PATTERN-BASED PROCESS CHARACTERIZATION AND GAIN SCHEDULING FOR NONLINEAR CHEMICAL PROCESSES

By

MANISH SINHA

Bachelor of Technology

Indian Institute of Technology

Bombay, India

1993

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 1995.

PATTERN-BASED PROCESS CHARACTERIZATION

AND GAIN SCHEDULING FOR NONLINEAR

CHEMICAL PROCESSES

Thesis Approved:

James R. Whiteley

Thesis Adviser

Randy S. Lewis

Gary L. Foutch

Thomas C. Collins

Dean of the Graduate College

PREFACE

Accurate characterization of process dynamics from on-line sensor data is the key issue in successful implementation of gain scheduling for controlling chemical processes. This work presents a development of pattern-based gain scheduling for process control. The approach employs process state maps constructed from windowed slices of multi-sensor plant trend data. Process identification is done using principles of similarity based pattern recognition. This technique provides a straightforward means to associate unique gain, integral time and/or derivative time controller settings with different states of the process. Simulation results show that better control performance may be achieved by use of gain scheduled controller as compared to the conventional fixed feedback systems.

I am sincerely grateful to my thesis advisor, Dr. Rob Whiteley. His guidance went well beyond that of academic supervision and technical advise, and showed a deep concern for the professional and personal development of his students. I wish to thank Dr. Randy S. Lewis and Dr. Gary L. Foutch who served on my committee. A special thanks also goes to Dr. Eduardo Misawa whose courses gave me a good understanding of nonlinear control theory.

I also wish to thank all my friends who gave me company during long, late nights at the lab. A special thanks goes to my roommates for their friendship. They have made my stay at OSU a most pleasant and enjoyable experience.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

LIST OF NOTATION

Abbreviations

ART	Adaptive Resonance Theory
COA	Center of Area
CSTR	Continuous Stirred Tank Reactor
EXACT	Expert Adaptive Controller Tuning
FOPDT	First Order Plus Dead Time
LPV	Linear Parameter Varying
LTIS	Linear Time- Invariant System
IAE	Integral of Absolute Error
ISE	Integral of Square Error
ITAE	Integral of Time-Weighted Absolute Error
ODE	Ordinary Differential Equation
OUR	Oxygen Uptake Rate
PI	Proportional-Integral
PID	Proportional-Integral-Derivative
SISO	Single Input Single Output

Roman Letters

e(t)	Controller error
f	sampling frequency (hr^{-1})
k	Reaction rate (hr^{-1})
p(t)	Value of controlled variable
p_s	Controller bias
x_i^j	Normalized sensor data representing j^{th} reading of the i^{ih} sensor
$y_m(t)$	Value of measured variable
y_{sp}	Value of setpoint
d_i	Euclidean norm of pattern vectors
C_P	Heat capacity (Btu/ $lb_m {}^oR$)
C_{PJ}	Heat capacity of coolant (Btu/ $lb_m {}^oR$)
F	Reactant flowrate (ft^3/hr)
F_C	Coolant flowrate (ft^3/hr)
K_C	Controller gain ($ft^3/hr\ {}^oR$)
K_f	Gain of final control element
K_m	Gain of measuring device
K_P	Process gain (${}^oR\ hr/ft^3$)
$\vec{M}$	Minkowski metric
S_i	Similarity measured by ART2 neural network
T	Temperature (${}^oR$)
T_C	Coolant temperature (${}^oR$)
X_i^j	Sensor data representing ^{jth} reading of the i^{ih} sensor

X^i_o	Zero value of i^{th} sensor

Greek Letters

$\Delta e(t)$	Rate of change of error
ΔH_{RXN}	Exothermic heat of reaction (Btu/lb.mol)
τ_I	Reset time (hr)
τ_D	Derivative controller parameter (hr/^oR)
τ_P	Process time constant (hr)
θ_I	Nominal pattern vector
ρ	Vigilance parameter
μ	Membership function
ω	Window length (hr)

CHAPTER I

Introduction

1.1 Motivation

Control of chemical processes has traditionally been performed using linear feedback controllers. Feedback control adjusts the manipulated variable in order to force the process to conform to a desired behavior. Such controllers are typically designed on an assumption that the process dynamics can be approximated by a linear time-invariant system (LTIS). With the advances in linear control theory, designing a good feedback controller for a LTIS plant is a relatively straightforward exercise. Unfortunately, the dynamics exhibited by chemical processes are typically nonlinear. For such nonlinear systems the controllers are often detuned to maintain an adequate stability margin but they perform poorly when the process drifts away from the design condition. Thus there is an incentive to investigate alternate methods for design of controllers with widely varying parameter dependent dynamics.

Gain scheduling, or more rigorously controller parameter scheduling, offers a good solution to compensate for process nonlinearities (Mellichamp et. al., 1966b, and Pott, 1984). In this approach controller settings are expressed as a function of one or more measured process outputs and are calculated on-line to maintain optimum and stable performance. This control methodology is termed gain scheduling because initially it was used to accommodate changes in process gain (K_P) only (Åström, 1983).

1.2 PID Controllers

The most popular feedback controllers used in the chemical industries are PI (proportional-integral) and PID (proportional-integral-derivative). As the name suggests, they adjust the manipulated variable depending on the magnitude of error (proportional), the cumulative error integrated over time (integral) and the rate of change of error or derivative of the error (derivative). Derivative action cannot always be used since it is sensitive to noise in the error signal. This control structure is shown in Figure 1.1. A controller weights the proportional, integral and derivative action depending on the tuning parameters. The general equation of a PID controller can be expressed as:

$$p(t) = \mathbf{K_C} \left[e(t) + \frac{1}{\tau_I} \int e(t^*)\, dt^* + \tau_D \frac{de(t)}{dt} \right] + p_s \qquad (1.1)$$

Error (e(t)) is defined in equation 1.1 as the difference between the desired value ($y_{SP}(t)$, the set point) and measured variable $y_M(t)$

$$e(t) = y_{SP}(t) - y_m(t) \quad (1.2)$$

where: p(t) is the manipulated variable.

p_S is the controller bias.

K_C is the proportional gain of the controller (adjustable).

τ_I is the integral time constant or reset time (adjustable).

τ_D is the derivative time constant (adjustable).

The controller bias (p_S) is the value of the manipulated variable at steady state, or when there is no net error (e(t)), and, hence, no control action. The variable p_S is also called the *controller bias*. For a PI controller τ_D is zero and thus insensitive to the rate of change of error.

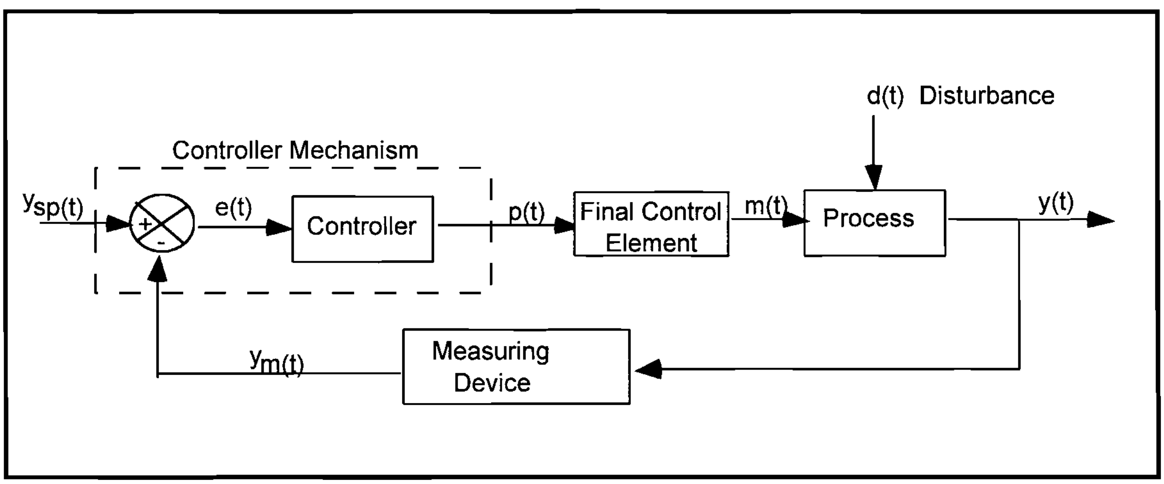


Figure 1.1: Schematic of a feedback control loop.

1.3 Gain Scheduled PID Controllers

Gain scheduling adjusts the controller parameters (K_C, τ_I and τ_D) to compensate for the changes in process dynamics. The key to successful implementation of gain scheduling lies in accurate characterization of the process dynamics. Mellichamp (1966) suggested that dynamic information can be inferred from estimation of the process gain (K_P). Bristol (1977) describes a pattern recognition method that characterized process dynamics from response to step input changes. In this work process dynamics are inferred using a pattern map of the process. All three methods are discussed in the following chapter.

While theoretically attractive, the practical implementation of gain scheduling remains difficult. Work by Shamma and Athans (1990) provides guidelines for selection of scheduling variables and operating conditions. One of the main drawbacks of gain scheduling is that the controller parameters are adapted in an open-loop fashion. There is no feedback to compensate for an incorrect schedule since the adaptation is performed in a feedforward mode (Åström, 1983). Thus the key issue for successful implementation of gain scheduling is to be able to accurately characterize the process dynamics, i.e. recognize the current state of the process and the state to which the process is evolving. Also, the way in which the scheduling is performed becomes critical because smooth transition in controller parameters is essential if one expects the properties of local linear controllers to carry over to the gain scheduled control system. This thesis addresses both issues.

1.4 Contribution of Thesis

This thesis is built on the initial work done by Anderson (1993) which looked at the use of similarity based pattern recognition to perform gain scheduling. Anderson showed that a multi-sensor pattern-based gain scheduling system performs better than a conventional fixed PI controller. Previous work used a neural network based pattern recognition approach.

The current work focuses more on how the sensor data can be effectively used to identify the process state rather than the tool to measure similarity between pattern vectors. Extracting information regarding the state of the process from information embedded in the multi-sensor trend patterns plays a key role in successful design of a gain scheduling control system. Guidelines to select scheduling variables are also presented. Thus the focus of this thesis has been to develop a novel method to accurately characterize process dynamics using the tools developed previously by Anderson (1993). This study has also investigated various interpolation techniques that can be used to perform gain scheduling. The emphasis has been to develop a strategy that is effective even during periods of transient operation. Using a nonisothermal continuous stirred tank reactor (CSTR) as a demonstration system, a formal methodology to design a gain scheduled controller has been developed.

1.5 Organization of Thesis

This thesis is organized into six chapters. In Chapter II different approaches to design a gain scheduled control system are presented. Our pattern-based approach is also introduced and various successful applications of gain scheduling are discussed.

In Chapter III an on-line identification technique to characterize different operating conditions is presented. A pattern based methodology to effectively utilize multi-sensor data is developed. Issues, like selection of scheduling variables and use of a sliding window for process characterization, are also discussed.

Chapter IV presents different strategies that can be used to perform controller parameter calculations after the state of the process has been established. The chapter begins with a discussion on the choice of the number of closest neighbors to be used for performing gain scheduling, followed by the presentation of three different interpolation strategies.

Chapter V demonstrates our pattern-based gain scheduling approach for a simulated CSTR. First the dynamics of the nonisothermal reactor are described. Next, a pattern-based gain scheduling control system is designed. Finally, the results are presented which are used to evaluate performance of our scheduling approach.

Concluding remarks, as well as a discussion on issues for future work, are documented in Chapter VI.

CHAPTER II

Gain Scheduling: An Adaptive Control Strategy

2.1 Overview

This chapter presents an overview of different methods that have been used to perform controller parameter adaptation. This chapter starts with a discussion on a need for controller parameter adaptation. The key issue in implementation of a gain scheduled system is identification of the process dynamics. Three methods, as well as the differences in the methodology used to characterize the process dynamics, are presented. Finally, some successful commercial applications of gain scheduling are discussed.

2.2 Need for Controller Parameter Adaptation

According to linear state feedback control theory, the best way of controlling any process is by measuring all the state variables in order to manipulate the process inputs in some desired fashion (such as the PID algorithm). This is rarely done in practice because all the state variables cannot be measured using on-line sensors and a rigorous knowledge of the existing process dynamics is not known. The most common control methodology utilizes a *single input single output* (SISO) structure. This is very popular because of the relative ease of the control system design. The main disadvantage of such a system is that it disregards the process dynamics of other measured and unmeasured process variables.

One alternative is to retain SISO methodology, but incorporate the effect of other process variables on the process dynamics by adaptively changing the controller parameter settings (e.g. K_C, τ_I and τ_D). For example, a chemical reactor is often controlled by measuring the temperature of the reactants and adjusting the coolant flow rate. It is well known that the dynamics of a reactor will vary with different feed and product compositions. Thus, the use of more than one variable to measure the process dynamics in a continuous manner becomes imperative. Information extracted from more than one process variable can then be used to schedule controller gains for the variations in the process dynamics. Such a control strategy is termed adaptive since the controller adapts itself to maintain satisfactory control for a nonlinear process.

Different approaches have been investigated to adapt PI controllers to respond to variations in process dynamics. Different approaches to gain scheduling are discussed in this Chapter with the aim at implementing this controller adaptation technique in real time.

2.3 Gain Scheduling Based on Process Gain

Mellichamp (1966b) presented an adaptive control system designed to maintain good control characteristics for processes showing wide variations in process gain. This method takes advantage of the fact that a good control can be maintained in spite of time varying process by holding the overall control loop gain constant. The overall open-loop gain for a PID control structure is the product of the gains of all the elements in the feedback control loop. Open-loop gain is expressed in equation 2.1 as:

$$K_{overall} = K_p K_m K_C K_f \tag{2.1}$$

where: K_P is the process gain.

K_m is the gain of the measurement device. (For example thermocouple)

K_C is the controller proportional gain.

K_f is gain of the final control element. (For example flow control valve)

If the objective of controller adaptation is to keep the open-loop gain constant, then a simple gain scheduling can be expressed as shown in equation 2.2. This is further illustrated in Figure 2.1.

$$K_C = \frac{\text{Constant}}{K_P K_m K_f} \tag{2.2}$$

The controller gain K_C is adapted to compensate for the variations in the process gain, K_P. This method can also compensate for the variations in the gains of the other control elements such as a control valve or other measuring device. The gain associated with the final control element (K_f) and measuring device (K_m) can be identified in a straightforward manner since they are inherent characteristic of these mechanical devices. The key step to successfully apply this method is to be able to accurately characterize the gain of a time varying nonlinear process (K_P). In an earlier paper, Mellichamp (1966a) presented a method for continuously estimating the gain of a process by applying sinusoidal perturbations to the process.

