td>
<td>1.278</td>
<td>1.521</td>
<td>1.762</td>
<td>1.999</td>
<td>2.229</td>
<td>2.452</td>
<td>2.665</td>
</tr>
<tr>
<td>9.5</td>
<td>1.885</td>
<td>0.919</td>
<td>1.050</td>
<td>1.289</td>
<td>1.542</td>
<td>1.791</td>
<td>2.032</td>
<td>2.265</td>
<td>2.489</td>
<td>2.703</td>
</tr>
<tr>
<td>10</td>
<td>1.897</td>
<td>1.775</td>
<td>1.289</td>
<td>1.376</td>
<td>1.603</td>
<td>1.874</td>
<td>2.137</td>
<td>2.391</td>
<td>2.635</td>
<td>2.868</td>
</tr>
<tr>
<td>10.5</td>
<td>1.909</td>
<td>1.935</td>
<td>2.110</td>
<td>1.896</td>
<td>1.936</td>
<td>2.112</td>
<td>2.368</td>
<td>2.687</td>
<td>2.999</td>
<td>3.310</td>
</tr>
</tbody>
</table>
FIGURE 38
Integral Absolute Error (IAE) vs. kc and tau_i
Nonlinear Level Control in Conical Tank
Step in Set Point (0.9 to 0.6); Inlet Flow Rate, 0.5
Neural Network Model
Global Minimum at kc, 4.0 and tau_i, 0.5
### TABLE 7

**IAE Nonlinear Tank Network Simulation**

*Upper Disturbance – Step Up in Set Point (0.9 to 0.6); Inlet Flow Rate, 0.5; Twenty Time Steps Into Future*

**Global Minimum at \( k_c \), 4.0 and \( \tau_1 \), 0.5**

<table>
<thead>
<tr>
<th>( k_c )</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>5.78</td>
<td>5.247</td>
<td>4.951</td>
<td>4.737</td>
<td>4.594</td>
<td>4.472</td>
<td>4.34</td>
<td>4.259</td>
<td>4.202</td>
<td>4.188</td>
</tr>
<tr>
<td>1.5</td>
<td>1.482</td>
<td>2.139</td>
<td>2.799</td>
<td>2.996</td>
<td>3.015</td>
<td>3.079</td>
<td>3.043</td>
<td>3.001</td>
<td>2.957</td>
<td>2.904</td>
</tr>
<tr>
<td>2</td>
<td>1.098</td>
<td>1.189</td>
<td>1.402</td>
<td>1.534</td>
<td>1.56</td>
<td>1.55</td>
<td>1.514</td>
<td>1.482</td>
<td>1.451</td>
<td>1.404</td>
</tr>
<tr>
<td>2.5</td>
<td>0.892</td>
<td>0.968</td>
<td>1.087</td>
<td>1.144</td>
<td>1.161</td>
<td>1.173</td>
<td>1.164</td>
<td>1.178</td>
<td>1.172</td>
<td>1.184</td>
</tr>
<tr>
<td>3</td>
<td>0.773</td>
<td>0.851</td>
<td>0.955</td>
<td>1.036</td>
<td>1.084</td>
<td>1.13</td>
<td>1.176</td>
<td>1.221</td>
<td>1.274</td>
<td>1.329</td>
</tr>
<tr>
<td>3.5</td>
<td>0.713</td>
<td>0.804</td>
<td>0.908</td>
<td>1.019</td>
<td>1.095</td>
<td>1.18</td>
<td>1.259</td>
<td>1.335</td>
<td>1.425</td>
<td>1.509</td>
</tr>
<tr>
<td>4</td>
<td>0.69</td>
<td>0.777</td>
<td>0.905</td>
<td>1.029</td>
<td>1.139</td>
<td>1.246</td>
<td>1.344</td>
<td>1.446</td>
<td>1.556</td>
<td>1.647</td>
</tr>
<tr>
<td>4.5</td>
<td>0.696</td>
<td>0.758</td>
<td>0.915</td>
<td>1.052</td>
<td>1.182</td>
<td>1.302</td>
<td>1.42</td>
<td>1.545</td>
<td>1.659</td>
<td>1.752</td>
</tr>
<tr>
<td>5</td>
<td>0.767</td>
<td>0.749</td>
<td>0.937</td>
<td>1.082</td>
<td>1.223</td>
<td>1.354</td>
<td>1.497</td>
<td>1.629</td>
<td>1.739</td>
<td>1.849</td>
</tr>
<tr>
<td>5.5</td>
<td>0.936</td>
<td>0.771</td>
<td>0.963</td>
<td>1.108</td>
<td>1.264</td>
<td>1.408</td>
<td>1.555</td>
<td>1.693</td>
<td>1.806</td>
<td>1.907</td>
</tr>
<tr>
<td>6</td>
<td>1.051</td>
<td>0.806</td>
<td>0.994</td>
<td>1.152</td>
<td>1.319</td>
<td>1.467</td>
<td>1.605</td>
<td>1.744</td>
<td>1.864</td>
<td>1.966</td>
</tr>
<tr>
<td>6.5</td>
<td>1.238</td>
<td>0.858</td>
<td>1.064</td>
<td>1.232</td>
<td>1.383</td>
<td>1.539</td>
<td>1.671</td>
<td>1.796</td>
<td>1.912</td>
<td>2.02</td>
</tr>
<tr>
<td>7</td>
<td>1.303</td>
<td>0.949</td>
<td>1.194</td>
<td>1.349</td>
<td>1.487</td>
<td>1.626</td>
<td>1.763</td>
<td>1.878</td>
<td>1.978</td>
<td>2.077</td>
</tr>
<tr>
<td>7.5</td>
<td>1.389</td>
<td>1.031</td>
<td>1.322</td>
<td>1.466</td>
<td>1.628</td>
<td>1.75</td>
<td>1.854</td>
<td>1.969</td>
<td>2.057</td>
<td>2.136</td>
</tr>
<tr>
<td>8</td>
<td>1.453</td>
<td>1.222</td>
<td>1.584</td>
<td>1.695</td>
<td>1.831</td>
<td>1.938</td>
<td>2.03</td>
<td>2.104</td>
<td>2.177</td>
<td>2.246</td>
</tr>
<tr>
<td>8.5</td>
<td>1.536</td>
<td>1.373</td>
<td>1.893</td>
<td>2.069</td>
<td>2.176</td>
<td>2.29</td>
<td>2.393</td>
<td>2.453</td>
<td>2.496</td>
<td>2.537</td>
</tr>
<tr>
<td>9</td>
<td>1.679</td>
<td>1.528</td>
<td>2.207</td>
<td>2.357</td>
<td>2.469</td>
<td>2.57</td>
<td>2.658</td>
<td>2.737</td>
<td>2.774</td>
<td>2.823</td>
</tr>
<tr>
<td>9.5</td>
<td>1.639</td>
<td>1.763</td>
<td>2.311</td>
<td>2.475</td>
<td>2.67</td>
<td>2.812</td>
<td>2.896</td>
<td>2.98</td>
<td>3.068</td>
<td>3.119</td>
</tr>
<tr>
<td>10</td>
<td>1.678</td>
<td>2.072</td>
<td>2.352</td>
<td>2.545</td>
<td>2.808</td>
<td>2.967</td>
<td>3.157</td>
<td>3.264</td>
<td>3.392</td>
<td>3.466</td>
</tr>
<tr>
<td>10.5</td>
<td>1.845</td>
<td>2.246</td>
<td>2.403</td>
<td>2.642</td>
<td>2.901</td>
<td>3.119</td>
<td>3.369</td>
<td>3.565</td>
<td>3.724</td>
<td>3.777</td>
</tr>
</tbody>
</table>
again, even though the value of $k_C$ predicted by the network is not exact, the effect on IAE resulting from adopting the tuning parameters of the network model are nearly equivalent to that of the actual model for this upper disturbance.

The lower disturbance was analyzed in the same manner. For IAE calculated with the DDE model, a surface plot of the grid is shown in Figure 39, and the values are listed in Table 8. The topology of this surface is different from that of the upper disturbance in that the hills are not quite as high. In addition, the global minimum $k_C$ value moves from 5.5 to 7.5.

The surface plot IAE from network simulation for the lower disturbance is shown in Figure 40 and the values are listed in Table 9. This surface is similar to that of the DDE model except for distortions of the hill at low $k_C$ and of the valley at low $\tau_i$. The global minimum has changed from $k_C$, 7.5 and $\tau_i$, 0.5 to $k_C$, 4.0 and $\tau_i$, 1.0 or a reduction of 47% for $k_C$ and an increase of 100% for $\tau_i$. The actual IAE value at $k_C$, 7.5 and $\tau_i$, 0.5 is 0.419 and that at $k_C$, 4.0 and $\tau_i$, 1.0 is 0.512 or a difference of 22%.
FIGURE 39
Integral Absolute Error (IAE) vs. kc and tau
Nonlinear Level Control in Conical Tank
Discretized Difference Equation Model - Lower Disturbance
Step in Set Point (0.2 to 0.4); Inlet Flow Rate, 0.5
Global Minimum at kc, 7.5 and tau, 0.5
TABLE 8

IAE Non-Linear Tank

Discretized Differential Equation Model

Lower Disturbance - Step Up in Set Point from 0.2 to 0.4;
Inlet Flow Rate, 0.5;
Forty Time Steps into Future

Global Minimum at \( k_c \), 7.5 and \( \tau_i \) 0.5

<table>
<thead>
<tr>
<th>( k_c )</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>2.768</td>
<td>2.799</td>
<td>2.673</td>
<td>2.740</td>
<td>2.860</td>
<td>2.880</td>
<td>2.851</td>
<td>2.800</td>
<td>2.741</td>
<td>2.687</td>
</tr>
<tr>
<td>1</td>
<td>1.758</td>
<td>1.803</td>
<td>1.813</td>
<td>1.818</td>
<td>1.795</td>
<td>1.798</td>
<td>1.817</td>
<td>1.857</td>
<td>1.878</td>
<td>1.885</td>
</tr>
<tr>
<td>1.5</td>
<td>1.116</td>
<td>1.285</td>
<td>1.288</td>
<td>1.300</td>
<td>1.311</td>
<td>1.316</td>
<td>1.319</td>
<td>1.335</td>
<td>1.357</td>
<td>1.372</td>
</tr>
<tr>
<td>2</td>
<td>0.837</td>
<td>0.935</td>
<td>0.977</td>
<td>1.003</td>
<td>1.013</td>
<td>1.029</td>
<td>1.039</td>
<td>1.051</td>
<td>1.072</td>
<td>1.087</td>
</tr>
<tr>
<td>2.5</td>
<td>0.683</td>
<td>0.740</td>
<td>0.752</td>
<td>0.753</td>
<td>0.766</td>
<td>0.776</td>
<td>0.784</td>
<td>0.795</td>
<td>0.807</td>
<td>0.814</td>
</tr>
<tr>
<td>3</td>
<td>0.601</td>
<td>0.628</td>
<td>0.624</td>
<td>0.627</td>
<td>0.622</td>
<td>0.624</td>
<td>0.620</td>
<td>0.616</td>
<td>0.611</td>
<td>0.606</td>
</tr>
<tr>
<td>3.5</td>
<td>0.544</td>
<td>0.558</td>
<td>0.556</td>
<td>0.558</td>
<td>0.563</td>
<td>0.573</td>
<td>0.600</td>
<td>0.657</td>
<td>0.712</td>
<td>0.767</td>
</tr>
<tr>
<td>4</td>
<td>0.508</td>
<td>0.513</td>
<td>0.518</td>
<td>0.536</td>
<td>0.576</td>
<td>0.650</td>
<td>0.724</td>
<td>0.798</td>
<td>0.870</td>
<td>0.939</td>
</tr>
<tr>
<td>4.5</td>
<td>0.482</td>
<td>0.482</td>
<td>0.501</td>
<td>0.556</td>
<td>0.644</td>
<td>0.733</td>
<td>0.821</td>
<td>0.907</td>
<td>0.991</td>
<td>1.073</td>
</tr>
<tr>
<td>5</td>
<td>0.462</td>
<td>0.460</td>
<td>0.504</td>
<td>0.600</td>
<td>0.700</td>
<td>0.800</td>
<td>0.898</td>
<td>0.995</td>
<td>1.088</td>
<td>1.179</td>
</tr>
<tr>
<td>5.5</td>
<td>0.447</td>
<td>0.453</td>
<td>0.527</td>
<td>0.636</td>
<td>0.745</td>
<td>0.854</td>
<td>0.961</td>
<td>1.066</td>
<td>1.168</td>
<td>1.266</td>
</tr>
<tr>
<td>6</td>
<td>0.437</td>
<td>0.452</td>
<td>0.550</td>
<td>0.667</td>
<td>0.783</td>
<td>0.899</td>
<td>1.013</td>
<td>1.125</td>
<td>1.234</td>
<td>1.338</td>
</tr>
<tr>
<td>6.5</td>
<td>0.428</td>
<td>0.461</td>
<td>0.569</td>
<td>0.692</td>
<td>0.815</td>
<td>0.937</td>
<td>1.058</td>
<td>1.175</td>
<td>1.289</td>
<td>1.399</td>
</tr>
<tr>
<td>7</td>
<td>0.422</td>
<td>0.472</td>
<td>0.586</td>
<td>0.714</td>
<td>0.843</td>
<td>0.970</td>
<td>1.095</td>
<td>1.218</td>
<td>1.337</td>
<td>1.451</td>
</tr>
<tr>
<td>7.5</td>
<td>0.419</td>
<td>0.482</td>
<td>0.600</td>
<td>0.733</td>
<td>0.866</td>
<td>0.998</td>
<td>1.128</td>
<td>1.255</td>
<td>1.378</td>
<td>1.496</td>
</tr>
<tr>
<td>8</td>
<td>0.425</td>
<td>0.490</td>
<td>0.612</td>
<td>0.750</td>
<td>0.887</td>
<td>1.023</td>
<td>1.157</td>
<td>1.288</td>
<td>1.414</td>
<td>1.535</td>
</tr>
<tr>
<td>8.5</td>
<td>0.438</td>
<td>0.497</td>
<td>0.624</td>
<td>0.765</td>
<td>0.905</td>
<td>1.045</td>
<td>1.182</td>
<td>1.316</td>
<td>1.446</td>
<td>1.570</td>
</tr>
<tr>
<td>9</td>
<td>0.476</td>
<td>0.504</td>
<td>0.633</td>
<td>0.778</td>
<td>0.922</td>
<td>1.064</td>
<td>1.205</td>
<td>1.342</td>
<td>1.474</td>
<td>1.601</td>
</tr>
<tr>
<td>9.5</td>
<td>0.811</td>
<td>1.100</td>
<td>0.642</td>
<td>0.789</td>
<td>0.936</td>
<td>1.082</td>
<td>1.225</td>
<td>1.364</td>
<td>1.499</td>
<td>1.629</td>
</tr>
<tr>
<td>10</td>
<td>1.436</td>
<td>1.515</td>
<td>0.650</td>
<td>0.800</td>
<td>0.949</td>
<td>1.097</td>
<td>1.243</td>
<td>1.385</td>
<td>1.522</td>
<td>1.653</td>
</tr>
<tr>
<td>10.5</td>
<td>1.486</td>
<td>0.520</td>
<td>0.657</td>
<td>0.809</td>
<td>0.961</td>
<td>1.112</td>
<td>1.259</td>
<td>1.403</td>
<td>1.542</td>
<td>1.676</td>
</tr>
</tbody>
</table>
FIGURE 40
Integral Absolute Error (IAE) vs. kc and taul
Nonlinear Level Control in Conical Tank
Neural Network Model - Lower Disturbance
Set Point Step (0.2 to 0.4); Inlet Flow Rate, 0.5
Global Minimum at kc, 4.0 and taul, 1.0
### TABLE 9