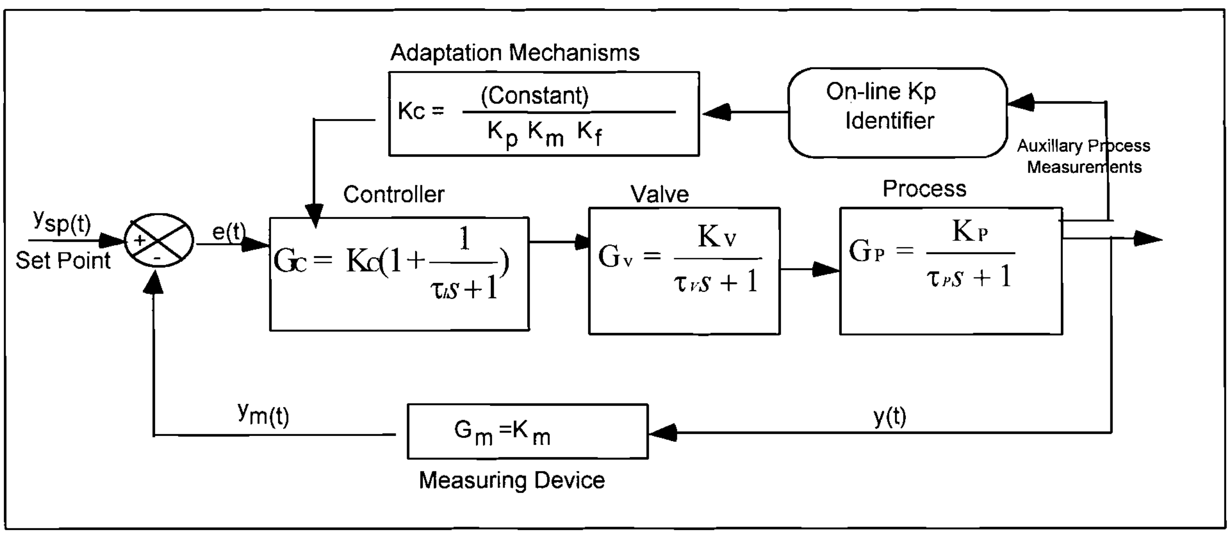


Figure 2.1: Feedback control loop and adaptation mechanism.

Such a control strategy is limited only to those systems whose dynamics can be approximated by a *first order plus dead time* (FOPDT) model. Many processes, such as exothermic reactions, have dynamics which cannot be approximated as FOPDT and are not amenable to Mellichamp's method. For the demonstration system used in this study the K_P varies considerably as the process is open loop unstable. This is explained later when the dynamics of the demonstration system are discussed. Keeping the open loop gain ($K_{overall}$) constant for such a system would lead to unstable controller performance. Moreover such a scheme neglects variation in other controller settings such as the reset time, τ_I.

Mellichamp's gain scheduling strategy assumes that the process dynamics can be characterized by estimating the value of the process gain (K_P). This simple gain scheduling approach may result in poor control unless the process dynamics are also considered (Seborg et al., 1986). For a process which has a long time-delay, this approach to controller parameter adaptation may result in a performance worse than conventional PID control unless some kind of time delay compensation is employed (Wong and Seborg, 1985).

2.4 Gain Scheduling Based on Error Diagnostics

Another way of approaching gain scheduling is as a self-tuning algorithm which adjusts the controller parameters based on the transient error pattern. Bristol (1977) was first to propose an algorithm to adapt the controller parameters based on a pattern recognition approach. Typically, with this approach the closed loop is perturbed and the resulting pattern of the response is observed. This pattern is compared with one that is personally desired. Key characteristics, such as damping and overshoot, are extracted from the recorded response to characterize the dynamics of the process. Such a self-tuning PID controller automatically adjusts the controller setting to result in a desired damping and overshoot of the response pattern.

Recently, much work has been done to characterize various process states and associate them with different pattern characteristics (Cao and McAvoy, 1990; Megan and Cooper, 1992). In this manner a gain scheduling algorithm can be set up which aims at driving the error pattern to a desired form. Figure 2.2 shows some typical transient pattern which can be analyzed to interpret the state of the process. The best response to a unit step in manipulated variable is represented by the center pattern in Figure 2.2. The deviation in characteristics of the error pattern recorded from desired pattern can be used to update the parameters of the controller. A table of pattern characteristics corresponding to different states of the process (as shown in Figure 2.2) is used. Once the transient pattern is characterized in reference to the table of different nominal patterns, the process gain (K_P) and the process time constant (τ_P) can be established. A rule along the lines of Zeigler-Nichols (1942) method can then be used to correlate the most appropriate controller settings.

The aim of a gain scheduler is thus to recognize characteristics of an error pattern as the process is perturbed by a unit step change in the manipulated variable. Controller parameter are adapted to drive the error response to the desired form (Cooper and Lalonde, 1990). This method, like any tuning exercise, is an iterative process. The characteristics of error response are first extracted and then controller adaptation is performed. This is repeated till the final error characteristic is similar to the desired error pattern. Highly advanced pattern recognition algorithms such as the ART2-A neural network, have been used for this purpose (Megan and Cooper, 1992, 1994, 1995).

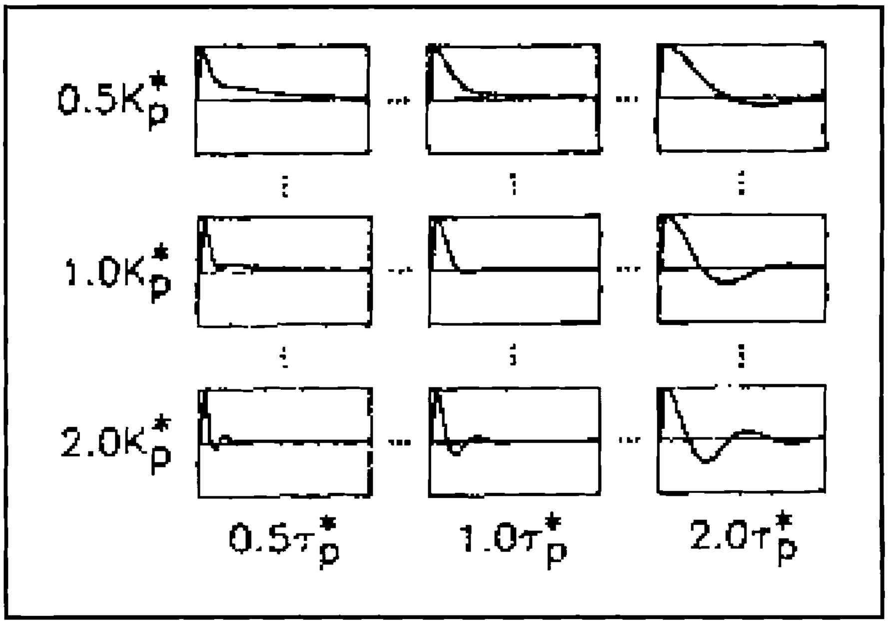


Figure 2.2: Exemplar patterns used for pattern recognition. (Megan and Cooper, 1992)

Controller tuning based on error diagnostics has been successfully commercialized (Kraus and Myron, 1984) and offers an excellent way to adapt controller settings for a time varying system. This commercially available self tuning controller has been dubbed as "EXACT" which stands for Expert Adaptive Controller Tuning by the Foxboro company. The EXACT controller utilizes expert-system techniques and artificial intelligence to tune PID coefficients with no need for process modeling. This has been successfully used to tune PID coefficients for chemical processes showing large dead-time and nonlinearities.

Recently, fuzzy reasoning has also been used for this application in an effort to implement gain scheduling in real time. Zhao et al. (1993) arrived at a gain scheduling methodology which utilizes fuzzy rules and reasoning to determine the controller parameters based on the error signal and its first derivative (rate of change of error). This approach is schematically presented in Figure 2.3. The strategy of adapting the controller parameter with reference to error (e(t)) and rate of change of error (Δe(t)) is represented in a form of rulebase. A typical fuzzy rule could be:

"If *the error* is small and the *rate of change of error* is large, then reduce the controller gain (K_C) and increase the reset time (τ_I)."

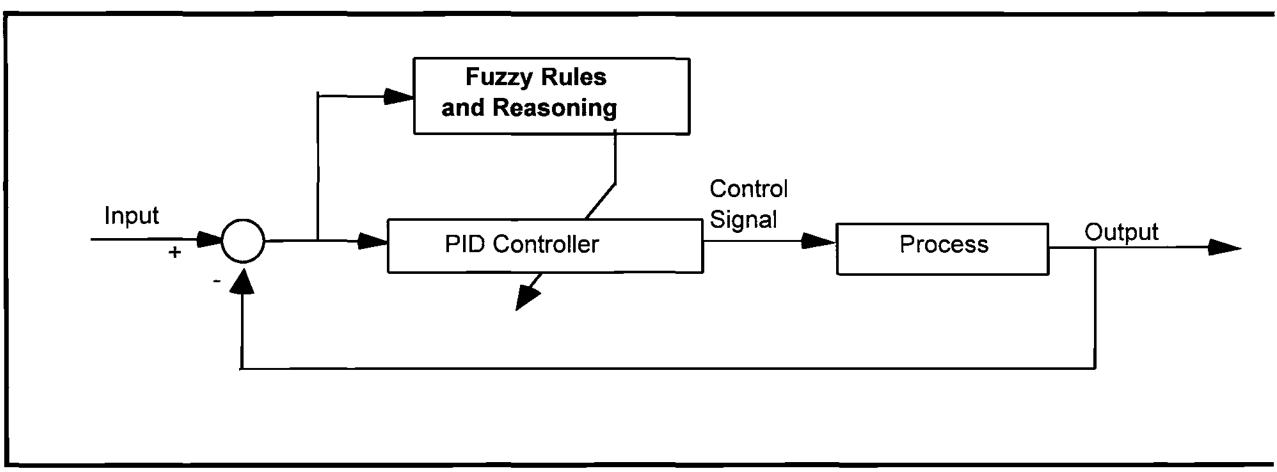


Figure 2.3 : Schematic of a gain scheduler using fuzzy reasoning. (Zhao et al., 1993)

For this strategy the gain scheduling is performed only when a step change in setpoint is made. The scheme is ineffective when the process is in steady state or while the process is evolving to a new steady state.

Controller parameter adaptation based on error diagnostics cannot be used to schedule PI controllers in real time. The main reason being that the steady state operation of the process needs to be perturbed in order to obtain the error diagnostics. Nevertheless, this technique is an excellent alternative to *trial and error* tuning (or "hand" tuning) routinely performed by the process expert to update the controller parameters. In a way this methodology has automated the "*art*" of controller tuning by use of artificial intelligence.

2.5 Gain Scheduling Using Pattern-Based Tuning Map

Perturbations in input variables to record error diagnostics are often not possible since it disrupts the smooth operation of a plant. Characterizing process dynamics based on the process gain (K_P) is also not effective for processes which cannot be modeled as an FOPTD model. Thus there is a big incentive to develop a method to accurately characterize process dynamics using some other approach. Our approach uses a process state map constructed from windowed slices of multi-sensor plant trend data. This pattern-based process state map is used to characterize process dynamics at different operating conditions. Thus a method to uniquely associate gain, integral and/or derivative time controller settings with different states of the process is developed.

In this approach controller parameter adaptation is performed by treating a nonlinear plant as a set of localized linear processes (Rugh, 1991). The number of operating conditions at which the process is linearized to approximate the plant dynamics depends on the extent of process nonlinearity and are normally chosen to cover the expected range of plant dynamics. The dynamics of the process are characterized at different operating conditions and appropriate controller settings are determined for each of these operating conditions. An automated "tuner" such as Foxboro's EXACT controller may be used to determine a good set of controller settings at each of the operating conditions. A table of controller settings at each of the operating conditions is referred to as a "*gain map*" or "*gain table*" in this study. This "gain map" relates the process operating conditions to the desired controller settings. During plant operation the process state is identified on-line and as the process moves from one operating condition to another, the controller parameters are automatically changed. The adaptation strategy is schematically presented in Figure 2.4.

Our approach uses a novel pattern-based technique for scheduling controller parameters. More than one process variable is considered over a finite period of time (window) to accurately characterize the process state. The features extracted from these multi-sensor pattern trends are analyzed and the process state is identified by measuring the similarity of the pattern vector corresponding to the process state with each of the nominal operating points.

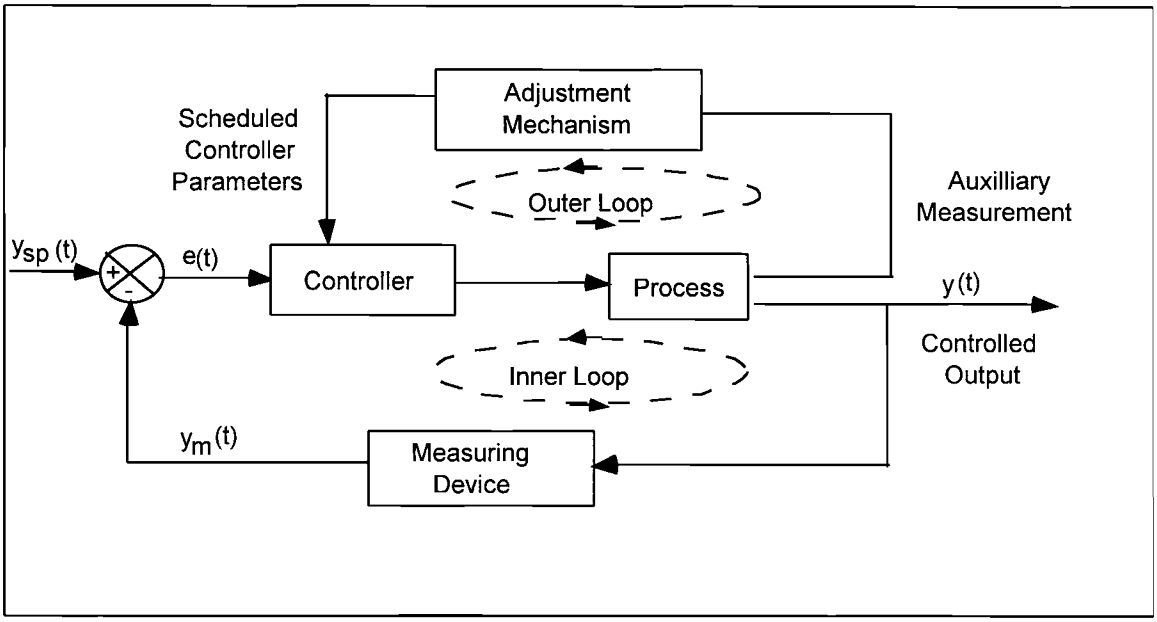


Figure 2.4: Gain scheduling: A programmed adaptive control system.