**IAE Nonlinear Tank Network Simulation**

Lower Disturbance - Step Up in Set Point (0.2 to 0.4);
Inlet Flow Rate, 0.5;
Twenty Time Steps in Future

**Global Minimum at $k_C$, 4.0 and $\tau_u$, 1.0**

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>0.5</td>
<td>3.054</td>
<td>3.586</td>
<td>3.327</td>
<td>2.514</td>
<td>2.805</td>
<td>2.506</td>
<td>2.963</td>
<td>3.209</td>
<td>3.440</td>
<td>3.493</td>
</tr>
<tr>
<td>1</td>
<td>2.582</td>
<td>2.482</td>
<td>2.440</td>
<td>2.690</td>
<td>2.815</td>
<td>2.826</td>
<td>2.756</td>
<td>2.634</td>
<td>2.526</td>
<td>2.384</td>
</tr>
<tr>
<td>1.5</td>
<td>1.644</td>
<td>1.904</td>
<td>1.825</td>
<td>1.805</td>
<td>1.907</td>
<td>2.046</td>
<td>2.115</td>
<td>2.149</td>
<td>2.169</td>
<td>2.173</td>
</tr>
<tr>
<td>2</td>
<td>1.102</td>
<td>1.124</td>
<td>1.265</td>
<td>1.347</td>
<td>1.408</td>
<td>1.506</td>
<td>1.583</td>
<td>1.626</td>
<td>1.653</td>
<td>1.670</td>
</tr>
<tr>
<td>2.5</td>
<td>0.845</td>
<td>0.807</td>
<td>0.863</td>
<td>0.940</td>
<td>1.005</td>
<td>1.074</td>
<td>1.121</td>
<td>1.149</td>
<td>1.171</td>
<td>1.184</td>
</tr>
<tr>
<td>3</td>
<td>0.696</td>
<td>0.649</td>
<td>0.689</td>
<td>0.739</td>
<td>0.766</td>
<td>0.771</td>
<td>0.769</td>
<td>0.769</td>
<td>0.760</td>
<td>0.761</td>
</tr>
<tr>
<td>3.5</td>
<td>0.669</td>
<td>0.579</td>
<td>0.605</td>
<td>0.616</td>
<td>0.616</td>
<td>0.626</td>
<td>0.636</td>
<td>0.648</td>
<td>0.663</td>
<td>0.677</td>
</tr>
<tr>
<td>4</td>
<td>0.732</td>
<td>0.542</td>
<td>0.547</td>
<td>0.564</td>
<td>0.596</td>
<td>0.632</td>
<td>0.675</td>
<td>0.720</td>
<td>0.773</td>
<td>0.819</td>
</tr>
<tr>
<td>4.5</td>
<td>0.811</td>
<td>0.543</td>
<td>0.550</td>
<td>0.572</td>
<td>0.631</td>
<td>0.690</td>
<td>0.755</td>
<td>0.824</td>
<td>0.900</td>
<td>0.975</td>
</tr>
<tr>
<td>5</td>
<td>0.914</td>
<td>0.572</td>
<td>0.559</td>
<td>0.600</td>
<td>0.672</td>
<td>0.748</td>
<td>0.828</td>
<td>0.920</td>
<td>1.007</td>
<td>1.091</td>
</tr>
<tr>
<td>5.5</td>
<td>1.035</td>
<td>0.612</td>
<td>0.593</td>
<td>0.631</td>
<td>0.712</td>
<td>0.797</td>
<td>0.897</td>
<td>0.995</td>
<td>1.090</td>
<td>1.177</td>
</tr>
<tr>
<td>6</td>
<td>1.000</td>
<td>0.654</td>
<td>0.628</td>
<td>0.663</td>
<td>0.746</td>
<td>0.848</td>
<td>0.954</td>
<td>1.058</td>
<td>1.157</td>
<td>1.248</td>
</tr>
<tr>
<td>6.5</td>
<td>1.017</td>
<td>0.660</td>
<td>0.657</td>
<td>0.697</td>
<td>0.783</td>
<td>0.887</td>
<td>0.999</td>
<td>1.105</td>
<td>1.208</td>
<td>1.305</td>
</tr>
<tr>
<td>7</td>
<td>1.023</td>
<td>0.741</td>
<td>0.715</td>
<td>0.742</td>
<td>0.820</td>
<td>0.925</td>
<td>1.035</td>
<td>1.147</td>
<td>1.251</td>
<td>1.347</td>
</tr>
<tr>
<td>7.5</td>
<td>0.959</td>
<td>0.811</td>
<td>0.769</td>
<td>0.790</td>
<td>0.857</td>
<td>0.955</td>
<td>1.068</td>
<td>1.182</td>
<td>1.288</td>
<td>1.388</td>
</tr>
<tr>
<td>8</td>
<td>1.006</td>
<td>0.855</td>
<td>0.804</td>
<td>0.815</td>
<td>0.884</td>
<td>0.984</td>
<td>1.096</td>
<td>1.210</td>
<td>1.317</td>
<td>1.418</td>
</tr>
<tr>
<td>8.5</td>
<td>0.916</td>
<td>0.945</td>
<td>0.850</td>
<td>0.846</td>
<td>0.909</td>
<td>1.008</td>
<td>1.116</td>
<td>1.233</td>
<td>1.345</td>
<td>1.445</td>
</tr>
<tr>
<td>9</td>
<td>0.943</td>
<td>0.943</td>
<td>0.891</td>
<td>0.891</td>
<td>0.938</td>
<td>1.027</td>
<td>1.136</td>
<td>1.255</td>
<td>1.366</td>
<td>1.470</td>
</tr>
<tr>
<td>9.5</td>
<td>1.016</td>
<td>0.923</td>
<td>0.880</td>
<td>0.901</td>
<td>0.944</td>
<td>1.040</td>
<td>1.155</td>
<td>1.275</td>
<td>1.387</td>
<td>1.491</td>
</tr>
<tr>
<td>10</td>
<td>0.949</td>
<td>0.909</td>
<td>0.925</td>
<td>0.918</td>
<td>0.953</td>
<td>1.056</td>
<td>1.173</td>
<td>1.294</td>
<td>1.409</td>
<td>1.512</td>
</tr>
<tr>
<td>10.5</td>
<td>0.809</td>
<td>0.914</td>
<td>0.944</td>
<td>0.933</td>
<td>0.980</td>
<td>1.074</td>
<td>1.193</td>
<td>1.314</td>
<td>1.426</td>
<td>1.530</td>
</tr>
</tbody>
</table>
PART 2. II. B. 2. b. Second Training Cycle

The prediction of the global optimal tuning parameters is a good estimate for the upper disturbance and a fair one for the lower disturbance. Perhaps these predictions can be improved if the recommended tuning parameters of the network model are implemented and the network retrained with any new dynamical information that results from the retuned system. The retuned controller is closer to the optimal tuning. For any disturbance, the period containing dynamic information is shorter, hence there is less dynamical information in the disturbance data. Even with this fact, there should be enough additional information to make a difference.

PART 2. II. B. 2. b. 1. Grid Search

To begin the second identification process, new tuning parameters must be selected and implemented. The new parameters are those which correspond to the global minimum of a composite IAE parameter formed by the sum of the IAE data for the upper and lower disturbances. A surface plot of this composite parameter is presented in Figure 41 and the values are listed in Table 10. The global minimum IAE for this surface is $k_c$, 4.5 and $\tau_{ai}$,
FIGURE 41
Integral Absolute Error (IAE) vs. kc and taui
Nonlinear Level Control in Conical Tank
Steps in Set Point from 0.9 to 0.6 & from 0.2 to 0.4
Neural Network Model
Global Minimum at kc 4.5, and taui, 1.0
TABLE 10

Combined IAE Non-Linear Tank Network Simulation
Sum of Upper and Lower IAE

Lower Disturbance - Step Up in Set Point (0.2 to 0.4);
   Inlet Flow Rate, 0.5;
   Twenty Time Steps into Future

Upper Disturbance - Step Down in Set Point (0.9 to 0.6);
   Inlet Flow Rate, 0.5;
   Twenty Time Steps into Future

Global Minimum - $k_c$, 4.5 and $\tau_i$, 1

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>0.5</td>
<td>8.834</td>
<td>8.833</td>
<td>8.278</td>
<td>7.251</td>
<td>6.399</td>
<td>6.978</td>
<td>7.303</td>
<td>7.468</td>
<td>7.642</td>
<td>7.681</td>
</tr>
<tr>
<td>1.5</td>
<td>3.126</td>
<td>4.043</td>
<td>4.624</td>
<td>4.801</td>
<td>4.922</td>
<td>5.125</td>
<td>5.158</td>
<td>5.150</td>
<td>5.126</td>
<td>5.077</td>
</tr>
<tr>
<td>2.5</td>
<td>1.737</td>
<td>1.775</td>
<td>1.950</td>
<td>2.084</td>
<td>2.166</td>
<td>2.247</td>
<td>2.285</td>
<td>2.327</td>
<td>2.343</td>
<td>2.368</td>
</tr>
<tr>
<td>3</td>
<td>1.469</td>
<td>1.500</td>
<td>1.644</td>
<td>1.775</td>
<td>1.850</td>
<td>1.901</td>
<td>1.947</td>
<td>1.990</td>
<td>2.034</td>
<td>2.090</td>
</tr>
<tr>
<td>3.5</td>
<td>1.382</td>
<td>1.383</td>
<td>1.513</td>
<td>1.635</td>
<td>1.711</td>
<td>1.806</td>
<td>1.895</td>
<td>1.983</td>
<td>2.088</td>
<td>2.186</td>
</tr>
<tr>
<td>4</td>
<td>1.422</td>
<td>1.319</td>
<td>1.452</td>
<td>1.593</td>
<td>1.735</td>
<td>1.878</td>
<td>2.019</td>
<td>2.166</td>
<td>2.329</td>
<td>2.466</td>
</tr>
<tr>
<td>4.5</td>
<td>1.507</td>
<td>1.301</td>
<td>1.465</td>
<td>1.624</td>
<td>1.813</td>
<td>1.992</td>
<td>2.175</td>
<td>2.369</td>
<td>2.559</td>
<td>2.727</td>
</tr>
<tr>
<td>5</td>
<td>1.681</td>
<td>1.321</td>
<td>1.496</td>
<td>1.682</td>
<td>1.895</td>
<td>2.102</td>
<td>2.325</td>
<td>2.549</td>
<td>2.746</td>
<td>2.940</td>
</tr>
<tr>
<td>5.5</td>
<td>1.971</td>
<td>1.383</td>
<td>1.556</td>
<td>1.739</td>
<td>1.976</td>
<td>2.205</td>
<td>2.452</td>
<td>2.688</td>
<td>2.896</td>
<td>3.084</td>
</tr>
<tr>
<td>6</td>
<td>2.051</td>
<td>1.460</td>
<td>1.622</td>
<td>1.815</td>
<td>2.065</td>
<td>2.315</td>
<td>2.559</td>
<td>2.802</td>
<td>3.021</td>
<td>3.214</td>
</tr>
<tr>
<td>6.5</td>
<td>2.255</td>
<td>1.518</td>
<td>1.721</td>
<td>1.929</td>
<td>2.166</td>
<td>2.426</td>
<td>2.670</td>
<td>2.901</td>
<td>3.120</td>
<td>3.325</td>
</tr>
<tr>
<td>7.5</td>
<td>2.348</td>
<td>1.862</td>
<td>2.091</td>
<td>2.256</td>
<td>2.485</td>
<td>2.705</td>
<td>2.922</td>
<td>3.151</td>
<td>3.345</td>
<td>3.524</td>
</tr>
</tbody>
</table>
1.0. The tuning parameters are changed from the initial $k_C$, 1.5 and $\tau_i$, 2.0 to these new settings. Disturbances used in network training are rerun, and the data screened for new training vectors. These new vectors are added to the existing training set, and the network is retrained. The retrained network is used in simulation to generate new grids of IAE data for the upper and lower disturbances.