Our pattern-based gain scheduling approach involves three basic steps:

- Selection of nominal conditions representing steady operating process conditions.
- Characterization or identification of the process operating state from multi-sensor pattern trends.
- Scheduling the controller parameters.

Though this work has looked at implementing gain scheduling for a PI controller, the strategy is not limited to PID type controllers. Rather, the same strategy can be used for any linear controller whose parameters can be scheduled to compensate for the nonlinearities in the process.

2.6 Applications of Gain Scheduling

A number of industrial applications of gain scheduling implementations can be identified. The popularity of such a design technique lies in the fact that it is an adaptation strategy to the ubiquitous PID algorithm which has time proven utility and applicability to many of the industrial feedback control problems.

2.6.1 Aircraft control

Early development of gain scheduling included applications in high performance aircraft for design of an auto pilot (Seborg et al., 1986). It was found that monitoring the mach number and dynamic pressure allowed a suitable schedule to be developed (Åström, 1987). Åström's work looked at the effect of state variables other than the controlled variables (altitude and direction) on the flight dynamics. Since then gain scheduling has become a primary method to compensate for variations in flight control problems. It is used extensively in the design of an auto pilot system (Stien, 1980). Gain scheduling was initially limited to the aircraft industry. With the arrival of computer control systems, gain scheduling has become easier to implement (Åström and Wittenmark, 1989).

2.6.2 Ship Auto Control

This involves use of gain scheduling to compensate for parameter changes due to environmental changes. These changes include wind velocity, water currents and ship movements such as sway and yaw. A table look up approach has been used to do gain scheduling (Kallstrom et al., 1979). The main goal of this system was to reduce drag in the ship movement. The tuning values were changed as a continuous function of a ship's speed. Basically the controller parameters were read directly from a table when the change in wind velocity and water currents were determined.

2.6.3 *Process Control*

The "EXACT" controller is a commercially available adaptive controller that uses pattern recognition to achieve desired closed loop characteristics. Although the complete algorithm is proprietary, its basic features have been published (Bristol and Kraus, 1984). This self tuning PID controller provides a micro-processor based tool for use at the front line of process control (Kraus and Myron 1984). The Foxboro company has dubbed this new controller "EXACT" for Expert Adaptive Controller Tuning.

One notable case where on-line adaptive control has been widely used is to control pH of a reacting system. The wide variations in titration curves with changes in buffering makes pH control ideal for on-line adaptive control methods. Gain scheduling has been successfully used in controlling pH of a reacting system (Åström, 1987). Gain scheduling has been performed by storing the parameters in a chart form. The feedback controller can be effectively used in the entire range of operating conditions.

Cardello and San (1988) have looked at gain scheduling for batch bioreactors. Batch processes are extremely nonlinear and their dynamics show wide variations with time. They found that Oxygen Uptake Rate (OUR) can be used as a scheduling variable. A table look-up is used to select the gain depending on the OUR measurement. In comparison to a fixed PID controller and feedforward - feedback controller, the integral of the square of the error (ISE) was 20% less for the gain scheduled method. Cardello and San found that scheduling was an effective method for controlling dissolved oxygen levels in a batch fermentor. These conclusions are important as the fermentor had large variations in process load.

Leuba et al. (1992) have extended the concept of gain scheduling and fuzzy logic to set up an adaptive PI controller. The controller controls the fluid level in U-tube steam generators. The fuzzy logic circuit analyzes the disturbance and based on that information, decides what the controller effort should be. The gain scheduling aspect of the system changes controller parameters based on the temperature of the feed water. The gain is adjusted as a linear function of the feed water temperature. This led to a much smoother and stable performance.

2.7 Need for Future Research

Gain scheduling has emerged as a very powerful and useful technique to reduce the effect of parameter variations. It is in fact the predominant method to handle parameter variations in flight control systems. Gain scheduling seems to have gained wide acceptance for processes whose operating state can be identified by some auxiliary variables. Auxiliary variables are measurable process variables other than the controlled variable. From a review of the process control applications where some kind of gain scheduling has been applied it is clear that controller parameters are scheduled using a table look-up approach. Once a new operating condition is identified by a process expert, the controllers are retuned. In applications, such as fluid level control in U-tube steam generators (see subsection 2.6.3), an instantaneous value of a single variable is used to characterize the state of the process. These techniques have not been widely accepted in the process industries because it is often difficult to characterize and differentiate one operating condition from another. There is thus a need to develop a methodology to apply this technique in an on-line fashion. Microprocessor based controllers can be easily programmed to store a table of controller settings and even calculate them on-line. For process control systems which have large time constants and time delay there needs to be a proper method to be able to identify the process condition by looking at the plant variables or the sensor readings. The aim of this study thus focuses on development of a technique which uses pattern-based information to characterize process dynamics from on-line sensor data.

CHAPTER III

Pattern-Based Process Characterization

3.1 Overview

An on-line identification technique to characterize different operating conditions is presented in this chapter. The characterization of process dynamics is the key issue for designing a gain scheduled controller. The first step in setting up a gain scheduled controller is to approximate the nonlinear plant by a set of linearized models. The method to represent each of the *nominal conditions* at which the plant dynamics are linearized is discussed in this chapter. Once a set of operating conditions are identified the process dynamics at these conditions are mapped to form a *pattern-based tuning map.* The key step in this pattern-based approach is to map the process dynamics at different operating conditions. Which scheduling variables are chosen to construct the pattern-based tuning map is also an important issue. The use of such a map to characterize process dynamics in an on-line fashion is discussed next.

3.2 Formation of Pattern-Based State Map

A gain scheduling algorithm needs a way to map the controller parameters to variations in process dynamics. A state map relates the process dynamic characteristics at different operating conditions to a set of well tuned parameters which will result in satisfactory and stable control action at those process conditions. A state map of all possible operating conditions corresponds to a "*gain surface*". A gain surface can be described as a plot of controller parameters (for example controller gain, K_C) at all possible process operating conditions. The topology of this "surface" is the characteristic of the process under consideration. We do not propose to associate controller parameters at all possible operating conditions to arrive at this "gain surface". Rather, the gain surface is approximated by interpolating the controller parameters when the process operates between the nominal points.

Formation of state maps typically involves three different steps:

- Selection of nominal conditions.
- Selection of scheduling variables to identify these nominal conditions.
- Represent the nominal conditions by the multi-sensor pattern trends typically seen at those conditions

3.2.1 Selection of Nominal Conditions

Nominal conditions are operating conditions at which process dynamics are represented by a linear model and at which satisfactory controller parameters are available. The choice of nominal operating conditions is often based on the historical data of the plant. The conditions at which the plant normally operates are the first choice as nominal operating conditions. Generally, a few more operating conditions are chosen so that the entire nonlinear operating region can be reasonably approximated.

A process state can often be associated with sensor trends of certain measured variables. Most chemical processes can be modeled by mass momentum and energy balances. Mass, energy and momentum in turn can be characterized by variables such as density, concentration, temperature and flow rate. These characterizing variables are called state variables and their values define the state of the processing system (Stephanopoulos, 1990). According to linear control theory, the process dynamics are directly related to the values of "*state variables*" rather than the values of measured variables. For the demonstration system the controller settings are also distributed in the same manner as the operating conditions in the state space and that there can be one to one mapping between the two. That is, no two operating conditions have similar controller parameters associated with them. Thus the nominal conditions are chosen such that they cover the range of dynamics of the plant and represent the conditions at which the plant normally operates.

3.2.2 Selection of Scheduling Variables

The next implementation issue is to identify process variables that can then be used to characterize the process state. Often the state of the process is inferred from the instantaneous value of the controlled variable. For example the dynamics of a reactor is often inferred from the temperature at which the reaction is occurring. Since the process dynamics are a function of more than just the controlled variable there is a need to identify other process variables which can be considered for interpreting the process dynamics. The most common method for selecting schedulings variables is to look at the physics of the process and choose variables which can help characterize it. The designer is often limited to choose the scheduling variables from the measured process variables. For example often it is not possible to measure the production rate since the product concentration cannot be analyzed in an on-line fashion.

Shamma and Athans (1992) in their work on developing a theoretical analysis of gain scheduled systems present two guidelines for selection of scheduling variables. They are:

1. The scheduling variables should capture plant nonlinearity.
2. The scheduling variables should move slowly.

The first guideline is only a reminder that the plant models are only linearized approximations to the nonlinear plant. Similarly the second guideline that the scheduling variables should vary slowly is a reminder that the design model explicitly assumes a fixed operating condition. Previously these heuristic rules of thumb were verified by simulations though the work by Shamma (1988) and shows that these guidelines have a rigorous mathematical justification.

Since the identification of a nominal operating condition is performed by looking at the patterns of the scheduling variables, the sets of process variables representing a particular operating condition should be as distinct as possible from "neighboring" operating conditions. Often the number of process variables which can be measured by on-line sensors are limited. For example concentration cannot be accurately measured on-line. Thus there may be a need to identify certain "*virtual variables*" which can be computed on-line to help establish the current process state. A virtual variable is a combination of one or more measured variables whose trend provides valuable information regarding the process dynamics. An example of a virtual variable for a reactor could be the "concentration" of the reactant inferred from reactor temperature and a reaction kinetic model. Often it is possible to identify virtual variables that not only capture information regarding plant dynamics, but also move "slowly" during periods when the process is under transition from one operating state to another. Since at times it is not possible to identify slow varying real variables which can be used to uniquely associate different operating conditions, the use of virtual variables becomes important.

One advantage of using patterns of these scheduling variables is to make process characterization during periods of transition much more slow and smooth. During periods of transition some variables fluctuate a lot before they settle down to the value corresponding to the new steady state. By using patterns of process data the effect of such fluctuations during period of transition can be reduced. The guidelines to select scheduling variables to form a pattern map are as follows:

1. Patterns of scheduling variables should capture plants nonlinearites
2. Patterns of scheduling variables should vary slowly as the process moves from one operating condition to another.

3. The pattern representation (in a vector form) of the scheduling variables characterizing an operating condition should be as distinct from each other as possible.

3.2.3 Representation of Nominal Conditions Using Multi-Sensor Pattern Trends.

The final implementation issue in formation of a state map is to arrive at a way to represent the *"nominal conditions"*. It is generally not possible to accurately characterize the process dynamics by just measuring the instantaneous value of scheduling variables. An instantaneous view of the process gives no information regarding the process state during periods of transition. In another words the process state cannot be accurately characterized by instantaneous views.

The use of an instantaneous view of the process variables is prone to errors because of disturbances and transient periods. The fact that the instantaneous view of the process can result in wrong process characterization will be demonstrated using an illustrative example. The process state of a reactor is often inferred from the reactant and coolant temperatures. Gain scheduling is done based on the steady state at which the process is operating. When the process is in transition and evolving from one steady state to another an instantaneous view can result in a wrong process characterization. If only the instantaneous value of a single variable or more than one variable is used to establish the state of the process, then there is always a danger that the process may "appear" to go through other steady state operating conditions during transition periods. This is shown in Figure 3.1. If only an instantaneous value of temperature is used then the process appears to be operating at steady state 3 (SS_3) while the process is actually in transition (see Figure 3.1a). If two variables (reactant and coolant temperature) are used, even then the instantaneous view of the process may result in a wrong process characterization (see Figure 3.1b). A view of the time history of the process or the pattern trend of temperature data (see Figure 3.1a) reveals that the process actually went through a period of transition before evolving to steady state 2 (SS_2). Thus if the process state is viewed from windowed slices of sensor data then it is possible to accurately characterize the state of the process. The process dynamics at any operating condition would be very different depending on whether the process is steady or is in transition at that condition. Furthermore if the objective of the on-line gain scheduler is to track the process and simultaneously schedule parameters depending on the operating conditions, then the idea of operating condition being represented by a single dimensional vector of process variables can result in a wrong schedule. There is a need to develop a method that uniquely identifies each of the operating conditions even during periods of transition.