The new surface plot for the upper disturbance is shown in Figure 42, and the accompanying data is listed in Table 11. When this surface is compared to that for the DDE model, one can notice a fictitious local minimum develops at low $k_C$. This modeling error is not important, since the values in the local minimum are still too high for the region to be considered as an operating zone. The rest of the surface appears to have developed features closer to that of the DDE model. This observation is born out by the change of the predicted global minimal $k_C$ from 4.0 to 6.0, while the minimal $\tau_i$ remains at the previous value of 0.5. These new tuning parameters exactly correspond to those of the global minimum values of the DDE model. This is the desired goal of the second identification. The success indicates that the additional dynamical information from a retuned system enhances the
FIGURE 42
IAE vs. kc and tau_i (Upper Disturbance)
Nonlinear Level Control in Conical Tank
Step in Set Point (0.9 to 0.6); Inlet Flow Rate, 0.5
Neural Network Model - Second Identification
Global Minimum at kc, 6.0 and tau_i, 0.5
TABLE 11

IAE Non-Linear Tank Network Simulation

1st Identification - $k_C$, 1.5 and $\tau_u$, 2.0
2nd Identification - $k_C$, 4.5 and $\tau_u$, 1.0

Upper Disturbance - Step Down in Set Point (0.9 to 0.6);
Inlet Flow Rate, 0.5;
Twenty Steps Into Future

Global Minimum at $k_C$, 6.0 and $\tau_u$, 0.5

<table>
<thead>
<tr>
<th>$k_C$</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>0.5</td>
<td>5.375</td>
<td>4.675</td>
<td>3.816</td>
<td>2.738</td>
<td>1.969</td>
<td>1.916</td>
<td>2.359</td>
<td>2.587</td>
<td>2.774</td>
<td>2.892</td>
</tr>
<tr>
<td>2</td>
<td>3.613</td>
<td>3.092</td>
<td>2.518</td>
<td>2.135</td>
<td>1.863</td>
<td>1.667</td>
<td>1.449</td>
<td>1.326</td>
<td>1.229</td>
<td>1.164</td>
</tr>
<tr>
<td>2.5</td>
<td>2.321</td>
<td>1.567</td>
<td>1.320</td>
<td>1.250</td>
<td>1.209</td>
<td>1.172</td>
<td>1.212</td>
<td>1.308</td>
<td>1.450</td>
<td>1.592</td>
</tr>
<tr>
<td>3</td>
<td>1.803</td>
<td>1.038</td>
<td>1.054</td>
<td>1.132</td>
<td>1.185</td>
<td>1.270</td>
<td>1.389</td>
<td>1.558</td>
<td>1.735</td>
<td>1.867</td>
</tr>
<tr>
<td>3.5</td>
<td>1.326</td>
<td>0.864</td>
<td>1.010</td>
<td>1.120</td>
<td>1.231</td>
<td>1.364</td>
<td>1.537</td>
<td>1.722</td>
<td>1.889</td>
<td>2.030</td>
</tr>
<tr>
<td>4</td>
<td>1.025</td>
<td>0.820</td>
<td>0.999</td>
<td>1.137</td>
<td>1.280</td>
<td>1.441</td>
<td>1.631</td>
<td>1.823</td>
<td>1.989</td>
<td>2.129</td>
</tr>
<tr>
<td>4.5</td>
<td>0.839</td>
<td>0.810</td>
<td>0.992</td>
<td>1.155</td>
<td>1.317</td>
<td>1.504</td>
<td>1.708</td>
<td>1.897</td>
<td>2.058</td>
<td>2.200</td>
</tr>
<tr>
<td>5</td>
<td>0.738</td>
<td>0.808</td>
<td>0.995</td>
<td>1.171</td>
<td>1.352</td>
<td>1.555</td>
<td>1.765</td>
<td>1.949</td>
<td>2.113</td>
<td>2.256</td>
</tr>
<tr>
<td>5.5</td>
<td>0.683</td>
<td>0.807</td>
<td>1.002</td>
<td>1.189</td>
<td>1.381</td>
<td>1.594</td>
<td>1.802</td>
<td>1.997</td>
<td>2.157</td>
<td>2.304</td>
</tr>
<tr>
<td>6</td>
<td>0.664</td>
<td>0.808</td>
<td>1.012</td>
<td>1.206</td>
<td>1.405</td>
<td>1.629</td>
<td>1.838</td>
<td>2.026</td>
<td>2.194</td>
<td>2.337</td>
</tr>
<tr>
<td>6.5</td>
<td>0.691</td>
<td>0.806</td>
<td>1.018</td>
<td>1.219</td>
<td>1.428</td>
<td>1.655</td>
<td>1.867</td>
<td>2.054</td>
<td>2.217</td>
<td>2.367</td>
</tr>
<tr>
<td>7</td>
<td>0.801</td>
<td>0.806</td>
<td>1.026</td>
<td>1.237</td>
<td>1.446</td>
<td>1.679</td>
<td>1.887</td>
<td>2.079</td>
<td>2.243</td>
<td>2.391</td>
</tr>
<tr>
<td>7.5</td>
<td>0.879</td>
<td>0.812</td>
<td>1.044</td>
<td>1.254</td>
<td>1.469</td>
<td>1.695</td>
<td>1.906</td>
<td>2.092</td>
<td>2.259</td>
<td>2.409</td>
</tr>
<tr>
<td>8</td>
<td>1.018</td>
<td>0.817</td>
<td>1.084</td>
<td>1.283</td>
<td>1.493</td>
<td>1.704</td>
<td>1.912</td>
<td>2.103</td>
<td>2.265</td>
<td>2.416</td>
</tr>
<tr>
<td>8.5</td>
<td>1.072</td>
<td>0.852</td>
<td>1.166</td>
<td>1.358</td>
<td>1.536</td>
<td>1.727</td>
<td>1.915</td>
<td>2.099</td>
<td>2.267</td>
<td>2.413</td>
</tr>
<tr>
<td>9</td>
<td>1.356</td>
<td>0.913</td>
<td>1.275</td>
<td>1.479</td>
<td>1.631</td>
<td>1.811</td>
<td>1.991</td>
<td>2.181</td>
<td>2.358</td>
<td>2.507</td>
</tr>
<tr>
<td>9.5</td>
<td>1.617</td>
<td>1.016</td>
<td>1.376</td>
<td>1.600</td>
<td>1.767</td>
<td>1.970</td>
<td>2.190</td>
<td>2.377</td>
<td>2.560</td>
<td>2.706</td>
</tr>
<tr>
<td>10</td>
<td>1.770</td>
<td>1.119</td>
<td>1.397</td>
<td>1.687</td>
<td>1.905</td>
<td>2.149</td>
<td>2.387</td>
<td>2.591</td>
<td>2.787</td>
<td>2.900</td>
</tr>
<tr>
<td>10.5</td>
<td>1.962</td>
<td>1.192</td>
<td>1.490</td>
<td>1.767</td>
<td>2.053</td>
<td>2.351</td>
<td>2.633</td>
<td>2.845</td>
<td>3.020</td>
<td>3.138</td>
</tr>
</tbody>
</table>
predictive capacities of the network model for the upper disturbance.

The same analysis is conducted for the lower disturbance, and the results are shown in a surface plot in Figure 43. The respective values are given in Table 12. A comparison of this surface with that of the DDE model reveals that this model is less representative in the low $k_C$ region, but is more accurate for higher values of $k_C$. The predicted global minimal $k_C$ also more accurate: it moves from $k_C$, 4 to 8, and $\tau_u$ remains at 1.0. The optimum predicted by the DDE model was $k_C$, 7.5 and $\tau_u$, 0.5. The new $k_C$ is high by only 7%, whereas before it was low by 47%. The actual IAE global minimum is 0.419 and that for this new tuning is 0.490 or higher by 17%. Before the second identification, the difference was 22%. From the results, it is concluded that in the case of the lower disturbance, the second identification has significantly improved the accuracy of predicting the global minimum.
FIGURE 43
IAE vs. kc and tau (Lower Disturbance)
Nonlinear Level Control in Conical Tank
Step in Set Point (0.2 to 0.4); Inlet Flow Rate, 0.5
Neural Network Model - Second Identification
Global Minimum at kc, 8.0 and tau, 1.0
### TABLE 12

**IAE Non-Linear Tank Network Simulation**

1st Identification - $k_C$, 1.5 and $\tau_i$, 2.0  
2nd Identification - $k_C$, 4.5 and $\tau_i$, 1.0  

Lower Disturbance - Step Up in Set Point from 0.2 to 0.4;  
Inlet Flow Rate, 0.5;  
Twenty Steps Into Future

Global Minimum at $k_C$, 8.0 and $\tau_i$, 1.0

<table>
<thead>
<tr>
<th>$k_C$</th>
<th>0.5</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>3.5</th>
<th>4</th>
<th>4.5</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>0.5</td>
<td>3.598</td>
<td>3.879</td>
<td>3.385</td>
<td>2.621</td>
<td>1.889</td>
<td>1.680</td>
<td>1.989</td>
<td>2.212</td>
<td>2.356</td>
<td>2.548</td>
</tr>
<tr>
<td>1.5</td>
<td>2.144</td>
<td>2.198</td>
<td>2.199</td>
<td>2.307</td>
<td>2.484</td>
<td>2.592</td>
<td>2.649</td>
<td>2.656</td>
<td>2.654</td>
<td>2.636</td>
</tr>
<tr>
<td>2</td>
<td>1.792</td>
<td>1.836</td>
<td>1.788</td>
<td>1.781</td>
<td>1.816</td>
<td>1.903</td>
<td>2.011</td>
<td>2.078</td>
<td>2.112</td>
<td>2.114</td>
</tr>
<tr>
<td>2.5</td>
<td>1.259</td>
<td>1.381</td>
<td>1.401</td>
<td>1.350</td>
<td>1.341</td>
<td>1.373</td>
<td>1.423</td>
<td>1.480</td>
<td>1.515</td>
<td>1.540</td>
</tr>
<tr>
<td>3</td>
<td>1.073</td>
<td>1.071</td>
<td>1.102</td>
<td>1.067</td>
<td>1.036</td>
<td>1.021</td>
<td>1.023</td>
<td>1.036</td>
<td>1.028</td>
<td>1.007</td>
</tr>
<tr>
<td>3.5</td>
<td>0.953</td>
<td>0.923</td>
<td>0.898</td>
<td>0.878</td>
<td>0.844</td>
<td>0.825</td>
<td>0.806</td>
<td>0.782</td>
<td>0.754</td>
<td>0.742</td>
</tr>
<tr>
<td>4</td>
<td>0.849</td>
<td>0.811</td>
<td>0.776</td>
<td>0.748</td>
<td>0.736</td>
<td>0.711</td>
<td>0.709</td>
<td>0.726</td>
<td>0.799</td>
<td>0.880</td>
</tr>
<tr>
<td>4.5</td>
<td>0.768</td>
<td>0.731</td>
<td>0.689</td>
<td>0.682</td>
<td>0.672</td>
<td>0.689</td>
<td>0.778</td>
<td>0.857</td>
<td>0.949</td>
<td>1.028</td>
</tr>
<tr>
<td>5</td>
<td>0.715</td>
<td>0.668</td>
<td>0.640</td>
<td>0.635</td>
<td>0.668</td>
<td>0.766</td>
<td>0.869</td>
<td>0.963</td>
<td>1.059</td>
<td>1.137</td>
</tr>
<tr>
<td>5.5</td>
<td>0.674</td>
<td>0.630</td>
<td>0.609</td>
<td>0.624</td>
<td>0.718</td>
<td>0.824</td>
<td>0.930</td>
<td>1.034</td>
<td>1.133</td>
<td>1.229</td>
</tr>
<tr>
<td>6</td>
<td>0.650</td>
<td>0.610</td>
<td>0.599</td>
<td>0.647</td>
<td>0.758</td>
<td>0.872</td>
<td>0.984</td>
<td>1.089</td>
<td>1.193</td>
<td>1.289</td>
</tr>
<tr>
<td>6.5</td>
<td>0.622</td>
<td>0.591</td>
<td>0.598</td>
<td>0.675</td>
<td>0.792</td>
<td>0.911</td>
<td>1.028</td>
<td>1.137</td>
<td>1.243</td>
<td>1.334</td>
</tr>
<tr>
<td>7</td>
<td>0.604</td>
<td>0.576</td>
<td>0.606</td>
<td>0.697</td>
<td>0.823</td>
<td>0.944</td>
<td>1.063</td>
<td>1.176</td>
<td>1.278</td>
<td>1.377</td>
</tr>
<tr>
<td>7.5</td>
<td>0.594</td>
<td>0.566</td>
<td>0.652</td>
<td>0.718</td>
<td>0.847</td>
<td>0.973</td>
<td>1.093</td>
<td>1.207</td>
<td>1.319</td>
<td>1.414</td>
</tr>
<tr>
<td>8</td>
<td>0.597</td>
<td>0.560</td>
<td>0.634</td>
<td>0.735</td>
<td>0.866</td>
<td>0.996</td>
<td>1.121</td>
<td>1.237</td>
<td>1.363</td>
<td>1.444</td>
</tr>
<tr>
<td>8.5</td>
<td>0.614</td>
<td>0.561</td>
<td>0.646</td>
<td>0.750</td>
<td>0.887</td>
<td>1.018</td>
<td>1.143</td>
<td>1.259</td>
<td>1.372</td>
<td>1.471</td>
</tr>
<tr>
<td>9</td>
<td>0.659</td>
<td>0.571</td>
<td>0.656</td>
<td>0.764</td>
<td>0.903</td>
<td>1.037</td>
<td>1.164</td>
<td>1.283</td>
<td>1.393</td>
<td>1.494</td>
</tr>
<tr>
<td>9.5</td>
<td>0.729</td>
<td>0.582</td>
<td>0.665</td>
<td>0.776</td>
<td>0.916</td>
<td>1.051</td>
<td>1.182</td>
<td>1.301</td>
<td>1.410</td>
<td>1.512</td>
</tr>
<tr>
<td>10</td>
<td>0.799</td>
<td>0.602</td>
<td>0.673</td>
<td>0.788</td>
<td>0.930</td>
<td>1.066</td>
<td>1.196</td>
<td>1.317</td>
<td>1.429</td>
<td>1.529</td>
</tr>
<tr>
<td>10.5</td>
<td>0.829</td>
<td>0.610</td>
<td>0.680</td>
<td>0.798</td>
<td>0.941</td>
<td>1.082</td>
<td>1.210</td>
<td>1.333</td>
<td>1.443</td>
<td>1.548</td>
</tr>
</tbody>
</table>
PART 2. II. B. 2. b. 2. Disturbance Tests

The new tuning parameters from the second identification are implemented in the upper and lower portions of the tank respectively. The performance of the new tuning parameters are tested with several disturbances. The disturbances chosen were neither used in the training nor tuning process. The first disturbance is in the lower tank and is a step down in inlet flow rate from 0.7 to 0.3 with the set point constant at 0.23. The response curves for various tunings are shown in Figure 44, namely the initial tuning \((k_C, 1.5 \text{ and } \tau_i, 2.0)\) and the optimum tuning predicted by the network model \((k_C, 8 \text{ and } \tau_i, 1)\) and the tuning predicted by the actual or DDE model \((k_C, 7.5 \text{ and } \tau_i, 0.5)\). Note that the network and actual optimal responses for both liquid level and outlet flow rate are close. Both have significant improvements in performance from that of the initial tuning. The departure of the liquid level curve from the set point is small, and so is the outlet flow rate departure from its steady state value.

An additional test is run for a step up in inlet flow rate from 0.2 to 0.65 in the upper tank with constant set point at 0.65. The result is shown in Figure 45. Again,
FIGURE 44
Comparison of Responses For Different Tunings
Step Down in Inlet in Lower Tank
Simulated Nonlinear System

![Diagram showing normalized liquid level and normalized flow rate over sample time with different tuning parameters.]
FIGURE 45
Comparison of Responses For Different Tunings
Step Up in Inlet Flow Rate in Upper Tank
Simulated Nonlinear System

Normalized Liquid Level

- Set Point
- Initial Tuning (kc, 1.5 & tau, 2.0)
- Network & Actual Optimum (kc, 6 & tau, 0.5)

Normalized Flow Rate

- Inlet
- Outlet (Initial Tuning)
- Outlet (Network & Actual Optimum Tuning)

Sample Time
the improvement in liquid level and outlet flow rate responses are excellent.

Besides improving responses to steps in inlet flow rate, the new tuning parameters also have positive effects on the responses to steps in set point. The first set point disturbance is a step down in set point from 0.47 to 0.27 with the inlet flow rate constant at 0.64 (Figure 46). The liquid level for the optimal tuning approaches the new set point at the fastest rate with virtually no undershoot, whereas for the initial tuning is somewhat slower to respond with prominent undershoot. The outlet flow rate behavior for the optimal curves is more of the "bang-bang" or "on/off" than that of the initial tuning.

The same simulation experiment is repeated for a second step in set point from 0.6 to 0.8 in the upper tank with constant set point at 0.43. The result is shown in Figure 47. Note that the liquid level for the optimal tuning is slightly oscillatory. This causes an oscillatory behavior in the outlet flow rate response.

The oscillatory behavior indicates that choosing the operating point exactly at the global minimum for IAE may not be beneficial in terms of smooth controller action. This is demonstrated by a test on the sensitivity of the response to a slight change in $k_c$ about the point of
FIGURE 46
Comparison of Responses For Different Tunings
Step Down in Set Point in Lower Tank
Simulated Nonlinear System

Normalized Liquid Level

- - - -  Set Point
- - - -  Initial Tuning (kc, 1.5 & tau, 2.0)
- - - -  Network Optimum Tuning (kc, 8 & tau, 1)
△  Actual Optimum Tuning (kc, 7.5 & tau, 0.5)

Normalized Flow Rate

- - - -  Inlet
- - - -  Outlet (Initial Tuning)
- - - -  Outlet (Network Optimum Tuning)
△  Outlet (Actual Optimum Tuning)

Sample Time

0  5  10  15  20  25  30  35
FIGURE 47
Comparison of Responses For Different Tunings
Step Up in Set Point in Upper Tank
Simulated Nonlinear System

Normalized Liquid Level

- Set Point
- Initial Tuning (kc, 1.5 & tau, 2.0)
- Network & Actual Optimum (kc, 6 & tau, 0.5)

Normalized Flow Rate

- Inlet
- Outlet (Initial Tuning)
- Outlet (Network & Actual Optimum Tuning)

Sample Time
global minimum IAE. The result is shown in Figure 48. Lowering of $k_c$ from 6 to 5 smooths the slightly oscillatory behavior of the liquid level and dramatically calms the outlet flow rate behavior. Raising $k_c$ from 6 to 6.5 has a moderately adverse effect on the liquid level control in that it becomes more oscillatory. This in turn causes the outlet flow rate response to enter a pronounced on/off cycle. This analysis indicates that the response of the outlet flow rate may be very unstable in certain tuning directions near the global minimum IAE. For this reason, IAE may not give a complete picture of tuning objective. The behavior of the manipulated variable should also be taken into account. A more complete objective is to simultaneously achieve adherence to the set point and smooth controller action. This issue will be addressed further in the experimental section.
FIGURE 48
Sensitivity of Response Near IAE Global Minimum Tuning
Step Up in Set Point in Upper Tank
Simulated Nonlinear System

Global Minimum at $k_c$, 6.0 and $\tau_i$, 0.5

- Set Point
- $k_c$, 5.0 and $\tau_i$, 0.5
- $k_c$, 6.5 and $\tau_i$, 0.5

Normalized Liquid Level

Normalized Flow Rate

Sample Time
PART 2. II. B. 2. b. 3. Conclusion and Summary

In conclusion, a model can be constructed with closed-loop data from an initial tuning. If this model is used to discover improved tuning parameters and these new parameters implemented, a second identification will further improve the mapping provided the data is consistent. Consistency in this context means that the physical system for the first identification is same as that for the second. If the data is not consistent then the mapping may be degraded by a second identification. In the example involving a second identification, the system data was consistent, and the predictive power of the network for this nonlinear system was enhanced by the incorporation of the dynamics of the new tuning. The response curves that resulted from implementation of the network based optimal tuning parameters were dramatic improvements over the initial tunings, however some consideration of the smoothness of control is warranted.

In summary, closed-loop data from a simulated nonlinear system were used to construct a one-step-ahead network model. This model was employed in a recursive fashion to simulate the responses of two disturbances, an upper tank step in set point and a lower one. The IAE values for these disturbances were calculated for a grid
of $k_C$ and $\tau_i$'s. From this data, the tuning parameters which correspond to a global minimum IAE were determined. The predictions of behavior of IAE were adequate. Additional accuracy was gained by implementing the predicted optimal tuning parameters, then repeating the identification. The preliminary conclusion is that the network identification of this nonlinear system is sufficiently accurate to capture the behavior of IAE so that one obtains a more complete picture of available tuning options and is able to fine tune the controller successfully.
PART 2. II. C. Experimental Systems

The techniques established in tuning the simulated systems are applied to two bench scale level control experiments. These experiments are intended to assess the impact of factors such as delay, noise and unmodeled process dynamics (e.g. pumps, computer, differential pressure cell, etc.) on the methodology developed so far.

The experimental set up is intended to mimic that of the simulation in that the tank has an uncontrolled inlet flow rate and a controlled outlet. The system can be changed from linear to nonlinear by replacing the cylindrical tank with a conical one. A schematic of the experimental arrangement is shown in Figure 49. A tank is fed by a peristaltic pump from the top, and the level in the tank is controlled by adjusting the flow rate of an outlet peristaltic pump which is connected near the bottom of the tank. The pressure at the bottom of the water column is measured by a differential pressure (DP) cell which is connected at the tank bottom. Specific equipment parameters are listed in Table 13.

The liquid level in the tank is controlled by a digital computer. The level is determined through measurements from the DP cell. The normalizing level is the "declared" top of the tank and the outlet pump
Figure 49
Schematic of Experiment

DP: Differential Pressure
p1: Pump #1
p2: Pump #2
ST: Storage Tank
### Table 13.
EXPERIMENT EQUIPMENT SPECIFICATIONS

**Pumps**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pump Type</td>
<td>Peristaltic</td>
</tr>
<tr>
<td>Inlet Pump Maximum Flow Rate, ml/sec</td>
<td>2.5</td>
</tr>
<tr>
<td>Outlet Pump Maximum Flow Rate, ml/sec</td>
<td>3.0</td>
</tr>
<tr>
<td>Flow rate vs input voltage</td>
<td>linear</td>
</tr>
</tbody>
</table>

**Linear Tank**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diameter, inches</td>
<td>1.5</td>
</tr>
<tr>
<td>Total Height, inches</td>
<td>30</td>
</tr>
<tr>
<td>Declared Tank Bottom, inches</td>
<td>3</td>
</tr>
<tr>
<td>Declared Tank Top, inches</td>
<td>24</td>
</tr>
<tr>
<td>Sampling Period, seconds</td>
<td>2</td>
</tr>
</tbody>
</table>

**Nonlinear Tank**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top Diameter, inches</td>
<td>2</td>
</tr>
<tr>
<td>Bottom Diameter, inches</td>
<td>0.5</td>
</tr>
<tr>
<td>Total Height, inches</td>
<td>35</td>
</tr>
<tr>
<td>Sampling period, seconds</td>
<td>3</td>
</tr>
</tbody>
</table>

**Differential Pressure Cell**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage vs. liquid level</td>
<td>linear</td>
</tr>
</tbody>
</table>
connection is the "declared" bottom. The outlet and inlet flow rates are normalized using the voltages which correspond to the maximum flow rates of each pump respectively. Flow rates are the recorded values and are not the measured values since flow meters are not included in the experiment. Control is accomplished using the velocity form of the PI algorithm (Stephanopoulos, 1984). Both the DP cell and the pumps have linear behavior over most of their range, but near the low and high ranges both have slightly nonlinearities.

Much of the noise in the process appears to be system generated. The suction of the outlet peristaltic pump causes a pulse through the water column as a section of liquid moves through the pump. Since the suction is near the DP cell tap, any pulsation causes a pressure wave which is detected by the DP cell. As a result, the measured level rises slightly even though visual the level is not changing. This measurement error is incorporated into the calculations of the controller response and is amplified by the control algorithm and results in unwarranted action by the outlet pump. In this manner, a small error in level causes a noticeable systematic disturbance. When the outlet flow pump is near its maximum rpm, the noise is estimated to be as large as 5%,
whereas when the rpm is below half its maximum the noise is about 2 to 3%.

PART 2. II. C. 1. Linear Level Control Experiment

The first experiment involves level control of the linear system, a cylindrical tank. The sampling time for this system is two seconds and the delay of approximately one second. The initial controller tuning parameters are selected based on quarter decay at 1.5 and 5.0.

A series of closed loop disturbances are recorded for this system. The disturbances include 18 steps in set point and 11 steps in inlet flow rate and are listed in Table 14. These disturbances are used in all future experimental examples to train the network. The closed loop data is submitted to the network identification process that was described previously in Part 1 VII. Network vectors are constructed for each sample time and screened to form roughly 15,000 candidate set of vectors. Vectors are selected from this set to be members of the training set based on error between the current network prediction for the one-step-ahead behavior and the actual behavior. The number of vectors included in the final training set was 200 which was approximately double that for the noiseless simulation.
### TABLE 14
Training Disturbances
Experimental Systems

<table>
<thead>
<tr>
<th>Disturbance</th>
<th>From</th>
<th>To</th>
<th>Inlet Flow Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step in Set Point</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.50</td>
<td>0.75</td>
<td>0.50</td>
<td></td>
</tr>
<tr>
<td>0.60</td>
<td>0.35</td>
<td>0.40</td>
<td></td>
</tr>
<tr>
<td>0.70</td>
<td>0.80</td>
<td>0.55</td>
<td></td>
</tr>
<tr>
<td>0.85</td>
<td>0.66</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>0.78</td>
<td>0.48</td>
<td>0.25</td>
<td></td>
</tr>
<tr>
<td>0.75</td>
<td>0.30</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>0.83</td>
<td>0.25</td>
<td>0.60</td>
<td></td>
</tr>
<tr>
<td>0.50</td>
<td>0.30</td>
<td>0.72</td>
<td></td>
</tr>
<tr>
<td>0.55</td>
<td>0.22</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>0.58</td>
<td>0.33</td>
<td>0.27</td>
<td></td>
</tr>
<tr>
<td>0.20</td>
<td>0.37</td>
<td>0.30</td>
<td></td>
</tr>
<tr>
<td>0.75</td>
<td>0.83</td>
<td>0.76</td>
<td></td>
</tr>
<tr>
<td>0.87</td>
<td>0.68</td>
<td>0.22</td>
<td></td>
</tr>
<tr>
<td>0.60</td>
<td>0.45</td>
<td>0.50</td>
<td></td>
</tr>
<tr>
<td>0.58</td>
<td>0.48</td>
<td>0.25</td>
<td></td>
</tr>
<tr>
<td>0.34</td>
<td>0.26</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>0.20</td>
<td>0.30</td>
<td>0.60</td>
<td></td>
</tr>
<tr>
<td>0.85</td>
<td>0.28</td>
<td>0.46</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Disturbance</th>
<th>From</th>
<th>To</th>
<th>Inlet Flow Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step in Inlet Flow Rate</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.50</td>
<td>0.75</td>
<td>0.50</td>
<td></td>
</tr>
<tr>
<td>0.55</td>
<td>0.25</td>
<td>0.38</td>
<td></td>
</tr>
<tr>
<td>0.58</td>
<td>0.85</td>
<td>0.65</td>
<td></td>
</tr>
<tr>
<td>0.86</td>
<td>0.48</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>0.78</td>
<td>0.35</td>
<td>0.30</td>
<td></td>
</tr>
<tr>
<td>0.70</td>
<td>0.25</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>0.65</td>
<td>0.15</td>
<td>0.45</td>
<td></td>
</tr>
<tr>
<td>0.40</td>
<td>0.65</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>0.60</td>
<td>0.20</td>
<td>0.80</td>
<td></td>
</tr>
<tr>
<td>0.65</td>
<td>0.30</td>
<td>0.35</td>
<td></td>
</tr>
<tr>
<td>0.20</td>
<td>0.40</td>
<td>0.25</td>
<td></td>
</tr>
</tbody>
</table>
A test set was selected from the candidate set, and the network was trained by the tested train method described earlier. Weights developed during this training session are used in the simulations of responses to disturbances.