Traditionally a gain map relating the process operating conditions to fine tuned controller parameters were made using instantaneous views of single scheduling variable. We propose to replace the traditional gain map by a more robust, pattern-based gain map. Accurate characterization of a process state requires consideration of more than one variable for a finite period of time (Anderson and Whiteley, 1993a). In this new approach the single value scheduling variable is substituted by a multi-sensor pattern to characterize the process more accurately and thus significantly improve the gain scheduling during periods of transient operation. Use of multi-sensor pattern trends to

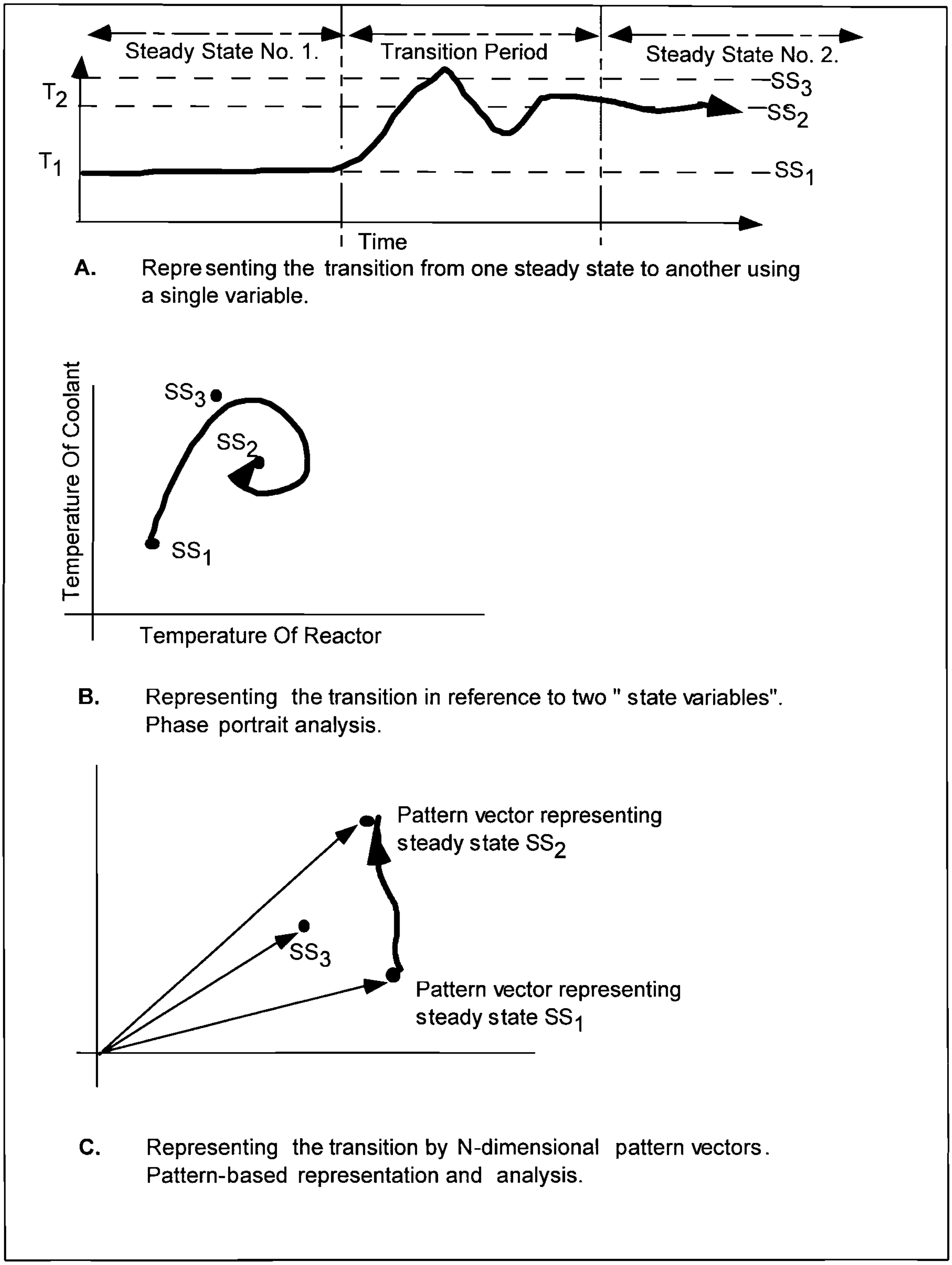


Figure 3.1 : Representation of the transient period as the process evolves to a new steady state.

identify process dynamics can result in considerable improvement in process characterization during periods of transition.

3.2.4 Arriving at a Pattern-Based Representation

The last step in the formation of a pattern-based state map is to represent the multi-sensor patterns by a vector in a multi-dimensional pattern space. First the time period over which the process variable needs to be considered must be specified. This time period is called the "*window length*". Window length is thus the time for which the process needs to be observed before its state can be characterized. The length of the window is a characteristic of the process for which the controller is being designed. Some processes show more "sluggish" response to changes in set point than others. How a window of process data can be transformed to be represented by a point in multi-dimensional pattern space is shown in Figure 3.2.

The time constant (τ_P) of a process is a measure of the time necessary for the process to adjust to a change in its input. For a first-order process, the process evolves to a new state in about five times the process time constant (τ_P) (Stephanopoulos, 1990). The same relation can be used to arrive at the window length as given in equation 3.1.

$$\text{Window Length } (\omega) = 5 \times \tau_P \qquad (3.1)$$

Where τ_P is the process time constant.

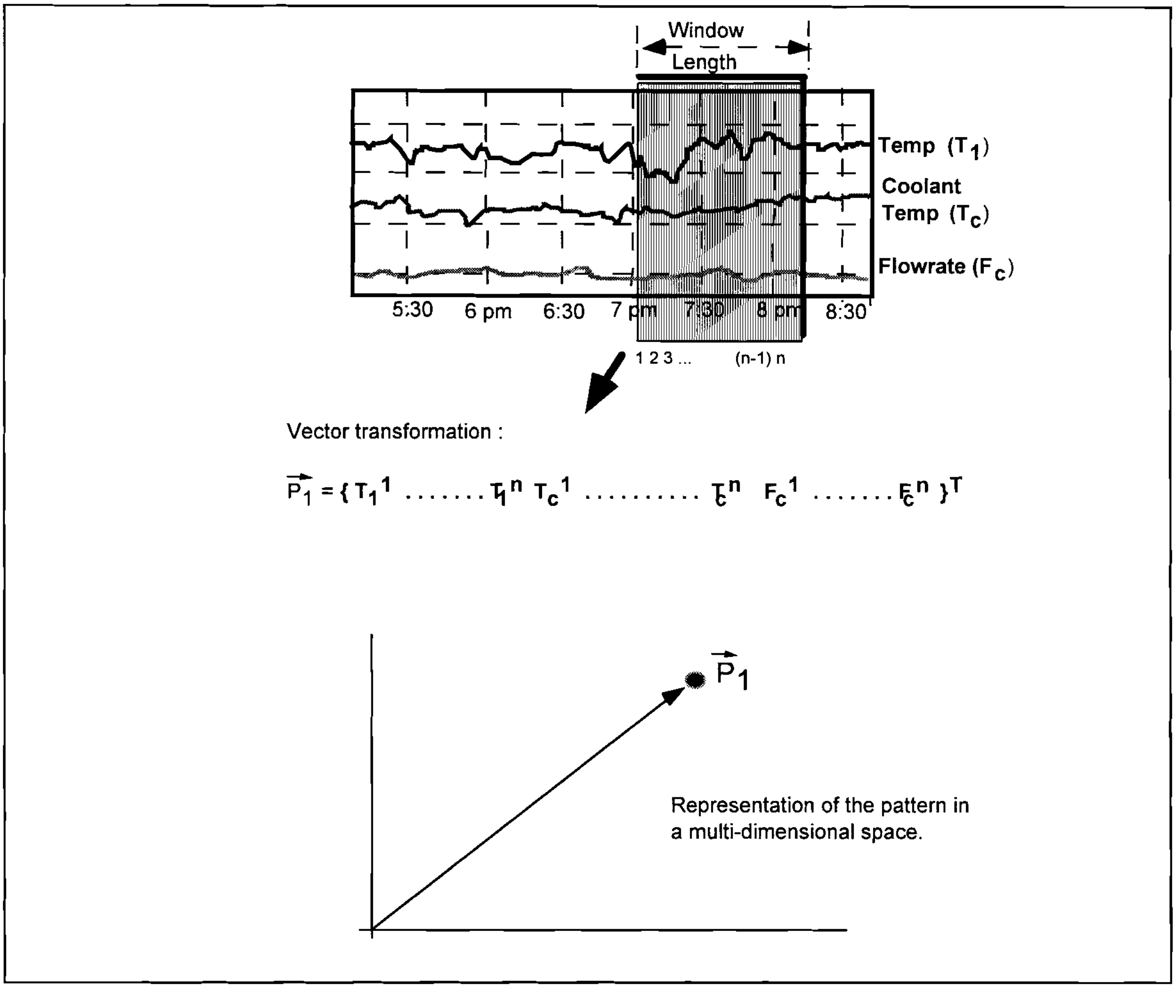


Figure 3.2: Pattern representation of an operating condition.

In Figure 3.2 it is shown that the a window of process data can be represented as a vector. To arrive at this vector representation the process data is first normalized. Normalization is done so that the variable can be expressed as a value between 0 and 1. This is important for doing any similarity analysis on the pattern vectors since they need to be expressed on a uniform scale. Equation 3.2 is used for normalization.

$$x_i^j = \frac{X_i^j - X_0}{span} \tag{3.2}$$

where

x_i^j is the normalized value corresponding to the i^{th} sample reading of the j^{th} sensor.

X_i^j is the actual sampled value of the i^{th} sample reading of the j^{th} sensor.

X_0 is the *zero* of the measurement device.

span is the range of the measurement device.

The normalized pattern vector can be expressed in equation 3.3 as:

$$\theta_i = (x_1^1, x_2^1, \ldots\ldots x_d^1, x_1^2, \ldots\ldots\ldots x_d^2, \ldots \ldots, x_1^\alpha, \ldots\ldots\ldots, x_d^\alpha)^T) \tag{3.3}$$

where:

x_i^j is the sample reading of j^{th} sensor at the i^{th} sensor reading.

d is the number of sampled data points ($d = \frac{\omega}{f}$) where ω is the window length and f is the sampling frequency.)

α: Number of different sensor patterns.

The total number of samples making up a single sensor pattern is d. Since there are α sensors the dimension of a pattern is "$\alpha \bullet d$". For example, a multi-sensor pattern composed of 10 discrete samples of three different variables can be represented in a 30 dimension vector space.

In this way all the nominal operating conditions can be represented in pattern space. Controller parameters are then associated with each of the nominal pattern vectors. This representation is thus termed as a "*pattern-based gain map*". Gain map is actually a misnomer since controller parameters other than the controller gain (K_C) may be associated with each of the nominal pattern vectors. The following section describes an on-line methodology to identify the process state based on the location of the process on the gain map.

3.3 On-Line Process Characterization

This section describes a method to identify the process state from a window of multi-sensor data. This work is based on earlier work done to develop a pattern recognition methodology to characterize process sensor data (Anderson, 1993).

The state of the process is characterized by measuring the similarity between the pattern vectors corresponding to the *prototypes* and the on-line sensor data. *Prototypes* are the pattern vector representation of the nominal operating conditions. During on-line operation the sensor readings are first normalized and represented in a pattern vector form as discussed in the previous section.

In this section, the similarity index to compare two pattern vectors will be presented. The similarity index will then be used to characterize the process state. Various proximity indices can be used to compare two multi-dimensional vectors. The most common index for such patterns is the Minkowski metric:

Minkowski metric ($\vec{\mathbf{M}}$) is defined by equation 3.5:

$$\vec{\mathbf{M}}(\mathbf{y},\theta_i) = [\sum |\theta_i^j - y_i^j|^r]^{1/r} \qquad \text{where } r \geq 1 \qquad (3.5)$$

where **y** is the vector representing the on-line sampled data.

θ_i is the vector representing one of the nominal operating condition in the pattern-based gain map. θ_I is also referred as "prototype".

Euclidean Distance ($\vec{\mathbf{d}}$)is defined in equation 3.6 by:

$$\vec{\mathbf{d}}(\mathbf{y},\theta_i) = [\sum |\theta_i^j - y_i^j|^2]^{1/2} \qquad (3.6)$$

Euclidean distance or Euclidean norm is a special case of Minkowski metric with n equal to 2. Euclidean norm, $\vec{\mathbf{d}}(\mathbf{y},\theta_i)$, is a measure of dissimilarity. The smaller the value of $\vec{\mathbf{d}}(\mathbf{y},\theta_i)$, the closer or more similar the pattern vectors. Euclidean distance is the most common of the Minkowski metrics. The familiar geometric notions of invariance to translations and rotations of the pattern space are valid only for Euclidean distance. Euclidean distance has been widely used in engineering work. Figure 3.3 shows how a two dimensional vector can be compared with a prototype.

The distance between two pattern vectors is measured as a Euclidean norm. As shown in Figure 3.3, there can be a locus of points having the same Euclidean norm. Accurate process characterization in those cases requires measurement of the angle also. Angle is a measure of how the vector points are oriented in the pattern space with respect to each other. For the purpose of gain scheduling, the process needs to be characterized in respect to more than one "nominal" operating condition. Euclidean norm as a similarity measure is adequate for gain scheduling purposes. This can be explained using Figure 3.3. For a two dimensional vector only two prototypes can be used to infer the process state in the pattern space. The locus of equidistant points intersect at two points. Using three prototypes one can uniquely associate the pattern vector in the pattern space. As shown, there can be only one unique combination of *d1:d2:d3* for the pattern vector shown. For a n dimensional space the number of prototypes needed to uniquely characterize any pattern vector is n+1. Thus even a distance measure (such as Euclidean norm) gives a good estimate of "where" the input pattern vector is located. The angle is not one of the more importance parameter since the similarity is reflected only in associating how much weightage the controller parameter of the nominal condition will have on the final controller value.

To determine the location of a pattern with respect to the prototypes, a gravitational pull analogy can be used. The closer the pattern vector is to any of the prototypes, the more "pull" or influence it should exert on the controller parameters. The final controller settings can be determined by combining the "pulls" or proximity indices of two or more.

How these "pulls" are analytically combined is a separate issue which will be discussed in detail in the following chapter.

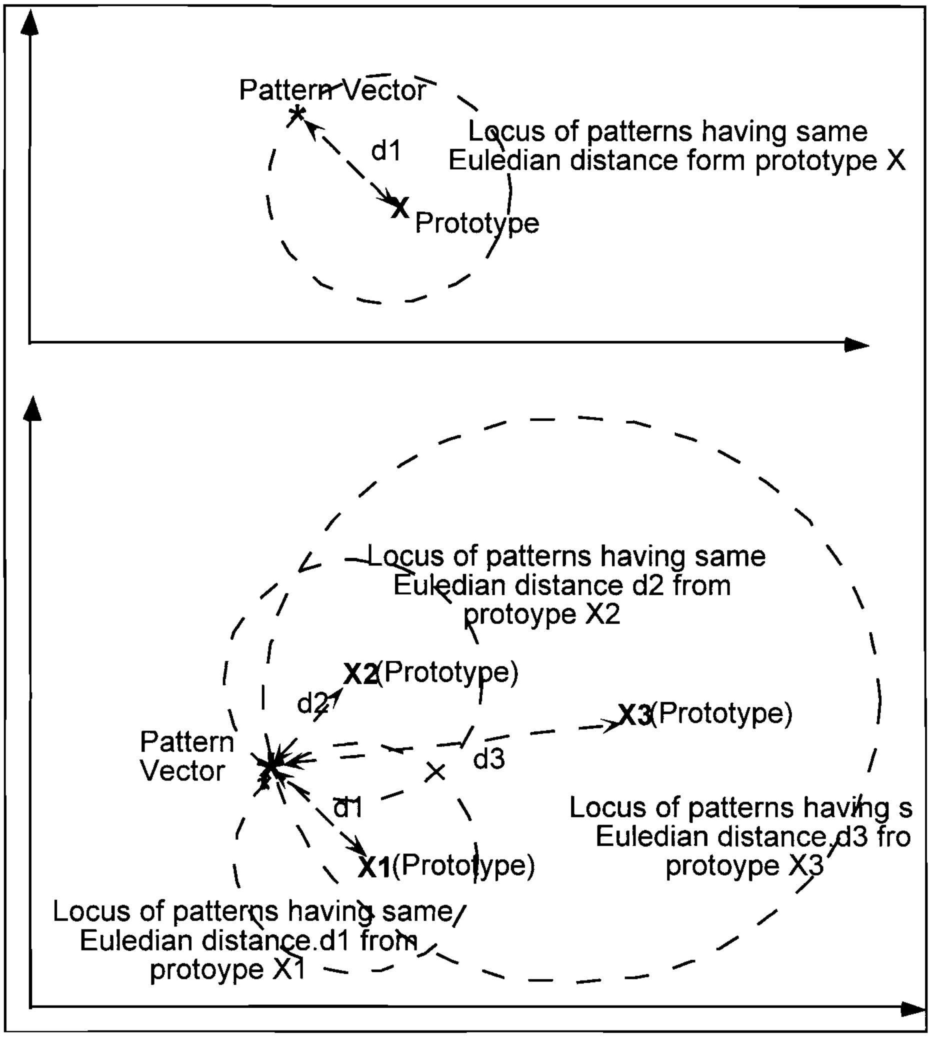


Figure 3.3: Identification of the process state using Euclidean norm.