A single disturbance is used to determine the behavior of IAE for different tuning parameters. The disturbance chosen is a step up in set point from 0.5 to 0.75 with the inlet flow rate constant at 0.5. The simulation uses the same tuning parameters for which the network was trained, namely $k_c$, 1.5 and $\tau_1$ at 5.0. A comparison of the experimental and simulated responses for the disturbance is shown in Figure 50. Notice that the network liquid level response is similar to that of the experiment, but they do not match exactly because of modeling error. The network response is smoother than the experiment. The network learning algorithm tends to smooth data on a mean-squares basis. The responses of the outlet flow rate are coincident until the cumulative errors created by the recursive nature of network simulation have their effect in the latter sample times.

The next task is to determine the IAE of this disturbance for a grid of tuning parameter settings with $k_c$ ranging from 0.5 to 25 and $\tau_1$ ranging from 0.5 to 10.
FIGURE 50
Experimental vs. Simulated Response
Step Up In Set Point
Linear Level Control Experiment

Initial Tuning Parameters (kc, 1.5 and tau 5.0)
This is accomplished both experimentally and by network simulation.

Experiments are conducted for the grid of tuning parameters, and the IAE for the disturbance is determined at each point. The results are shown as a surface plot of IAE versus $k_c$ and $\tau_i$ in Figure 51, and the specific IAE values of the grid are given in Table 15. From the plot, one can clearly detect regions of operation to be avoided and those which look attractive. The most important region to avoid is that where $k_c$ is low and a secondary area is where $\tau_i$ is low.

The region of operation which looks promising is the range where $k_c$ and $\tau_i$ have medium to high values. The most interesting area within this region is where the IAE is at a minimum for the grid, the global minimum. From the table, this minimum occurs at a value of 2.85 which corresponds to $k_c$ at 10.5 and $\tau_i$ at 5.0. Note from Figure 51, that the surface is very flat at this location, so that nearby points have about the same IAE value. The noise level for IAE values is determined to be between plus or minus 2%. This indicates that the global minimum should not be considered as a point but an area consisting of any point that within 2% of the minimum. Any point in this area will have tuning parameters that when
FIGURE 51
Integral Absolute Error (IAE) vs. kc and tau1
Linear Level Control in Cylindrical Tank
Experimental Results
Step in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
Global Minimum at kc, 10.5 and tau1, 5.0
### TABLE 15

**Experimental IAE Values for Linear Level Control**  
*Disturbance - Step in Set Point from 0.5 to 0.75*  
*Inlet Flow Rate at 0.5*

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>0.5</th>
<th>1.5</th>
<th>5</th>
<th>7</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>5.693</td>
<td>6.813</td>
<td>10.283</td>
<td>12.073</td>
<td>10.339</td>
</tr>
<tr>
<td>1.5</td>
<td>4.981</td>
<td>4.315</td>
<td>5.693</td>
<td>5.378</td>
<td>5.613</td>
</tr>
<tr>
<td>4.5</td>
<td>3.684</td>
<td>3.058</td>
<td>3.207</td>
<td>3.161</td>
<td>3.331</td>
</tr>
<tr>
<td>9</td>
<td>3.717</td>
<td>3.346</td>
<td>2.949</td>
<td>2.905</td>
<td>2.995</td>
</tr>
<tr>
<td>10.5</td>
<td>3.549</td>
<td>3.311</td>
<td>2.862</td>
<td>2.958</td>
<td>2.991</td>
</tr>
<tr>
<td>13</td>
<td>3.385</td>
<td>3.237</td>
<td>2.979</td>
<td>2.983</td>
<td>3.001</td>
</tr>
<tr>
<td>15</td>
<td>3.578</td>
<td>3.113</td>
<td>3.006</td>
<td>3.041</td>
<td>3.085</td>
</tr>
<tr>
<td>20</td>
<td>3.256</td>
<td>3.224</td>
<td>3.089</td>
<td>3.011</td>
<td>3.037</td>
</tr>
<tr>
<td>25</td>
<td>3.269</td>
<td>3.348</td>
<td>3.004</td>
<td>3.047</td>
<td>3.276</td>
</tr>
</tbody>
</table>
implemented yield behavior similar to that of the global minimum.

As an aside, the construction of this type of surface plot for an industrial process is usually not possible. One is not at liberty to vary tuning parameters in an such a fashion. In some cases, this would be dangerous or in others would be simply unprofitable. For this reason, a surface plot of this nature cannot be practically generated from the physical system, and simulation is the only alternative.

The simulated behavior is examined using the network model. The same grid of $k_c$ and $\tau_i$ parameters used in the experimental analysis is employed. The results are presented as a surface plot in Figure 52, and the numerical values are given in Table 16. Notice that the basic shape of this surface is the same as that for the experimental results. The prominent hill and valley arrangements is the same, and the flat region is in the same location. One area that is different is near $k_c$, 4.5 and $\tau_i$, 0.5. A local minimum occurs that does not appear in the experimental surface. This modeling error is of little consequence since the minimum is too high in value to be a candidate operating point. The topology near the global minimum is similar to the experimental case. As with the experiment surface, this simulated
FIGURE 52
Integral Absolute Error (IAE) vs. kc and tau
Linear Level Control in Cylindrical Tank
Network Simulation (Trained with Experimental Data)
Step in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
Global Minimum at kc 15.0 and tau 5.0
### TABLE 16

Simulated IAE Values for Linear Level Control Network Trained With Experimental Data
Disturbance - Step in Set Point from 0.5 to 0.75
Inlet Flow Rate at 0.5

<table>
<thead>
<tr>
<th>$\tau_i$</th>
<th>0.5</th>
<th>1.5</th>
<th>5</th>
<th>7</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.5</td>
<td>1.5</td>
<td>5</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>0.5</td>
<td>4.573</td>
<td>6.078</td>
<td>8.111</td>
<td>8.860</td>
<td>9.358</td>
</tr>
<tr>
<td>1.5</td>
<td>3.693</td>
<td>4.372</td>
<td>5.737</td>
<td>5.925</td>
<td>6.067</td>
</tr>
<tr>
<td>4.5</td>
<td>3.174</td>
<td>3.447</td>
<td>3.845</td>
<td>3.967</td>
<td>4.023</td>
</tr>
<tr>
<td>9</td>
<td>3.933</td>
<td>3.166</td>
<td>3.263</td>
<td>3.284</td>
<td>3.295</td>
</tr>
<tr>
<td>10.5</td>
<td>3.959</td>
<td>3.179</td>
<td>3.202</td>
<td>3.219</td>
<td>3.227</td>
</tr>
<tr>
<td>13</td>
<td>3.918</td>
<td>3.688</td>
<td>3.152</td>
<td>3.162</td>
<td>3.173</td>
</tr>
<tr>
<td>15</td>
<td>3.907</td>
<td>3.855</td>
<td>3.128</td>
<td>3.136</td>
<td>3.160</td>
</tr>
</tbody>
</table>
surface is useful in determining regions of $k_C$ and $\tau_i$ to avoid, namely areas of low $k_C$ and $\tau_i$ where the IAE values are high.

The most important area is near the global minimum IAE which occurs at $k_C$, 15 and $\tau_i$, 5. The value of the global minimum $k_C$ for the experiment was 10.5 so that this prediction is 43% higher. The value of $\tau_i$ is exactly the same as the experimental value. The experimentally determined IAE which corresponds to $k_C$, 15 and $\tau_i$, 5 is 3.01. This is only 5% higher than the experimental global minimum. As was seen in previous computer simulations, the predicted values of $k_C$ and $\tau_i$ may vary significantly from those that correspond to the actual minimum, but the value of IAE for the predicted parameters is still very close to the actual. This means that the IAE reduction that results from implementing the predicted parameters is usually as good as that of the experiment.

The improvement in performance that results in implementing the network recommended tuning parameters is demonstrated by comparing responses to a step up in set point from 0.5 to 0.75 for different tunings. The result is shown in Figure 53 which compares responses of three different tunings: the initial tuning ($k_C$, 1.5 and $\tau_i$, 5.0), the experimentally determined optimal tuning ($k_C$, 10.5 and $\tau_i$, 5.0) and the network predicted optimal
FIGURE 53
Comparison of Responses to Step Up in Set Point
Initial vs Experimental and Network Optimal Tunings
Linear Level Control Experiment

Normalized Liquid Level

Set Point
Initial Tuning
(kc, 1.5 and tau_i, 5.0)
Optimal Experimental Tuning
(kc, 10.5 and tau_i, 5.0)
Optimal Network Tuning
(kc, 15.0 and tau_i, 5.0)

Normalized Flow Rate

Inlet
Outlet (Initial tuning)
Outlet (Network optimum tuning)
Outlet (Experimental optimum tuning)

Sample Time
tuning \((k_C, 15.0 \text{ and } \tau_i, 5.0)\). The behavior of the liquid level curves of the experimental optimal and the network optimal tunings are dramatic improvements over that of the initial tuning. The response of the initial tuning tends to oscillate about the set point with significant overshoot. The network and experimental responses both approach the set point quickly and have negligible overshoot. The two liquid level curves are practically indistinguishable, although the outlet flow rate response curves are different. The outlet flow rate response for the initial tuning is smooth but oscillatory with a low frequency and large amplitude. The responses for the other two cases are noisy, and the oscillations are higher in frequency. The response for the network has the largest amplitude, because its \(k_C\) value is higher.

A further demonstration of the effect of tuning is the improvement in performance which occurs for a load disturbance, namely a step in inlet flow as shown in Figure 54. The scale of the liquid level is expanded to accentuate the differences between the network and experimental curves. The behavior of both the network and experimental optimal curves are again significant improvements over that of the initial tuning. The system returns to its original set point quickly with very small overshoots. The outlet flow rate behavior of both optimum
FIGURE 54
Comparison of Responses to Step Up in Inlet Flow Rate
Initial vs. Experimental and Network Optimal Tunings
Linear Level Control Experiment

Experimental Optimum Tuning
(kc, 10.5 and tau, 5.0)

Network Optimal Tuning
(kc, 15.0 and tau, 5.0)

Set Point

Initial Tuning
(kc, 1.5 and tau, 5.0)

Inlet

Outlet Initial Tuning

Outlet Experimental Optimum Tuning

Outlet Network Optimal Tuning

Normalized Liquid Level

Normalized Flow Rate

Sample Time
curves is again not as low in frequency as the initial tuning.

PART 2. II. C. 1. a.
Controller Variable Squared Error (CVSE)

Note that the response curves in Figures 53 and 54 show that tuning parameters which correspond to the minimum in IAE do not result in the smoothest controller action. In fact, the goals of minimizing IAE while at the same time providing the smoothest controller action are inconsistent. The smoothest controller action is no controller action at all. This tends to maximize IAE. For this reason, a compromise between minimizing IAE and attaining smoother controller action must be made.

The controller action can be characterized by a parameter which sums the square of the difference between the current value of the controller variable and its steady state value. This is called the Controller Variable Squared Error (CVSE). In this parameter, error is squared to penalize movements far from the steady state value. Minimization of this parameter yields the smoothest controller action.

The disturbance used to calculate the CVSE is the same step up in set point from 0.5 to 0.75 which was used
previously to determine IAE. Since the inlet flow rate for the disturbance is constant at 0.5, it equals the steady state outlet flow rate. The steady state value of the outlet flow rate is subtracted from the current value at each sample time, and the difference is squared and summed to obtain the CVSE for the current set of tuning parameters. CVSE is measured for the same grid of tuning parameters used in the IAE calculations. A CVSE grid is determined both experimentally and through network simulations.

A surface plot of the experimentally determined CVSE is shown in Figure 55, and the values are listed in Table 17. The areas of poor CVSE performance occur where the CVSE is high. These occur where $\tau_u$ is low. The regions of better performance occur for medium to low $k_c$ values and high $\tau_u$. The minimum CVSE for this grid occurs at $k_c$, 9 and $\tau_u$, 10, which is at the grid's border. If the grid is expanded, the minimum occurs when $k_c$ becomes zero and $\tau_u$ is infinitely large, which corresponds to no control.

The experimental CVSE behavior is compared to that produced by the network simulation. A surface plot of the simulated CVSE data is shown in Figure 56 and the values are presented in Table 18. From the surface plot, notice that some of the general features of the experimental plot
FIGURE 55
Controller Variable Squared Error (CVSE) vs. kc and taul
Linear Level Control in Cylindrical Tank
Experimental Results
Step Up in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
Global Minimum at kc, 9.0 and taul, 10.0
TABLE 17

Experimental CVSE Values for Linear Level Control Disturbance - Step in Set Point from 0.5 to 0.75
Inlet Flow Rate at 0.5

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>0.5</th>
<th>1.5</th>
<th>5</th>
<th>7</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>0.5</td>
<td>8.191</td>
<td>10.413</td>
<td>9.694</td>
<td>8.604</td>
<td>5.124</td>
</tr>
<tr>
<td>1.5</td>
<td>13.503</td>
<td>8.233</td>
<td>8.191</td>
<td>6.767</td>
<td>6.349</td>
</tr>
<tr>
<td>4.5</td>
<td>11.096</td>
<td>6.397</td>
<td>5.337</td>
<td>5.007</td>
<td>5.010</td>
</tr>
<tr>
<td>9</td>
<td>17.670</td>
<td>8.234</td>
<td>5.098</td>
<td>4.829</td>
<td>4.727</td>
</tr>
<tr>
<td>10.5</td>
<td>14.021</td>
<td>11.001</td>
<td>5.033</td>
<td>4.920</td>
<td>4.919</td>
</tr>
<tr>
<td>13</td>
<td>19.764</td>
<td>11.099</td>
<td>5.464</td>
<td>5.439</td>
<td>5.261</td>
</tr>
<tr>
<td>20</td>
<td>20.554</td>
<td>13.039</td>
<td>8.239</td>
<td>6.736</td>
<td>6.118</td>
</tr>
<tr>
<td>25</td>
<td>22.390</td>
<td>17.195</td>
<td>8.286</td>
<td>7.749</td>
<td>9.331</td>
</tr>
</tbody>
</table>
FIGURE 56
Controller Variable Squared Error (CVSE) vs. kc and tau
Linear Level Control in Cylindrical Tank
Network Simulation (Trained With Experimental Data)
Step Up in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
Global Minimum at kc, 20 and tau, 10
### TABLE 18

Simulated CVSE Values for Linear Level Control Network Trained With Experimental Data

**Disturbance - Step in Set Point from 0.5 to 0.75**

**Inlet Flow Rate at 0.5**

<table>
<thead>
<tr>
<th>$k_C$</th>
<th>0.5</th>
<th>1.5</th>
<th>5</th>
<th>7</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>3.380</td>
<td>3.383</td>
<td>2.928</td>
<td>2.701</td>
<td>2.626</td>
</tr>
<tr>
<td>1.5</td>
<td>2.963</td>
<td>3.246</td>
<td>3.260</td>
<td>3.291</td>
<td>3.193</td>
</tr>
<tr>
<td>4.5</td>
<td>2.541</td>
<td>2.727</td>
<td>2.774</td>
<td>2.712</td>
<td>2.641</td>
</tr>
<tr>
<td>9</td>
<td>3.632</td>
<td>2.526</td>
<td>2.486</td>
<td>2.460</td>
<td>2.403</td>
</tr>
<tr>
<td>10.5</td>
<td>3.734</td>
<td>2.506</td>
<td>2.459</td>
<td>2.425</td>
<td>2.371</td>
</tr>
<tr>
<td>13</td>
<td>3.841</td>
<td>3.332</td>
<td>2.428</td>
<td>2.397</td>
<td>2.330</td>
</tr>
<tr>
<td>15</td>
<td>3.923</td>
<td>3.612</td>
<td>2.408</td>
<td>2.367</td>
<td>2.310</td>
</tr>
<tr>
<td>20</td>
<td>4.027</td>
<td>3.897</td>
<td>2.677</td>
<td>2.353</td>
<td>2.297</td>
</tr>
<tr>
<td>25</td>
<td>3.992</td>
<td>3.924</td>
<td>3.342</td>
<td>3.201</td>
<td>3.039</td>
</tr>
</tbody>
</table>
are present, namely the high hill for high $k_c$ and low $\tau_{ui}$ and the valley for middle range $k_c$ and high $\tau_{ui}$. Differences due to modelling error are the valleys at low $k_c$ and low $\tau_{ui}$ and the one near the edge of the grid for low $k_c$. As it will be seen later, these errors are not problematic because the CVSE trend near the tuning parameters that correspond to global minimum IAE are of principle interest. In the critical region, the CVSE behavior is sufficiently accurate.

It should also be pointed out that the CVSE is more prone to modeling error than the IAE. A small consistent error in the simulated liquid level will produce a large error in the outlet flow rate. For this reason, liquid level modeling errors are amplified in the determination of the outlet flow rate and subsequent determination of CVSE.

PART 2. II. C. 1. b. Total Error (TE)

With both the CVSE and IAE determined, it is possible to combine these two measures to achieve smoother controller action. This is practical in some cases, because a slight increase in the IAE away from the global minimum may decrease the CVSE significantly. One way to combine the two parameters is by a weighed sum which forms
the composite cost function called the Total Error (TE). The two parameters are first normalized by dividing each value in the $k_C$ and $\tau_{ui}$ grid by the global minimum value. This guarantees that the two functions are of the same magnitude close to their respective global minima. The normalized IAE value is multiplied by a positive fraction called the weighting factor (WF), and the corresponding normalized CVSE is multiplied by one minus WF. The normalized and weighted cost functions are added to form the TE value. The equation for TE is given by

$$TE = WF \times IAE_{\text{norm}} + (1 - WF) \times CVSE_{\text{norm}}$$

The minimum TE within the grid of $k_C$ and $\tau_{ui}$ values is an appropriate optimum operating point. The location of this point depends on the value of the weighting factor, WF. If WF is one then the optimization is based solely on IAE, and correspondingly if it is zero then the optimization is based solely on CVSE. Figure 57 is a trajectory of the optimum tuning parameters as a function of WF based on network simulated results. Note that the tuning parameters change via a curved path from the CVSE optimum tuning at $k_C$, 20 and $\tau_{ui}$, 10, the IAE optimum at $k_C$, 15 and $\tau_{ui}$, 5. The user must choose the value of the
FIGURE 57
Optimal Tuning Parameters vs. Weighting Factor (WF)
Total Error, $TE = WF \times IAE + (1 - WF) \times CVSE$
Network Simulation Results
Network Trained on Linear Level Control Experimental Data
weighting factor based on the relative importance of IAE and CVSE. IAE determines the quality of control, namely the speed of response and the overshoot and undershoot and decay ratio. Good IAE behavior means fast response and strong adherence to the set point. CVSE determines the frequency and amplitude of the controller variable action. Good CVSE behavior implies that the controller device is not cycled from on to off in a jerky fashion. Rapid variations in the controller action tends to cause accelerated failure of the controller actuator, in this case the pump. When the quality of control is more important that smooth controller action, the weighting factor will be close to unity.

Suppose that the weighting factor is moved from 1 to 0.7. When experimental data is used to calculate TE, the tuning parameters change from \( k_C \), 10.5 and \( \tau_\text{d} \), 5 to \( k_C \), 9 and \( \tau_\text{d} \), 7. The IAE increases from 2.862 to 2.905 or 1.5 %, and the CVSE drops from 5.033 to 4.83 or 4.0 %. When the network simulation is used to determine TE, the tuning parameters change from \( k_C \), 15 and \( \tau_\text{d} \), 5 to \( k_C \), 15 and \( \tau_\text{d} \) 10. The simulated IAE increases from 3.128 to 3.160 or 1.0 % and the simulated CVSE drops from 2.408 to 2.310 or 4.1 %. If the network results are implemented then the experimental data shows that the true IAE
increases from 3.006 to 3.009 or 0.1\% and the CVSE drops from 6.367 to 5.525 or 13 \%.

In both experimental and simulated cases, a small increase in IAE yields a large decrease in CVSE. In addition, even though the global minimum IAE is different for the experimental and simulated cases, the shape of the IAE and CVSE surfaces are similar, so that the TE parameter is useful in smoothing the controller action while at the same time maintaining good controller performance. These results are shown graphically in Figure 58 which presents a comparison of liquid level responses for different tunings. The first two are for experimental data. The top most graph corresponds to the tuning parameters which were selected using only IAE as the basis. In this case, the weighting factor, WF, is one. The second graph uses both IAE and CVSE in the TE formulation with the weighting factor at 0.7. The performance for the two different tunings appears to be the same. Similar results are obtained in the third and fourth graphs which show the behavior using tunings derived from network simulation. These graphs indicate that the performance does not degrade significantly when TE is used for tuning and the weighting factor is kept high.
FIGURE 5B
Comparison of Liquid Level Responses for Different Tunings
Linear Level Control Experiment
Tuning Using Integral Absolute Error, IAE (Experiment)

Tuning Using TE (Experiment)

Controller Variable Squared Error (CVSE)
Weighting Factor (WF), 0.7
TE = WF * IAE + (1 - WF) * CVSE

Tuning Using IAE (Network Simulation)

Tuning Using TE (Network Simulation)

Set Point
Liquid Level
WF, 0.7

Normalized Level

Normalized Level

Normalized Level

Normalized Level

Sample Time
In Figure 59, the responses of the outlet flow rate are compared. Again the first two graphs are experimental results. A slight smoothing of the noisy behavior can be detected between the IAE and TE cases. The smoothed behavior is more noticeable in the bottom two graphs which are for network simulation.

To summarize the linear experiment, the closed-loop network model is accurate enough to capture both the IAE and CVSE behavior of the real system. This model is used in simulation to determine the tuning parameters which correspond to improve IAE behavior while at the same time attaining smoother control action through the use of CVSE.

PART 2. II. C. 2. Nonlinear Level Control Experiment

In the previous experiment, closed-loop identification with the BP network was used to successfully construct a model that was sufficiently accurate to compute the correct behavior for IAE and CVSE. This success leads to the second experiment which considers a more complicated system. The previous system was linear and a simple integrator. The new system is nonlinear and is not a simple integrator, in that the cylindrical tank is replaced with one that is conically shaped. The
FIGURE 59
Comparison of Outlet Flow Rate Responses for Different Tunings
Linear Level Control Experiment
Tuning Using IAE (Experiment)

\[ kc, 10.5 \text{ and } \tau_{ai}, 5 \]

Tuning Using TE (Experiment)

\[ kc, 9 \text{ and } \tau_{ai}, 7 \]

Controller Variable Squared Error (CVSE)
Weighting Factor (WF), 0.7
\[ TE = WF \times IAE + (1 - WF) \times CVSE \]

Tuning Using IAE (Network Simulation)

\[ kc, 15 \text{ and } \tau_{ai}, 5 \]

Tuning Using TE (Network Simulation)

\[ kc, 15 \text{ and } \tau_{ai}, 10 \]

Inlet
Outlet
WF, 0.7
smallest diameter of the cone is at the tank bottom. For the new system, the response to a load or set point disturbance in the top of the tank is much slower than that in the bottom. For the same difference between inlet flow rate and outlet flow rate, the change in liquid level is now a function of the reciprocal of the square of the radius. It can be shown that when this system is linearized, it is at least first order near the expansion point. This means that the integrating system now has some of the qualities a first order system and probably possesses elements of higher order.

The identification process is the same as described previously. The series of disturbances, 18 steps in set point and 12 steps in inlet flow rate which were presented previously in Table 14 are repeated for this system. Data from these disturbances are once again organized into vectors and screened to construct a training set for the neural network. Once the set is constructed, the network is fully trained using the tested training method, and the weights that result are used in the simulation process.

The simulation in the linear case entailed a single disturbance to calculate IAE and CVSE. However, in the nonlinear case, at least two disturbances are included to account for the nonlinear dependency of the system behavior on liquid level. Two steps in set point are
selected: the first is in the upper portion of the tank with the set point changing from 0.5 to 0.75 and the inlet flow rate constant at 0.5; and the other in the lower portion of the tank with the set point changing from 0.2 to 0.37 and the inlet constant at 0.3. Since the system behavior is different in the upper and lower portions of the tank, one can expect that the optimum tuning parameters will also be different in the two regions. Thus, the tuning parameters should also change accordingly as the operation switches between regions.

The actual responses to these disturbances are obtained through experiment for a grid of $k_C$ and $\tau_i$ values. The simulated responses are obtained using the network model. The values of $k_C$ range from 1 to 50 and that of $\tau_i$ range from 1 to 100 with non-uniform grid spacing. The range of $k_C$ and $\tau_i$ is larger than that used in the linear experiment, partially because the system is different, but also because it is an attempt to explore the behavior of IAE and CVSE surfaces more completely. The IAE value for each disturbance at the particular $k_C$ and $\tau_i$ is obtained both experimentally and by network simulation. Results for the upper disturbance are discussed first.
PART 2. II. C. 2. a. Upper Disturbance

PART 2. II. C. 2. a. 1. Experimental Results

A surface plot of the experimental IAE as a function of \( k_c \) and \( \tau_i \) for the upper disturbance is shown in Figure 60 and specific values are given in Table 19. The global minimum IAE occurs at \( k_c, 10 \) and \( \tau_i, 35 \). Note that the location of global minimum is in a valley whose bottom is somewhat flat in the direction of the \( \tau_i \) coordinate indicating that IAE is relatively insensitive to the reset time. A wide range of \( \tau_i \) values, from 15 to 50, yield approximately the same IAE performance as the minimum value. In addition, as one can see from the table, the local minimum at \( k_c, 25 \) and \( \tau_i, 15 \) is also close to the global minimum. When noise is taken into account, the two are practically indistinguishable.

The relative importance of these two global minima can be determined by using CVSE as a secondary selection criteria. The behavior of the CVSE is shown in a surface plot in Figure 61, and the specific values are listed in Table 20. As a result of expanding the grid of \( k_c \) and \( \tau_i \) values, one sees that the global minimum now occurs where theory predicts, namely at the lowest \( k_c \) and highest \( \tau_i \). Among the candidate global minima, the point at
FIGURE 60
Integral Absolute Error (IAE) vs. kc and taul
Nonlinear Level Control in Conical Tank
Experimental Results- Upper Tank Disturbance
Step Up in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
Global Minimum at kc, 10 and taul, 35
### TABLE 19

**Experimental IAE Values for Nonlinear Level Control**

**Disturbance in Upper Tank**

Step in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5

<table>
<thead>
<tr>
<th>$k_C$</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>5.569</td>
<td>5.488</td>
<td>5.974</td>
<td>6.266</td>
<td>6.384</td>
<td>6.375</td>
<td>5.780</td>
</tr>
<tr>
<td>5</td>
<td>4.646</td>
<td>4.841</td>
<td>4.958</td>
<td>5.113</td>
<td>5.016</td>
<td>4.947</td>
<td>5.880</td>
</tr>
<tr>
<td>25</td>
<td>5.229</td>
<td>4.828</td>
<td>4.539</td>
<td>4.673</td>
<td>4.838</td>
<td>5.046</td>
<td>7.972</td>
</tr>
<tr>
<td>40</td>
<td>4.983</td>
<td>4.698</td>
<td>4.576</td>
<td>4.804</td>
<td>5.131</td>
<td>5.473</td>
<td>8.782</td>
</tr>
</tbody>
</table>
FIGURE 61
Controller Variable Squared Error (CVSE) vs. kc and tau
Nonlinear Level Control in Conical Tank
Experimental Results for Upper Tank Disturbance
Step Up in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
### TABLE 20

Experimental CVSE Values for Nonlinear Level Control
Disturbance in Upper Tank
Step in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5

<table>
<thead>
<tr>
<th>( k_c )</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \tau_{au} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>15.145</td>
<td>9.185</td>
<td>8.466</td>
<td>8.102</td>
<td>7.463</td>
<td>6.747</td>
<td>3.568</td>
</tr>
<tr>
<td>10</td>
<td>30.162</td>
<td>8.721</td>
<td>8.221</td>
<td>7.505</td>
<td>6.793</td>
<td>5.907</td>
<td>3.375</td>
</tr>
<tr>
<td>30</td>
<td>38.144</td>
<td>20.660</td>
<td>14.300</td>
<td>11.067</td>
<td>9.523</td>
<td>8.800</td>
<td>5.886</td>
</tr>
</tbody>
</table>
\( k_c, \) 10 and \( \tau_{u_i}, \) 35 has the lowest CVSE and is the more interesting operating point.