3.3.1 Use of Sliding Window for On-Line Process Characterization

A sliding window is used to extract the most recent pattern of sensor data from the process. This is presented schematically in Figure 3.4. The sensor data is normalized and represented in a column vector form as discussed earlier. If the pattern is identical to one of the prototypes, then the corresponding values of the controller parameters are used. If the pattern vector falls between prototypes, then the process is in transition and evolving to a new steady state. In such cases the controller parameters to be scheduled must be interpolated.

3.3.2 Use of Time Smoothened Sensor Patterns

The gain map is formed from a number of nominal operating points. These nominal points typically correspond to steady state operations. In short, this method uses steady state process trend information as a basis for interpolation during transient periods. We propose to use time smoothening to extract the fundamental trend of a process variable during periods of transition. The value of a sensor is calculated based on an arithmetic mean of the previous sensor values. This can be considered as a compressed pattern information. Effect of time smoothening on a wildly fluctuating sensor pattern to extract its trend is shown in Figure 3.5. This is an illustrative example. The effect of time smoothening is also shown for the demonstration system in Chapter V.

The strength of our proposed technique is the ability to handle transient conditions. This method has been developed to match the performance of a skilled operator in recognizing the process state. An incorrect interpretation of a process trend data can give

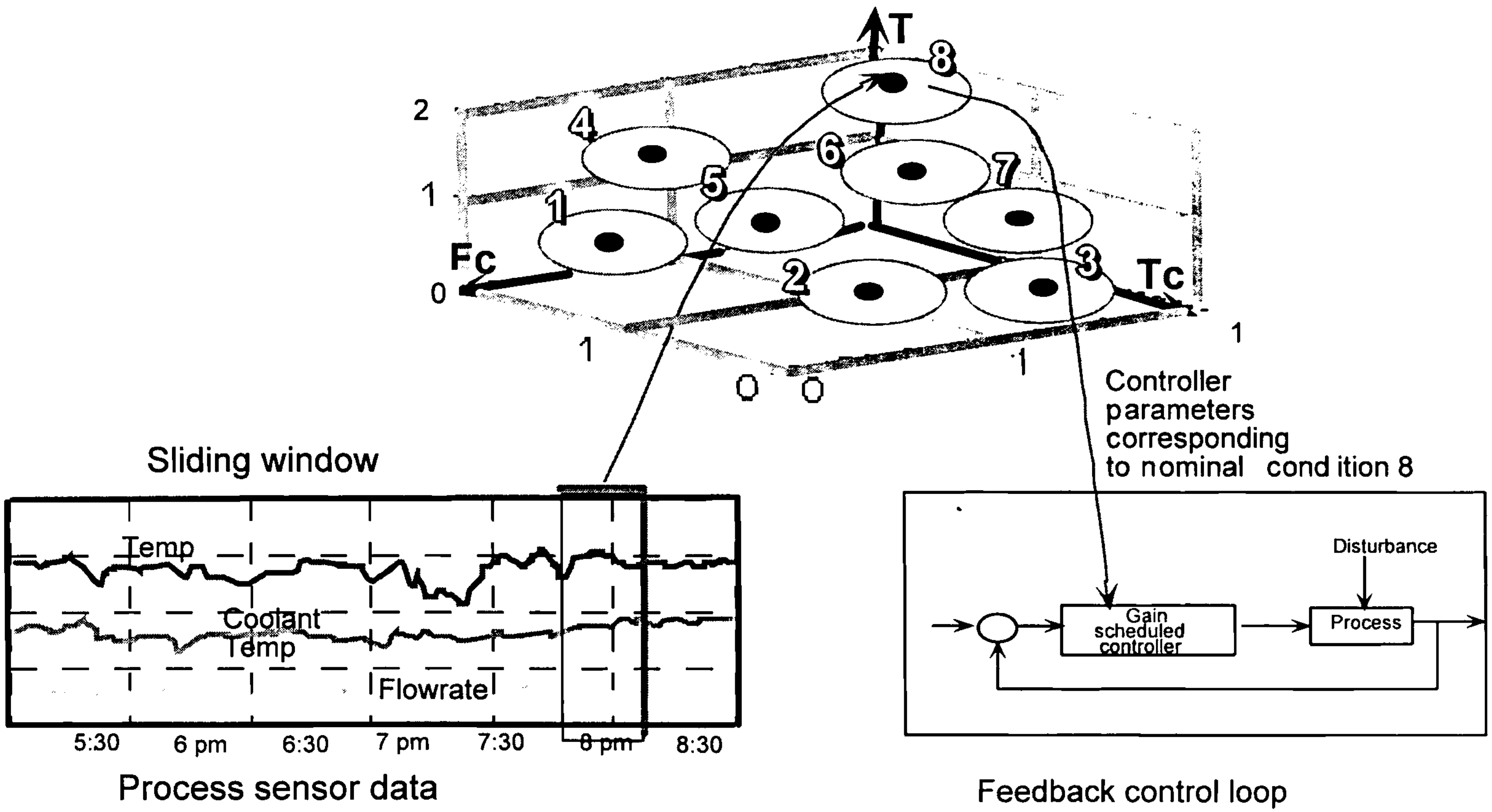


Figure 3.4: Use of a sliding window for establishing the state of the process.

rise to wrong parameters being scheduled and may result in an unstable response which seriously jeopardizes the safe operation of the plant.

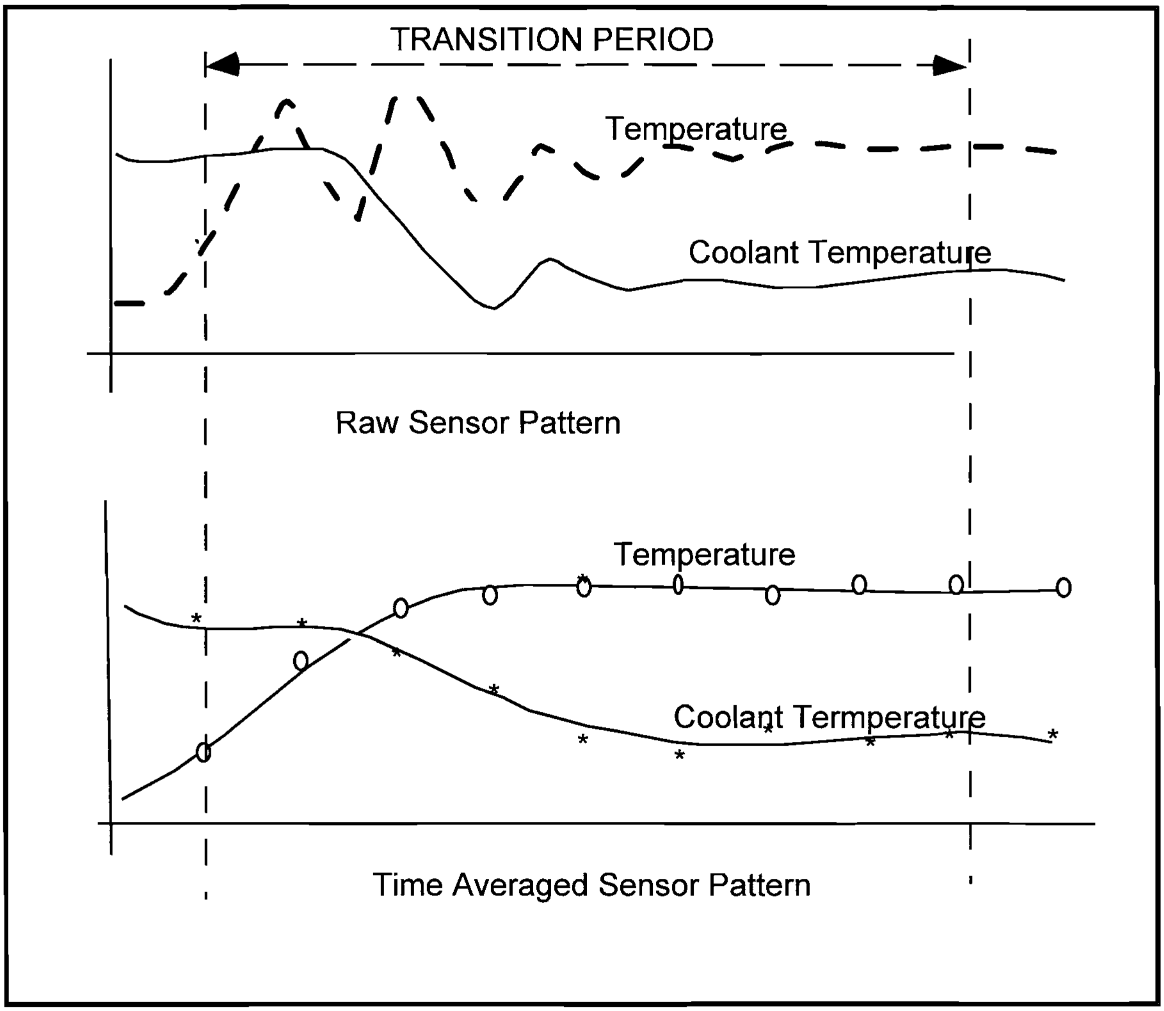


Figure 3.5: Effect of time smoothening to extract trend change

3.4 Concluding Remarks

This chapter describes our pattern-based method to characterize the state of a process. The process dynamics are captured by windowed slices of multi-sensor trend data. A sliding window is used to extract sensor data which is represented as a pattern vector. The current state is compared to any nominal state by measuring the similarity between pattern vectors. This approach provides an alternative to accurately characterize the state of the process even during periods of transition. Due to normal fluctuations in process variables during transition periods, it is not possible to measure process gain (K_P) or look at error diagnostics to decipher the process state. Thus the methods relating controller parameters to process gain or characteristics of error pattern (described in Chapter II) fail during transition periods.

The objective of a gain scheduler is to determine the process state and then schedule the controller parameters to maintain stable and satisfactory control action. The information regarding the distance between the current process state and its neighboring nominal conditions needs to be quantified in the form of a scheduled controller parameter. An interpolation technique which can perform this task is discussed in the next chapter.

CHAPTER IV

Controller Parameter Calculation

4.1 Overview

The final objective of a gain scheduler is to calculate the controller parameters as the process moves from one operating state to another. Different strategies to perform controller parameter calculations after the state of the process has been established is discussed in this chapter. The best way to calculate controller parameters is to approximate the gain surface by some continuous function of the process variables or patterns of process variables. Gain surface, as defined in Chapter 3, is a plot of controller parameters for all possible operating conditions. Arriving at a gain surface would require a good analytical model of the process and the controller parameter values at all possible operating conditions. One reason this is not possible is due to the presence of process uncertainties. Moreover it is not practical to establish controller parameters at all possible operating conditions. Thus there is a need to schedule the parameters based on some approximation of this gain surface.

4.2 Scheduling About a Few Nominal Points

We propose to do the gain scheduling about a few nominal operating points. This is based on the premise that some sort of interpolation between these operating conditions can be used to approximate the "gain surface." The interpolation strategies discussed in this chapter use the pattern-based state map, discussed in the previous chapter, to characterize the process dynamics. The process characterization is interpreted from the similarity measure of the current pattern vectors when compared to the nominal conditions. The scheduling or interpolation needed to establish the controller parameters is described in the following sections.

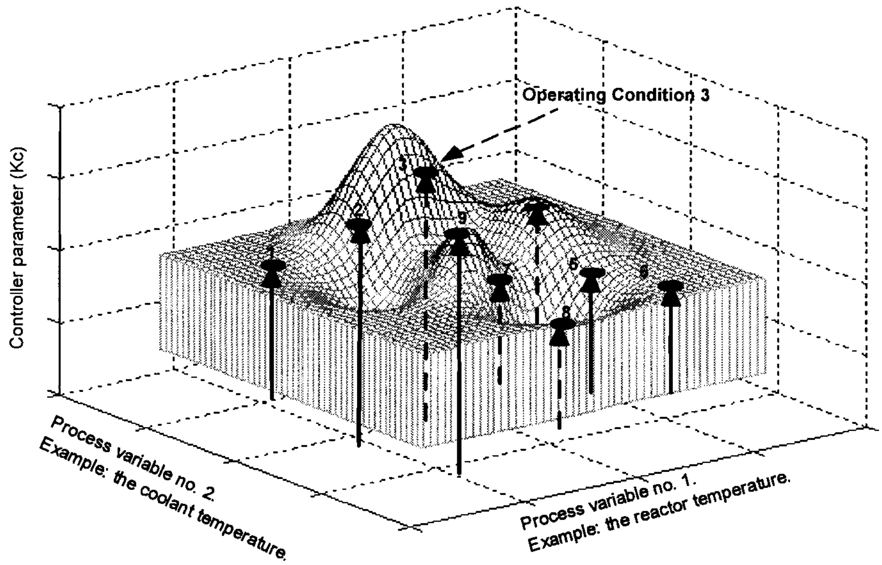
Our gain scheduling approach employs a set of linear stable controllers designed for different operating conditions. Our gain scheduled controller is thus a linear parameter varying (LPV) system. Earlier work done by Shamma (1993) shows that for such LPV systems a smooth transition in parameters is essential if one expects the properties of the linear controllers to carry over to the gain scheduled control system. No guarantee on overall stability of the system can be made (Shamma, 1990) while the process is in transition. Based on Shamma's theoretical analysis, one heuristic to govern interpolation has emerged. This heuristic states that even when rapid variations in plant parameters are present the scheduling should be performed slowly. Moreover gain scheduling is performed in a feedforward manner and there is no feedback for a wrong schedule. Thus, there is a need to develop a good strategy to interpret information regarding the state of the process to schedule the controller parameters.