PART 2. II. C. 2. a. 2. Network Results

The behavior of IAE obtained from network simulation is shown as a surface plot in Figure 62 and the values are presented in Table 21. The shape of the surface plot is similar to that of the experimental case in that the hills and valleys are in the same areas. However, the simulated surface features are more exaggerated. One key similarity is the flat valley containing the global minimum. The global minimum is at the same point as that determined by experiment, namely \( k_c \) at 10 and \( \tau_{u_i} \) at 35. As can be seen from the table, the set of points equivalent to the global minimum include values with \( \tau_{u_i} \) from 10 to 50. One additional point not a member of the experimental set is at \( k_c, \) 10 and \( \tau_{u_i}, \) 1. One point that is present in the experimental set but absent here is the second global minimum at \( k_c, \) 15 and \( \tau_{u_i}, \) 25. Since this minima was deemed as unimportant from CVSE considerations, its absence is not serious.

For the upper disturbance, the simulated CVSE behavior is shown in Figure 63, and the values are shown in Table 22. The surface plot shows that the general
FIGURE 62
Integral Absolute Error (IAE) vs. kc and tau
Nonlinear Level Control in Conical Tank
Upper Tank Disturbance
Step Up in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
Global Minimum at kc, 10 and tau, 35
<table>
<thead>
<tr>
<th>$k_c$</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>6.721</td>
<td>6.955</td>
<td>7.139</td>
<td>7.205</td>
<td>7.097</td>
<td>7.421</td>
<td>8.897</td>
</tr>
</tbody>
</table>
FIGURE 63
Controller Variable Squared Error (CVSE) vs. kc and tau
Nonlinear Level Control in Conical Tank
Network Simulation (Trained with Experimental Data)
Upper Tank Disturbance
Step Up in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5
TABLE 22

Simulated CVSE Values for Nonlinear Level Control
Network Trained with Experimental Data
Disturbances in Upper Tank
Step Up in Set Point (0.5 to 0.75); Inlet Flow Rate, 0.5

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>1</td>
<td>3.900</td>
<td>3.624</td>
<td>3.466</td>
<td>2.753</td>
<td>2.187</td>
<td>1.819</td>
<td>1.385</td>
</tr>
<tr>
<td>10</td>
<td>3.727</td>
<td>3.447</td>
<td>3.316</td>
<td>3.188</td>
<td>3.056</td>
<td>2.826</td>
<td>1.369</td>
</tr>
<tr>
<td>40</td>
<td>4.342</td>
<td>4.165</td>
<td>4.017</td>
<td>4.037</td>
<td>4.046</td>
<td>3.981</td>
<td>3.866</td>
</tr>
<tr>
<td>50</td>
<td>4.368</td>
<td>4.190</td>
<td>4.121</td>
<td>4.140</td>
<td>4.134</td>
<td>4.014</td>
<td>4.013</td>
</tr>
</tbody>
</table>
features of the simulated behavior resemble that of the experiment except, the distortion between experiment and simulation is larger than that for the IAE surface. This distortion will be shown later to be of little consequence.

PART 2. II. C. 2. a. 3. Tuned Responses

In the analysis of both the experiment and simulation, the global minimum IAE occurs at the same point, namely $k_C$, 10 and $\tau_u$, 35. When the tuning parameters are changed from their initial values at $k_C$, 3 and $\tau_u$, 6 to these optimal values, the performance improves significantly for upper tank disturbances. A comparison of the responses to a step in set point from 0.5 to 0.75 with constant inlet flow rate at 0.5 between the initial and the optimal tuning is shown in Figure 64. The behavior of the response for the initial tuning is oscillatory whereas the behavior of that of the optimal tuning is smooth and the set point approached in an exponential fashion. In terms of the outlet flow rate behavior, the curve for the initial tuning is oscillatory with large amplitude excursions from the steady state value whereas the optimal setting is also oscillatory, but with lesser magnitude.
FIGURE 64
Comparison of Responses to Step Up in Set Point
Initial vs. Optimum Tuning
Nonlinear Experiment

Normalized Liquid Level

Set Point
Initial Tuning (kc, 3 & tau_i, 5)
Optimum Tuning (kc, 10 & tau_i, 35)

Normalized Flow Rate

Inlet
Outlet Initial Tuning
Outlet Optimum Tuning

Sample Time

0 25 50 75 100 125 150 175 200
A comparison of responses is constructed for another disturbance, a step in inlet flow rate from 0.3 to 0.7, and is shown in Figure 65. Notice that the liquid level response of the initial tuning tends to have both an overshoot and undershoot, whereas that for the optimum tuning tends to have the overshoot but no undershoot. The IAE value for the initial tuning is 1.652 and that for the optimal tuning is 1.467 or a drop of 11%. The CVSE for the initial tuning is 1.564 and that for the optimal tuning is 1.210 or a drop of 23%. Even though the tuning optimization process was based on a particular step in set point, the optimal tuning appears to improve the liquid level response and smooths the controller variable response for load disturbances like this step in inlet flow rate.
FIGURE 65
Comparison of Responses to Step Up in Inlet Flow Rate
Initial vs. Optimum Tuning
Nonlinear Experiment

Normalized Liquid Level

Set Point
Initial Tuning (kc, 3 & tau_i, 6)
Optimum Tuning (kc, 10 & tau_i, 35)

Normalized Flow Rate

Inlet
Outlet Initial Tuning
Outlet Optimum Tuning

Sample Time
PART 2. II. C. 2. b. Lower Disturbance

PART 2. II. C. 2. b. 1. Experimental IAE and CVSE

The neural network model has been sufficiently accurate to reproduce the IAE and CVSE behavior of the experimental system for a disturbance in the upper tank. The model's behavior for the disturbance in the lower tank will be examined. The experimental behavior of IAE for the lower disturbance is given as a surface plot in Figure 66 and the values in Table 23. The hills in this surface appear roughly as those for the disturbance in the upper tank, however the flat valley where the global minimum is has developed two sharp depressions. These features can be detected more clearly from the tabular data. The first is located at $k_c$, 10 and $\tau_i$ in the range 15 to 35 and the second at $k_c$ in the range 30 to 40 and $\tau_i$ at 6. Because of noise, the global minimum for IAE can be located in either depression at $k_c$, 10 and $\tau_i$, 25 or $k_c$, 30 and $\tau_i$, 6.

The experimental behavior of the CVSE quickly identifies which depression is significant. CVSE behavior is characterized by the surface plot in Figure 67, and the values are given in Table 24. As can be seen, the CVSE is lower for the depression near $k_c$, 10 and $\tau_i$, 25. This
FIGURE 66
Integral Absolute Error (IAE) vs. kc and tau
Nonlinear Level Control in Conical Tank
Set Up in Step (0.2 to 0.37); Inlet Flow Rate, 0.3
Global Minima at kc, 10 & tau, 25 or kc, 30 & tau, 6
Experimental Results - Lower Disturbance
### TABLE 23

Experimental IAE Values for Nonlinear Level Control Disturbance in Lower Tank
Step Up in Set Point (0.2 to 0.37); Inlet Flow Rate, 0.3

<table>
<thead>
<tr>
<th>( k_C )</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
<th>200</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.967</td>
<td>4.113</td>
<td>5.246</td>
<td>5.540</td>
<td>5.206</td>
<td>5.379</td>
<td>5.059</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2.436</td>
<td>2.640</td>
<td>2.851</td>
<td>2.926</td>
<td>2.896</td>
<td>2.776</td>
<td>2.716</td>
<td>4.326</td>
</tr>
<tr>
<td>5</td>
<td>2.230</td>
<td>2.341</td>
<td>2.423</td>
<td>2.415</td>
<td>2.296</td>
<td>2.251</td>
<td></td>
<td>3.788</td>
</tr>
<tr>
<td>10</td>
<td>2.427</td>
<td>2.241</td>
<td>2.192</td>
<td>2.101</td>
<td>2.222</td>
<td>2.615</td>
<td></td>
<td>4.667</td>
</tr>
<tr>
<td>25</td>
<td>2.597</td>
<td>2.204</td>
<td>2.216</td>
<td>2.257</td>
<td>2.396</td>
<td>2.910</td>
<td></td>
<td>5.201</td>
</tr>
<tr>
<td>30</td>
<td>3.271</td>
<td>2.107</td>
<td>2.225</td>
<td>2.258</td>
<td>2.457</td>
<td>2.937</td>
<td></td>
<td>5.221</td>
</tr>
<tr>
<td>40</td>
<td>2.815</td>
<td>2.147</td>
<td>2.210</td>
<td>2.328</td>
<td>2.489</td>
<td>2.920</td>
<td></td>
<td>5.325</td>
</tr>
<tr>
<td>50</td>
<td>3.161</td>
<td>2.289</td>
<td>2.253</td>
<td>2.329</td>
<td>2.525</td>
<td>2.978</td>
<td></td>
<td>5.433</td>
</tr>
</tbody>
</table>
FIGURE 67
Controller Variable Squared Error (CVSE) vs. kc and tauti
Nonlinear Lever Control in Conical Tank
Step Up in Set Point (0.2 to 0.37); Inlet Flow Rate, 0.3
Experimental Results - Lower Disturbance
TABLE 24

Experimental CVSE Values for Nonlinear Level Control
Disturbance in Lower Tank
Step Up in Set Point (0.2 to 0.37); Inlet Flow Rate, 0.3

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
<th>200</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.643</td>
<td>3.499</td>
<td>3.286</td>
<td>2.592</td>
<td>1.751</td>
<td>1.291</td>
<td>0.876</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3.136</td>
<td>2.682</td>
<td>2.513</td>
<td>2.326</td>
<td>2.068</td>
<td>1.738</td>
<td>1.005</td>
<td>0.729</td>
</tr>
<tr>
<td>5</td>
<td>2.592</td>
<td>2.441</td>
<td>2.328</td>
<td>2.091</td>
<td>1.814</td>
<td>1.462</td>
<td>0.767</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>5.322</td>
<td>2.231</td>
<td>2.145</td>
<td>1.815</td>
<td>1.590</td>
<td>1.251</td>
<td>0.613</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>7.055</td>
<td>2.368</td>
<td>2.133</td>
<td>1.804</td>
<td>1.468</td>
<td>1.193</td>
<td>0.544</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>7.504</td>
<td>2.416</td>
<td>2.150</td>
<td>1.784</td>
<td>1.511</td>
<td>1.155</td>
<td>0.516</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>9.107</td>
<td>2.726</td>
<td>2.198</td>
<td>1.854</td>
<td>1.524</td>
<td>1.182</td>
<td>0.563</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>11.80</td>
<td>3.321</td>
<td>2.296</td>
<td>2.020</td>
<td>1.675</td>
<td>1.332</td>
<td>0.677</td>
<td></td>
</tr>
</tbody>
</table>
is the global minimum that is selected as the operating point.

PART 2. II. C. 2. b. 2. Network IAE and CVSE

These experimental results are compared to those of the network simulation. Simulated IAE behavior is shown as a surface plot in Figure 68, and the values are listed in Table 25. Again the general topography of the surface resembles that of the previously presented corresponding experimental surface. In particular, the global minimum occurs in a sharp depression at the same location that of the experiment. Note that unlike the experimental results, a second global minimum is not present. This modeling error is not serious, because this second minimum was found to be inappropriate as an operating point.

The CVSE behavior for the network simulation is shown in the surface plot in Figure 69, and the values are given in Table 26. The topography of this surface is a departure from that of the experimental in many aspects, however the key property still remains, namely the downward slope of the surface toward low $k_C$ and high $\tau_i$. 
FIGURE 68
Integral Absolute Error (IAE) vs. kc and tau
Nonlinear Level Control - Lower Tank Disturbance
Set Up in Set Point (0.2 to 0.37); Inlet Flow Rate, 0.3
Global Minimum at kc, 10 and tau, 25
Network Simulation (net trained with experimental data)

IAE

0 5 10 15 20 25 30 35 40 45 50 55
kc

0 10 20 30 40 50 60 70 80 90 100
tau
TABLE 25

Simulated IAE Values for Nonlinear Level Control
Network Trained With Experimental Data
Disturbance in Lower Tank
Step Up in Set Point (0.2 to 0.37); Inlet Flow Rate, 0.3

<table>
<thead>
<tr>
<th>$k_C$</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>tau_i</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>-------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>1</td>
<td>3.035</td>
<td>3.850</td>
<td>4.998</td>
<td>5.169</td>
<td>5.212</td>
<td>5.230</td>
<td>5.235</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2.606</td>
<td>2.544</td>
<td>2.941</td>
<td>3.198</td>
<td>3.242</td>
<td>3.132</td>
<td>2.991</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2.410</td>
<td>2.061</td>
<td>2.219</td>
<td>2.111</td>
<td>1.877</td>
<td>1.620</td>
<td>1.743</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2.265</td>
<td>1.691</td>
<td>1.651</td>
<td>1.524</td>
<td>2.025</td>
<td>2.725</td>
<td>3.622</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>2.342</td>
<td>2.263</td>
<td>2.078</td>
<td>2.472</td>
<td>2.978</td>
<td>3.389</td>
<td>3.570</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>2.386</td>
<td>2.387</td>
<td>2.431</td>
<td>2.801</td>
<td>3.166</td>
<td>3.580</td>
<td>3.717</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>2.448</td>
<td>2.874</td>
<td>2.948</td>
<td>3.196</td>
<td>3.521</td>
<td>3.669</td>
<td>3.752</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>2.449</td>
<td>2.959</td>
<td>3.019</td>
<td>3.281</td>
<td>3.582</td>
<td>3.610</td>
<td>3.670</td>
<td></td>
</tr>
</tbody>
</table>
FIGURE 69
Controller Variable Squared Error (CVSE) vs. kc and tau
Nonlinear Level Control in Conical Tank
Lower Tank Disturbance
Step Up in Set Point (0.2 to 0.37); Inlet Flow Rate, 0.3
Network Simulation (net trained with experimental data)
**TABLE 26**

Simulated CVSE Values for Nonlinear Level Control Network Trained With Experimental Data

Disturbance in Lower Tank

Step Up in Set Point (0.2 to 0.37); Inlet Flow Rate, 0.3

<table>
<thead>
<tr>
<th>$k_c$</th>
<th>1</th>
<th>6</th>
<th>15</th>
<th>25</th>
<th>35</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>1</td>
<td>3.260</td>
<td>2.840</td>
<td>1.779</td>
<td>1.295</td>
<td>1.083</td>
<td>0.931</td>
<td>0.773</td>
</tr>
<tr>
<td>3</td>
<td>3.436</td>
<td>2.973</td>
<td>2.737</td>
<td>2.462</td>
<td>2.341</td>
<td>2.210</td>
<td>2.161</td>
</tr>
<tr>
<td>5</td>
<td>3.551</td>
<td>2.960</td>
<td>2.805</td>
<td>2.650</td>
<td>2.602</td>
<td>2.631</td>
<td>2.692</td>
</tr>
<tr>
<td>10</td>
<td>3.690</td>
<td>2.932</td>
<td>2.830</td>
<td>2.721</td>
<td>2.742</td>
<td>2.857</td>
<td>3.061</td>
</tr>
</tbody>
</table>
PART 2. II. C. 2. b. 3 Nonlinear TE

At this point, both the IAE and CVSE behavior of the experiment and simulation have been examined in the upper and lower portions of the tank. The simulation has been found to be significantly accurate to be useful in improving the tuning in terms of the IAE behavior. As in the analysis of the linear experiment, the CVSE information can be used in an attempt to smooth the control action. As before, the Total Error (TE) is used to combine IAE and CVSE. Unlike the previous analysis, the weighting factor (WF) in TE is not predetermined but instead depends on the percentage that the IAE is allowed to differ from that of the global optimum. In this case, it is allowed to rise of approximately 5%. The IAE, CVSE and TE are calculated from network simulated data.