4.3 Gain Surface Approximation

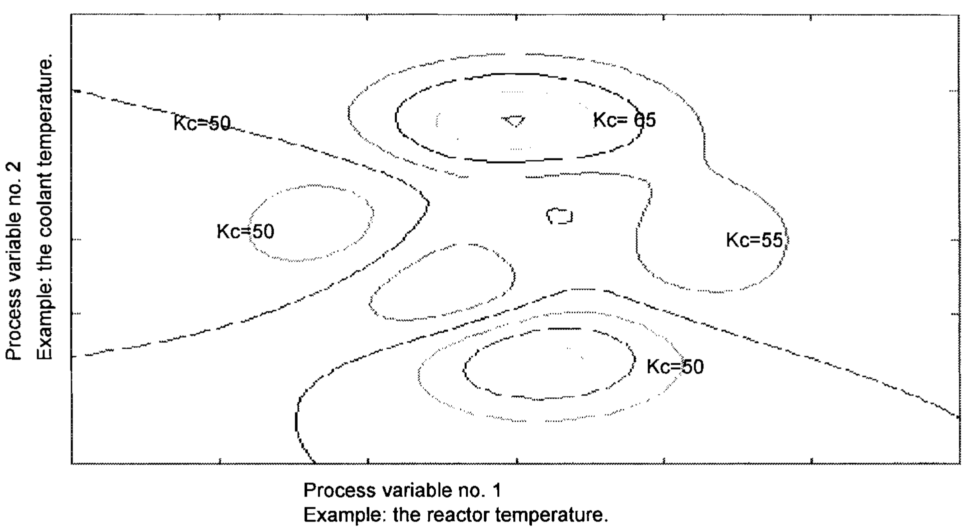
Gain surface is a plot of the controller parameters for all possible operating conditions. Gain scheduling is a relatively straightforward exercise if a function to map operating conditions onto controller parameters is available. In such a case once the process state is determined the controller parameters could be directly be read off the gain surface plot. This is practically impossible since controller parameters at all possible operating conditions are never known.

We are addressing this problem by approximating the gain surface about a few nominal conditions. In Figure 4.1 a hypothetical plot of a gain surface is presented. This plot has been generated using a continuous mathematical function and should not be confused with the gain surface of the demonstration system used in this study. Moreover this is a very simplified representation since a gain surface is unlikely to be in a three dimensional space. Also shown are a few operating conditions at which the controller parameters are empirically known. The aim of an interpolation strategy is to approximate the gain surface from the controller values at a few nominal conditions.

Once the process has been characterized at a few nominal operating conditions, a gain map can be constructed. This gain map relates the patterns of process variables, typically seen at these operating conditions, to the controller settings that needs to be used at these conditions. Such a representation of nominal conditions is also shown in Figure 4.1.



a : An illustration showing a possible gain surface and location of nominal conditions used to approximate it.



b. Contours of conditions having same controller settings.

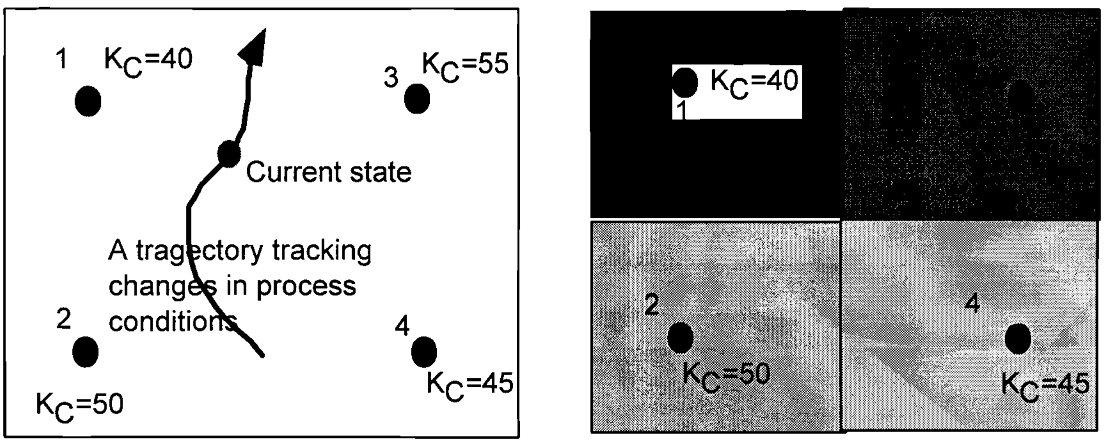
Figure 4.1: An illustration of a possible gain surface which is to be approximated about a few nominal conditions.

How close the gain surface is approximated depends on the number of nominal conditions used to approximate it. The curvatures in a gain surface occurs because of nonlinearities in the process. The extent of the nonlinearity governs the number of nominal conditions required to approximate the gain surface. For a linear process, the gain surface can be represented by a flat plane of constant controller parameters for all operating conditions. The next implementation issue is the choice of the number of neighbors or the prototypes to be uses for interpolation.

4.4 Choice of Number of Closest Neighbors

Each of the nominal conditions is associated with a set of controller parameters (K_C, τ_I, τ_D). When the process state is the same as one of the nominal conditions, then the controller settings corresponding to that operating condition are used. Interpolation is required when a process state is different from any of the nominal conditions. In such a case, the number of nominal conditions used to establish the scheduled controller setting plays is a critical issue.

The effect of different numbers of prototypes used for interpolation is illustrated in the Figure 4.2. Imagine a gain surface approximated by four nominal operating points. If only the closest neighbor is used to establish the controller parameter, then the gain surface can be represented as shown in Figure 4.2b. Each shaded region corresponds to the settings associated with the nominal condition that is enclosed in that region. Thus the gain surface can be viewed as made of four flat planes. Switching between these settings will occur whenever the process trajectory goes through more than one region. This can lead to very rapid variations in controller settings during periods of transition.



a Again map showing four nominal conditions.

b The gain map can be approximated as 4 different sectors if only one of the closest neighbor is used interpolate

Figure 4.2: Interpolation using a gain map formed using four nominal conditions.

The effect of different numbers of closest neighbors used for interpolation can be shown using a similar example. Figure 4.3 illustrates the characteristic of a gain trajectory with different number of operating points used. Gain trajectory is the record of changes in controller coefficients with changes in process conditions. When two or more closest neighbors are used to arrive at the final controller settings, then the final value is a

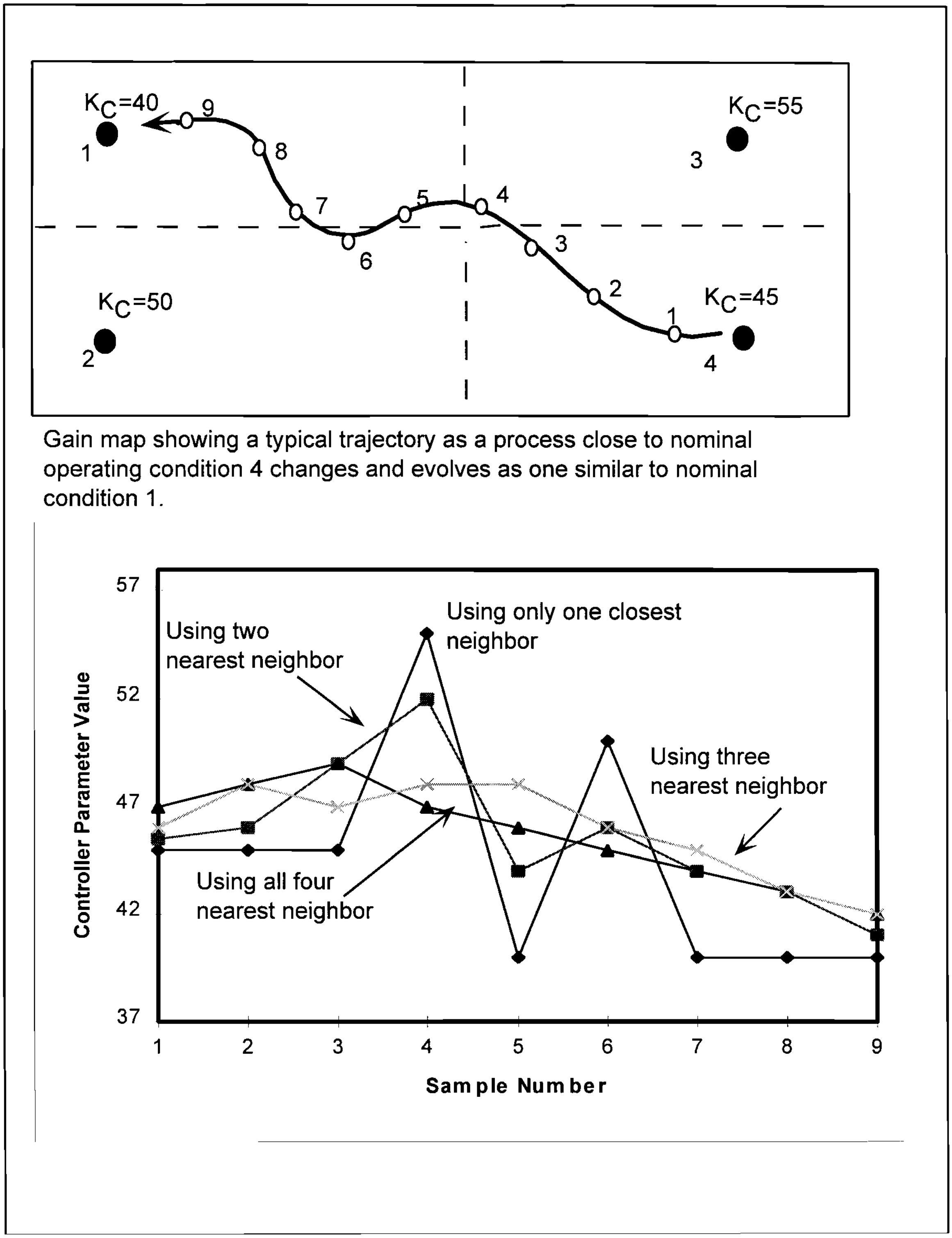


Figure 4.3: Effect of number of nominal conditions used for scheduling the controller parameter.

weighted average of the controller parameters of the nearest nominal conditions. As shown in Figure 4.2 the distance (d_i) is a measure of the distance between vectors in pattern space. The associated weights are inversely proportional to the distance.

The difference in gain trajectory when different number of closest neighbors are used is illustrated using the example shown in Figure 4.3. For this example an arbitrarily chosen gain trajectory is used. When only the closest neighbor is used lot of variations are seen depending on which sector the process state falls in. Similarly for the case using two neighbors the gain trajectory is jagged since pairs of nominal points may compete as the "closest two" while the process is in transition. For this example, all the nominal conditions should results in a smooth gain trajectory. This will increase the computational burden but is necessary if smooth transition in controller parameters is desired. Later in Chapter V the effect of using a different number of nominal conditions for interpolation for the test system is demonstrated.

4.5 Interpolation Methods

Different approaches to weight the similarity between pattern vectors can be used. How the similarity measured is interpreted is a critical issue in performing interpolation. In this study three different interpolation methods have been investigated. The gain scheduling is done using pattern-based gain map in all the three cases. Different pattern similarity measures to compare on-line pattern vectors to those representing nominal conditions can be used. The three different methods are:

1. Linear Interpolation Using Euclidean Distance.
2. Quadratic Interpolation Using Pattern Similarity Measured by ART2 Neural Network.
3. Fuzzy Interpolation.

The first two methods differ in the similarity measure used. The third method can be used for any similarity measure.

4.5.1 Linear Interpolation

Linear interpolation is based on the premise that the gain surface can be approximated by a hybrid surface made up of planes of constant slopes. The similarity between two pattern vectors is measured by the Euclidean norm between the two. Two perfectly similar patterns will have a Euclidean norm of zero. A schematic of linear interpolation is shown in Figure 4.5. Linear interpolation uses the lever rule to calculate the final controller parameter. Equation 4.1 is used to calculate the controller parameter.

$$\kappa = \frac{\sum_{j}^{n} (\kappa_i / d_i)}{\sum_{j}^{n} 1 / d_i} \qquad (4.1)$$

where: κ is the final controller parameter

κ_i = The controller parameters(e.g. K_c, τ_I or τ_D) associated with the i^{th} nominal condition.

$d_i = 1/(\delta)$ where δ is the Euclidean distance used to measure similarity between pattern vectors.

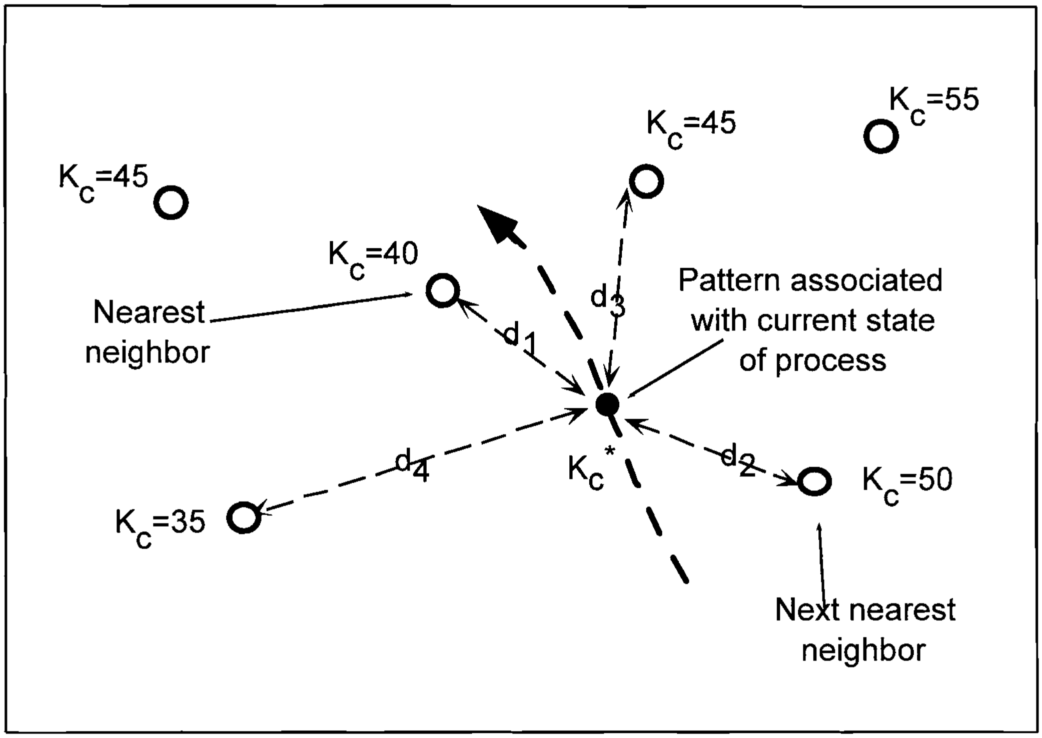


Figure 4.5: Schematic of linear interpolation.

For the case shown in Figure 4.5, the controller gain can be calculated as shown in Equation 4.2. In this illustrative example only four nearest neighbors are used to arrive at the final controller gain, K_C:

$$K_C = \frac{\frac{K_{c1}}{d_1} + \frac{K_{c2}}{d_2} + \frac{K_{c3}}{d_3} + \frac{K_{c4}}{d_4}}{\frac{1}{d_1} + \frac{1}{d_2} + \frac{1}{d_3} + \frac{1}{d_4}} \quad (4.2)$$

In the same manner the other controller parameters can be scheduled. The main advantage of linear interpolation is the ease of implementation.

4.5.2 Quadratic Interpolation Using Pattern Similarity Measured by ART2 Neural Network

The ART2 neural network is a modified version of an autonomous learning model based on Grossberg's adaptive resonance theory (Carpenter and Grossberg, 1987). Anderson (1993) previously modified the ART2 network to identify regions in pattern representation space which represent different operating conditions. In this approach the neural network is trained to recognize patterns previously seen at different nominal conditions. The nominal operating points are then represented by clusters rather than points.

Anderson's method to interpret the process state from a window of pattern information was developed on previous work by Whiteley and Davis (1993a, b). In this method a sliding window is used to continuously extract the most recent pattern of operation from the process. This pattern is input to the ART-2 network. If the pattern lies within the cluster of one of the nominal operating points, then the corresponding values of controller parameters are used. If the pattern lies outside the cluster then some sort of interpolation is applied. When a process pattern is completely identical to one of the nominal operating points the ART2 similarity value is 1.00.

ART2 based similarity measure is an alternative to Euclidean distance. This similarity measure depends on the distance as well as the orientation of patterns in the representation space. Addition of orientation improves the process characterization. Similarity between a process pattern and a nominal operating point is actually a quadratic function of the angle between the two (Whiteley et al., 1993).

The function used to find the quadratic distance between a process pattern and a prototype is :

$$d_i = \sqrt{1.000 - \rho} + \sqrt{\rho - S_i} \qquad (4.3)$$

where d_i is the distance from pattern to the cluster.

ρ is the radius of the pattern cluster, a user specified variable.

S_i is the ART2 similarity between the pattern and the i^{th} nominal operating point.

The motivation of using the ART2 based quadratic interpolation strategy is to utilize the orientation as well as the distance between the pattern and the prototype for calculating the similarity.

4.5.3 Fuzzy Interpolation

Fuzzy reasoning is associated with systems involving uncertainties. Fuzzy inference or fuzzy reasoning, often called approximate reasoning, has become a powerful technique for practical implementation in control system design (Zimmerman, 1991). We investigated fuzzy inference because of the uncertainties associated with controller settings. More often than not these settings are empirically based using a heuristic rule of thumb or trial and error tuning. There is rarely an exact set of tuning values associated with any process or operating condition. The use of "excellent," "good" or "bad" descriptions is often used to describe the efficacy of different controller parameter values. These linguistic variables (e.g. "excellent") can be represented by fuzzy sets. The controller parameters used at any nominal condition are also expressed as a fuzzy set. This is very similar to representing a nominal condition by a cluster in pattern representation space as described in the previous section. Figure 4.5 shows some typical fuzzy sets that can be used to represent a controller parameter for example controller gain K_C.

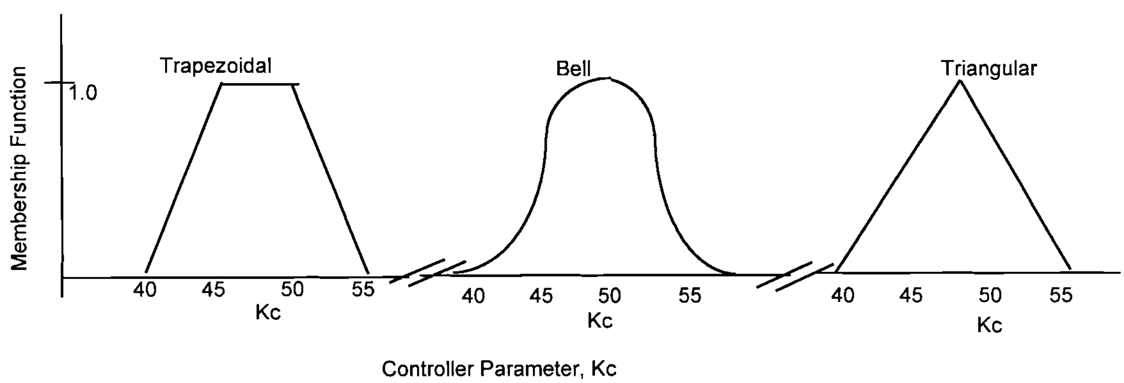


Figure 4.5: Some typical fuzzy sets to represent controller parameters.

Membership function (y axis) is a measure of the confidence associated with the controller parameters. In this work membership function is the degree of similarity between pattern vectors. For the first case shown in Figure 4.5, even when the pattern vectors are completely similar (membership function = 1.00) there is uncertainty in choice of gain value (45- 50). We have used triangular membership functions to accommodate the uncertainties in the controller parameters. The spread of which can be varied depending on how confident a designer is regarding the controller settings.

The controller settings at all operating conditions are thus represented by fuzzy sets. The degree of overlap is dependent on how different the controller settings are at these operating conditions and how much uncertainty is associated with them. During on-line implementation the process state is characterized using tools presented in Chapter III and the process pattern is compared with the patterns representing the nominal conditions. The similarity is normalized and scaled between 0 and 1. The similarity can now be inferred as a membership function. This is used to clip the fuzzy set of the controller parameter corresponding to the nominal condition as shown in Figure 4.6. Thus the process state is compared to each of the nominal operating conditions and its similarity is represented by the "clipped" fuzzy set. These clipped fuzzy sets can be used together to calculate the final controller setting. This is called "defuzzification." The number of clipped fuzzy sets used depends on the number of operating conditions chosen, as discussed in section 4.4.

In this study, the center of area (COA) defuzzification method (Driankov, 1990) has been used. Center of area looks at the region represented by clipped fuzzy sets and calculates its center of gravity. The controller parameter corresponding to the center of gravity of this fuzzy region is the final controller output. An example of this interpolation method is illustrated using Figure 4.7. In this example only the two closest neighbors are used to interpolate. It is clear that the current pattern is more similar to the operating condition "two."

The formula to compute a center of area is:

$$u^* = \frac{\sum_{i=1}^{k} u_i \times \mu_u(u_i)}{\sum_{i=1}^{k} \mu_u(u_i)} \tag{4.4}$$

where:

u^* = Final interpolated value of controller parameter.

μ_i = Normalized similarity value with respect to i^{th} nominal condition.

u_i = The controller setting at the i^{th} nominal condition.

Either the Euclidean norm or the ART2 similarity measures can be used to perform fuzzy interpolation. In this study we have implemented fuzzy interpolation using Euclidean norm as a measure of similarity. A sufficient number of nominal conditions

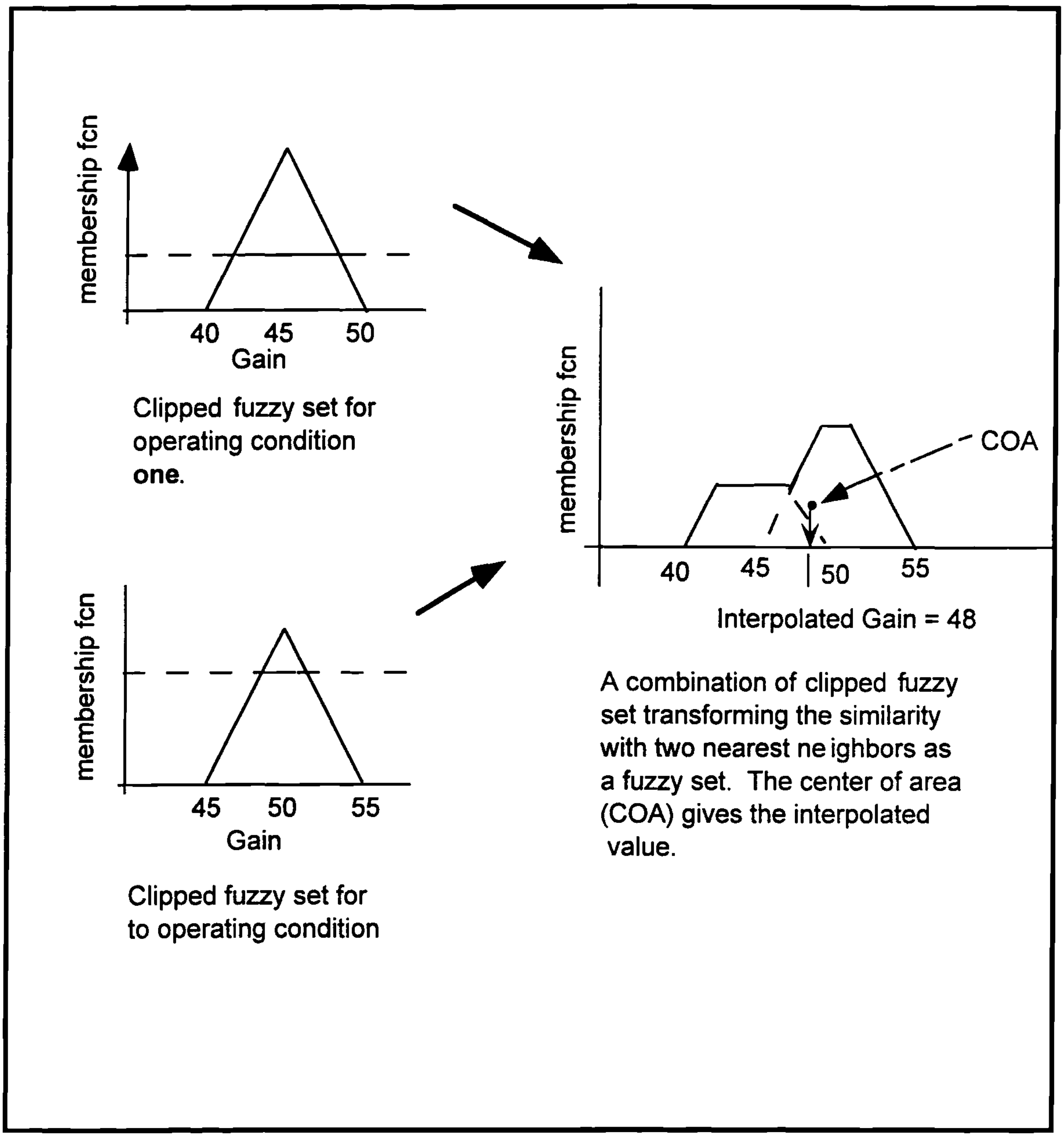


Figure 4.6: Schematic of fuzzy interpolation.

should be used so that all the fuzzy sets representing them overlap. This representation can thus result in a continuous approximation of the gain surface. This also gives a powerful tool to change the topology of the approximated gain surface by the use of different kinds of fuzzy sets and different defuzzification methods.

4.6 Concluding Remarks

This chapter presented different interpolation methods using a pattern-based tuning map. The methods presented aim at approximating the gain surface. Linear interpolation assumes that the gain surface can be represented as planes of constant slopes while the ART2 and fuzzy methods aim at representing them by piece wise smooth surfaces with variable topology. Each of the methods can be used to calculate the controller parameters in an on-line fashion. The controller parameters scheduled can never be labeled as "right" or "wrong". The performance of the gain scheduler can be inferred from the error trajectory of the controlled variable. All these three methods are implemented and investigated in Chapter V using a nonisothermal reactor as a demonstration system.

CHAPTER V

Demonstration of Pattern-Based Gain Scheduling

5.1 Overview

This chapter demonstrates our pattern-based gain scheduling approach for a simulated stirred tank reactor. The dynamics of the nonisothermal reactor are described in the next section. A pattern-based gain scheduled control methodology is then developed to control the reactor temperature. The third section presents results which we use to evaluate the performance of our gain scheduling approach. The importance of pattern-based information to characterize the state of the process is first presented. The effect of interpolation using a different number of nominal operating points is then shown. Finally, the performance of this gain scheduled controller at different operating conditions is demonstrated and compared with the performance of PI controllers having fixed controller parameters.

5.2 Nonisothermal Stirred Tank Reactor

The process used to demonstrate our approach exhibits dramatic changes in the process gain (K_P). The demonstration system is highly nonlinear and open loop unstable. A schematic of the nonisothermal jacketed CSTR is shown in Figure 5.1. The system here has been adapted from Luyben (1990).

The CSTR has one feed and one product stream and the control objective is to keep the reactor temperature at the desired set point by manipulating the flow rate of the coolant $F_c(t)$ in the outer jacket. A time delay of 30 seconds is used to compensate for imperfect mixing. A small amount of white noise is added to the temperature sensor reading to simulate measurement noise. Reactor parameters and the model equations appear in the Appendix.