In the analysis for the upper portion of the tank, as WF is varied from one to zero, the tuning parameters vary from $k_c$, 10 and $\tau_u$, 35 to $k_c$, 1 and $\tau_u$, 100 as shown in Table 27. Given the constraint on the maximum value of IAE, then WF cannot rise above 0.95 and the tuning parameters become $k_c$, 3 and $\tau_u$, 100. Experimental data shows that when the IAE optimum tuning parameters are replaced by these new parameters, the experimental IAE rises from 4.430 to 5.780 or 30%, and the
TABLE 27

Optimum Tuning Parameters vs. Weighting Factor (WF)

Disturbance in Upper Tank

<table>
<thead>
<tr>
<th>WF</th>
<th>k_c</th>
<th>tau_d</th>
<th>IAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>35</td>
<td>6.030001</td>
</tr>
<tr>
<td>0.97</td>
<td>3</td>
<td>100</td>
<td>6.123999</td>
</tr>
<tr>
<td>0.95</td>
<td>3</td>
<td>100</td>
<td>6.123999</td>
</tr>
<tr>
<td>0.9</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
<tr>
<td>0.6</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
<tr>
<td>0.4</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
<tr>
<td>0.3</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
<tr>
<td>0.2</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
<tr>
<td>0.1</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>100</td>
<td>6.439001</td>
</tr>
</tbody>
</table>

IAE Maximum Limit: 6.33150105
experimental CVSE drops from 6.793 to 4.611 or 32%. Modeling error principally in CVSE has resulted in a greater increase in IAE than was intended, but also a greater decrease in CVSE.

If the new parameters are implemented then a degradation in IAE performance occurs, but the CVSE performance improves. This is demonstrated by the experimental response curves shown in Figure 70 for a step in set point. Notice that the response of the liquid level for the TE optimal tuning does not adhere to the set point as well as that for the IAE optimal tuning, however the outlet flow rate response for TE optimal tuning is much smoother than that for IAE optimal tuning. The preliminary conclusion from this analysis is that for upper tank disturbances, the network model is sufficiently accurate in characterizing IAE and CVSE so that smoother controller action can be implemented.

The next task is to demonstrate that the network model is adequate to smooth controller action in the lower tank. As was done previously, the weighting factor for TE is varied from one to zero and the global optimum for the grid is recorded along with the corresponding IAE. These results are listed in Table 28. The maximum value of IAE is set at 1.631 which 5% above the IAE optimum of 1.524 at WF, 1. This limit restricts WF to 0.55 and makes the
FIGURE 70
Comparison of Responses to Step Up in Set Point
IAE Optimum vs. TE Optimum (Upper Tank)
Nonlinear Experiment
TABLE 29

Optimum Tuning Parameters vs. Weighting Factor (WF)

Disturbance in Lower Tank

IAE Maximum Limit: 1.631

<table>
<thead>
<tr>
<th>WF</th>
<th>k_c</th>
<th>tau_i</th>
<th>IAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>25</td>
<td>1.524</td>
</tr>
<tr>
<td>0.9</td>
<td>10</td>
<td>25</td>
<td>1.524</td>
</tr>
<tr>
<td>0.8</td>
<td>10</td>
<td>25</td>
<td>1.524</td>
</tr>
<tr>
<td>0.7</td>
<td>10</td>
<td>25</td>
<td>1.524</td>
</tr>
<tr>
<td>0.6</td>
<td>5</td>
<td>50</td>
<td>1.620</td>
</tr>
<tr>
<td>0.55</td>
<td>5</td>
<td>50</td>
<td>1.620</td>
</tr>
<tr>
<td>0.5</td>
<td>1</td>
<td>100</td>
<td>5.235</td>
</tr>
<tr>
<td>0.4</td>
<td>1</td>
<td>100</td>
<td>5.235</td>
</tr>
<tr>
<td>0.3</td>
<td>1</td>
<td>100</td>
<td>5.235</td>
</tr>
<tr>
<td>0.2</td>
<td>1</td>
<td>100</td>
<td>5.235</td>
</tr>
<tr>
<td>0.1</td>
<td>1</td>
<td>100</td>
<td>5.235</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>100</td>
<td>5.235</td>
</tr>
</tbody>
</table>
tuning parameters $k_c$, 5 and $\tau_{ai}$, 50 the optimum tuning in terms of TE.

When these new tuning parameters are implemented and compared to the results for the IAE optimal tuning, the experimentally measured IAE increases from 2.101 to 2.251 or 7%. and the CVSE decreases from 1.815 to 1.4624 or 19%. Again, a small degradation in IAE performance yields a large improvement in CVSE behavior. This is demonstrated experimentally in the response curves in Figure 71 for a step in set point in the lower tank. Notice that the responses of the liquid level for the TE optimum and IAE optimum are very similar. The response of the outlet flow rate for the TE optimum is smoother than that for the IAE optimum. Recall that a similar result as was also obtained for the upper tank.

The conclusion in this section is that the combination of IAE and CVSE through the parameter TE is useful in smoothing the behavior of the controller action for the nonlinear system. The weighting factor, WF, in TE should be based on the degree to which the IAE will be allowed to increase from the desired operating point. Even though modeling error is present especially in the CVSE parameter, the network model is adequate to allow for smoothing of the controller action significantly while at the same time maintaining the IAE behavior.
Figure 71
Comparison of Responses to Step Up in Set Point
IAE Optimum vs. TE Optimum (Lower Tank)
Nonlinear Experiment

---

Normalized Liquid Level

- Set Point
- IAE Optimum (kc, 10 and tau_i, 25)
- TE Optimum (kc, 5 and tau_i, 50)

---

Normalized Flow Rate

- Inlet
- Outlet IAE Optimum
- Outlet TE Optimum

---

Sample Time

0 25 50 75 100
PART 2. II. C. 2. c. Conclusions

Nonlinear identification using closed loop data was performed under the actual operating environment in the presence of noise and delay. The network model that was constructed from this data set performed excellently in one-step ahead prediction. The recursive prediction used in simulation was not perfect. However, it was sufficient to capture the behavior of IAE and CVSE for test disturbances in the upper and lower portions of the tank. The model was able to accurately predict the optimal operating points in terms of IAE for a large range of $k_c$ and $\tau_i$. It was also able to sufficiently predict CVSE so that the combined parameter $TE$ was successful at smoothing the controller action. In addition to its ability to point at the most promising operating points, the model was accurate in identifying areas to be avoided in terms of poor control performance and jerky controller variable action.
PART 2. III. Controller Optimization

A methodology for using BP networks to optimize the design of the controller algorithm is outlined in this section. Details of this strategy are left to future efforts. This methodology is an outgrowth of the previous tuning study, in that a model, like the closed-loop forward predicting model developed previously, can be used indirectly as a reference to aid in controller redesign. New control actions can be proposed and the model can be used to assess whether they improve performance.

This approach is different from the more direct inverted model approach whereby a model is structured which employs all past variables and to predict the current controller action (Gu, 1990). Identification of an inverted model is usually accomplished with open-loop data rather than closed-loop data, because of the higher information content of the former. Several inverse network models were constructed from closed-loop data from the test systems. The results suggest that the closed-loop inverse mapping does not automatically lead to an optimal control algorithm, but instead results in one that only approximately reproduces the controller used to train the network. This result may be different if continuous on-line identification is instituted, and the network
controller is implemented. Modeling errors which are likely to occur may then perturb the process and aid in the identification process. In effect, the modeling errors act much as the controller dither signal mentioned previously.

In the indirect method, the forward-predicting system model is used in teaching another network on the manner in which signals can be added to the existing controller action to achieve better control. This method begins by suggesting a number of test disturbances which span the entire dynamic range of the system. For any one disturbance, the network model can be used to simulate the closed-loop system response. Given that some method is available for calculating the action of the existing controller, this action is the starting point of a search for better control action. Perturbations to the action are proposed over a range several steps into the future. The forward-predicting model is then used to accept or reject the proposed actions according to some performance criteria. The criteria can be IAE and CVSE or TE as before. If the action deemed desirable, then the proposed perturbation is kept otherwise it is rejected. The exact number of searches required to determine the most desirable series of moves depends on the range that the perturbation is allowed to vary from the conventional
control action and how far into the future that the search
is made.

A perturbation identified in the search becomes the
output portion of training vector for a second BP network
called the compensator network. The input portion of the
vector is the concatenation of histories of variables that
characterize the system. The histories include variables
which occur concurrently and preceding the perturbation.
In the tuning application, this input vector was used to
predict the controlled variable in the next time step, but
here it is used to predict a compensation to the existing
controller action. The place of the compensator network
in the control scheme is shown in Figure 72. The function
of the compensator network is to recognize when a
particular disturbance is occurring and augment the
conventional control action with an additional action
which improves overall performance.

This methodology was tested in a preliminary manner
using the simulated linear tank model. A series of step
changes is set point were proposed that were thought to
span the dynamical range of the system. For each
disturbance, a grid of perturbations were made a few
percent plus and minus the existing control action. In
the eight cases examined, the model was able to identify
actions improved the simulated IAE of the respective
FIGURE 72

Optimal Network Compensator

Set Point, $y_{sp}$

Controller

Measurement

Network Compensator

System

Disturbance

$u_{c}$

$\triangle u$

$y$

$y_{sp}$ (history)

$y_{m}$ (history)

$u$ (history)
disturbance. This methodology was not automated, yet it should be in future research.

Several issues remain for future research. First, the degree to which the compensator signal can vary from the existing controller action should be limited from stability considerations. Second, the compensator must be prevented for issuing spurious signals when the system is at steady state. Third, the system must be allowed to learn from mistakes which result from modeling error. This will assist in the identification process for both networks. Fourth, the number of disturbance examples may be allowed to grow in accordance with those encountered in daily operations.

This type of adaptive model-based compensator potentially allows for on-line transformation of an existing controller algorithm to one that is optimally based. The required effort in fully examining this idea is beyond the scope of this work.
PART 3. THESIS REVIEW

PART 3. I. Summary

This thesis addresses the problem of identification of system models for control purposes. It examines application of the mapping algorithm called back-propagation which is a "black box" type of modeling software. This mapping algorithm offers the advantages of comparative simplicity of use, the ability to map by example and the capacity to form linear and the possibility of nonlinear mappings. The network model is constructed from closed-loop data in order to avoid the disruptive open-loop test. The techniques of closed-loop identification using back-propagation are devised and a method of automating the construction of a model suitable for process control applications is presented. These techniques are applied first to simulated systems and then to experimental systems, and the network representation is found to adequately reproduce disturbance responses for both linear and nonlinear systems. The application is then expanded to address the problem of tuning a PI controller. Simulations are performed on standard disturbances to determine the IAE and CVSE of the disturbance. The network model is sufficiently accurate to capture the important behavior of both variables.
Through a knowledge of IAE and CVSE behavior, tuning parameters are selected which dramatically improve the controller performance while at the same time achieve smooth controller action. This method is extended to suggest a search technique by which a conventional control algorithm may be improved by a back-propagation compensator.

Part 3. II. Conclusion

The use of back-propagation to model system dynamics appears to be practical for tuning purposes. The network is fully capable of modeling any system provided adequate examples of dynamical behavior are available. The process can be fully automated so that the time consuming nature of the modeling task can be minimized.

The use of network models derived from closed-loop identification for the purpose of model-based tuning appears to be promising. Fine tuning of PI or PID algorithms is a ubiquitous requirement in industry. Such models represent an excellent means of formulating more educated decisions on selecting various tuning regimes.
REFERENCES


Birky, Gregory J. and Mc Avoy, Thomas J., "A Neural Net to Learn the Design of Distillation Controls", Department of Chemical Engineering, University of Maryland, College Park, MD 20742, 1988.


Hecht-Nielsen, Robert, (Ref. b) "Applications of Counterpropagation Networks", Neural Networks, Volume 1, Number 2, 1988, pp. 131-139.


Liou, Cheng-Yuan, "Design of Neural Networks to Classify Sonar Targets", Neural Networks, Volume 1, Supplement 1, 1988.


Naidu, S. and Zafiriou, E. and Mc Avoy, T. J., "Application of Neural Networks on the Detection of Sensor Failure During the Operation of a Control System", Department of Chemical Engineering, University of Maryland, College Park, MD 20742, 1989.


SAIC (Science Applications International Corporation), ANSim Users Manual, Chapter 4, 1988


Wold, Svante, "Cross-Validatory Estimation of the Number of Components in Factor and Principal Component Models", Technometrics, Vol. 20, No.4, November, 1978