The main source of nonlinearity is the exponential dependence of the reaction rate on the reactant temperature. This exponential function is popularly known as the Arrhenius equation. A phase plane analysis for the process dynamics reveals that the reaction "runs away" at various ranges of temperature. The process gain (K_P) and the controller gain (K_C) are of opposite signs in the temperature ranges where the reactor is open loop unstable. What makes open loop unstable systems particularly challenging is that the controllers can be "detuned" to only a certain extent to account for model uncertainties but must be tuned "tightly" enough to maintain closed-loop stability. Detuning often leads to an unstable response or results in system entering limit cycles in

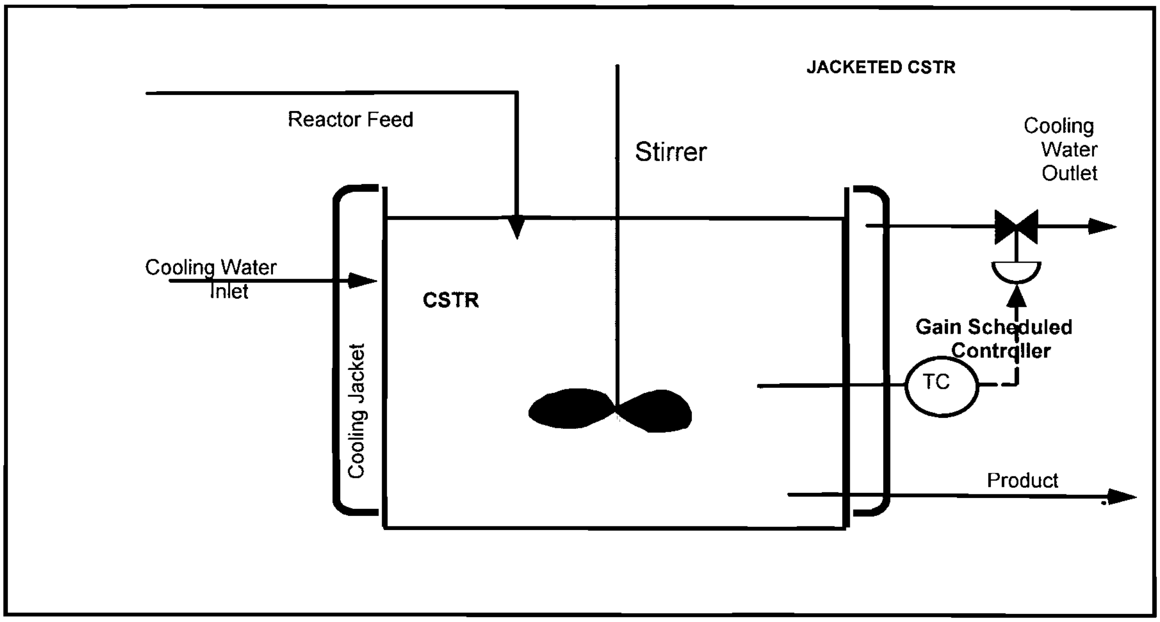
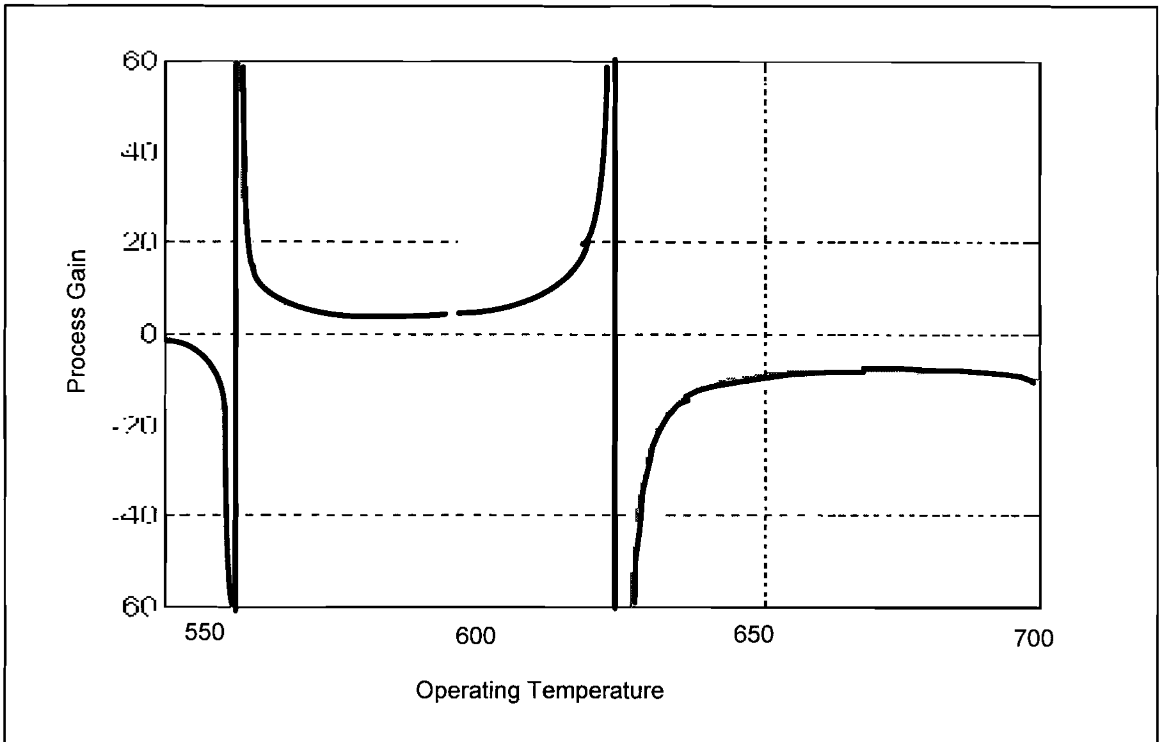


Figure 5.1: Schematic of the demonstration system.

the presence of saturators. Gain scheduling offers an excellent technique to control such processes.

A look at the process gain at different reactor temperatures is shown in the Figure 5.2. This system shows a wide variation in process gain. As shown in Figure 5.2, there are two temperatures at which the process gain switches from negative to positive infinity. The operating region between these two points is open loop unstable. This system offers proof that the knowledge of process gains alone is not enough to characterize the process dynamics. Details of a pattern-based gain scheduled control design for this CSTR is presented in the following section.



Note: For inlet reactant concentration = 1.0lb.mol A/ft^3.

Figure 5.2: Process gain at different operating temperatures.

5.3 Gain Scheduler Design

Design of the gain scheduler starts with formation of the pattern-based state map. First the key operating conditions are identified. Since this CSTR is a test system the conditions at which the reactor normally operates are decided arbitrarily. The nominal conditions are chosen such that the process dynamics vary from one condition to another. Reactor temperature and feed concentration are used to label or 'tag' each of these nominal operating conditions. Eight operating conditions are chosen as nominal operating conditions. These are listed in Table 5.1. The controller settings for each of these operating conditions have been arrived at by using continuous cycling approach (Zigler and Nichols, 1942). The parameters were then fine tuned using trial and error tuning (Jury 1973). Details of how the controller settings were established are documented in the Appendix.

Table 5.1
Tuning Parameters at Various Nominal Operating Conditions.

	Inlet Conc. = 0.6 lb.mole A/ft^3			Inlet Conc. = 1.0 lb.mole A/ft^3		
Tset = 560°R	1	Tset = 560°R T = 560°R Tc = 558°R Fc = 24.2ft^3/hr	K_C = -80 τ_I = 0.03	5	Tset = 560°R T = 560°R Tc = 557°R Fc = 80.0ft^3/hr	K_C = -120 τ_I = 0.04
Tset = 600°R	2	Tset = 600°R T = 600°R Tc = 595°R Fc = 49.9ft^3/hr	K_C = -60 τ_I = 0.05	6	Tset = 600°R T = 600°R Tc = 587°R Fc = 144ft^3/hr	K_C = -80 τ_I = 0.08
Tset = 640°R	3	Tset = 640°R T = 640°R Tc = 631°R Fc = 53.1ft^3/hr	K_C = -35 τ_I = 0.08	7	Tset = 640°R T = 640°R Tc = 618°R Fc = 153ft^3/hr	K_C = -50 τ_I = 0.12
Tset = 680°R	4	Tset = 680°R T = 680°R Tc = 671°R Fc = 40.0ft^3/hr	K_C = -30 τ_I = 0.15	8	Tset = 680°R T = 680°R Tc = 655°R Fc = 117.3ft^3/hr	K_C = -40 τ_I = 0.22

Note: K_C in ft^3/hr°R and τ_I in hr

The next step in the construction of a pattern-based gain map is to represent each of these nominal operating conditions as points in pattern representation space. The pattern vector was constructed from four scheduling variables. The scheduling variables used are the reactor temperature, coolant temperature and the coolant flowrate. The reactor temperature set point was used as the fourth scheduling variable. The setpoint is used to give additional weight to the operating condition where the process is headed. A six minute sliding window was used. The pattern vector is constructed from ten time samples for each variable. Thus the sampling frequency is 10/6 $min.^{-1}$ and each operating state is represented as a 40 dimensional vector in a pattern representation space.

Both Euclidean norm and similarity measure using ART2 neural network can be used for process characterization. Typically interpolation is done using the four closest neighbors. The controller can be programmed to use more nominal conditions for interpolation. For fuzzy interpolation triangular fuzzy sets are used to represent the controller settings.

In this way a gain scheduling algorithm is designed to control the demonstration system. The CSTR was simulated on Simulink by MathWorks Inc. as shown in Figure 5.3. Simulink is an extension of Matlab. The CSTR is modeled using four coupled ordinary differential equations(ODE). The simulation uses a 5^{th} order Runge-Kutta method to simultaneously integrate the four differential equations. Details of CSTR model development and simulation appears in the Appendix. This simulation can be changed to test different interpolation strategies. The performance of this gain scheduled design under different degrees of pattern information and different numbers of operating condition is investigated and presented in the next section.

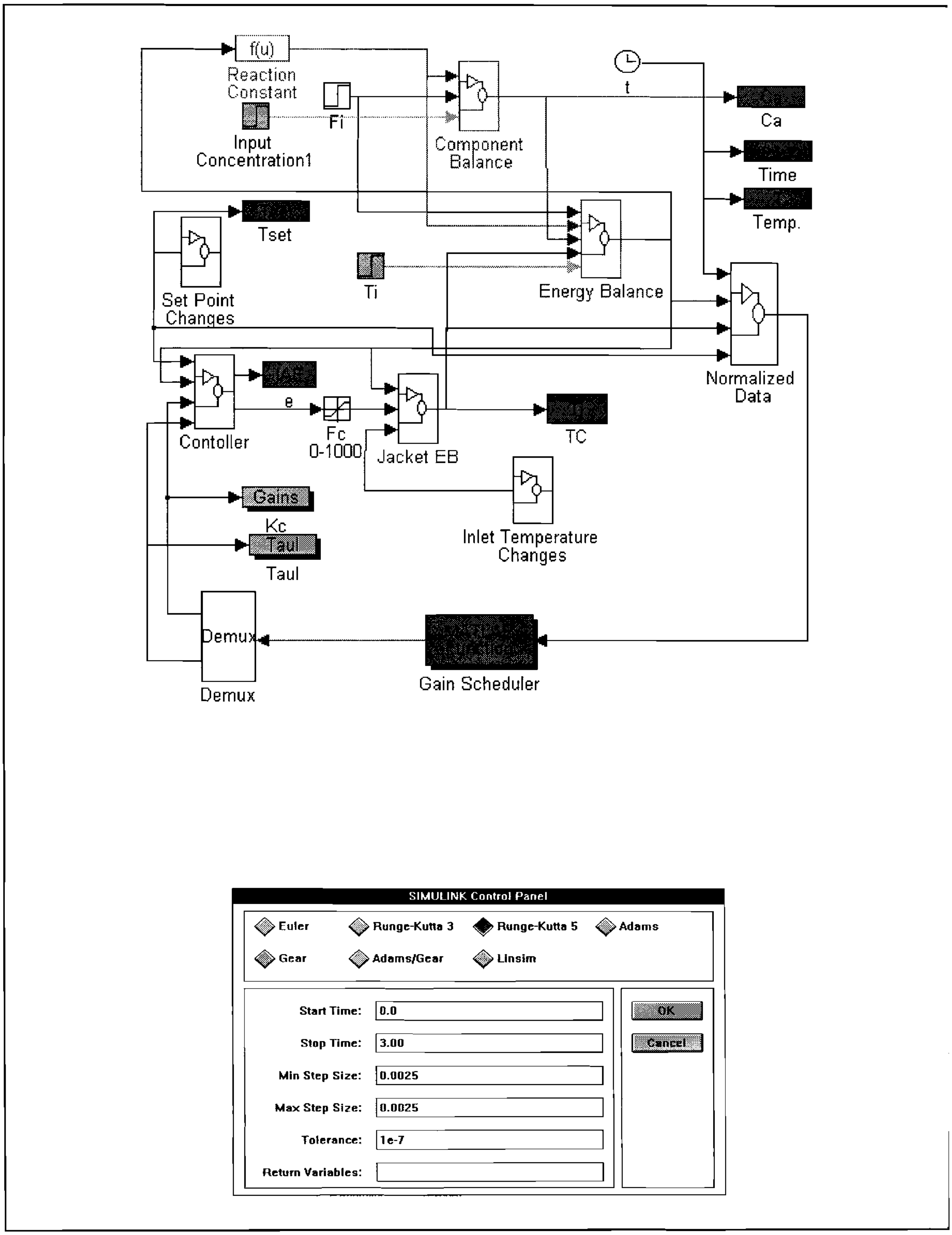


Figure 5.3: Simulink CSTR model with gain scheduled controller.

5.4 Results and Discussion

The performance of the gain scheduled controller is evaluated for different cases in this section. The aim is to identify the parameters that play a vital role in successful design of such a system. First the importance of pattern-based information is investigated. The next two sections aim at investigating how the gains should be scheduled once the process dynamics have been accurately characterized.

5.4.1 Importance of Pattern-Based Information

One simple way to perform gain scheduling is based on an instantaneous view of the process. In other words, no sliding window is used and raw sensor data are used to extract on-line process information. On the other extreme process state can be inferred from time smoothened pattern trends. This section explores the importance of pattern information for gain scheduling. The simulation system is controlled at various operating conditions using different degrees of pattern information. The importance of pattern-based information is inferred from controller performance during periods of transition.

The system is initially at the nominal condition of 640^{o}R temperature and perturbed with changes in set points. The Euclidean norm is used as a similarity measure and linear interpolation is used to arrive at the controller settings (K_C and τ_I). It is worth pointing out that the first and third setpoints are the nominal operating points while the second is not. Figure 5.4 shows performance of the gain scheduled controller using only the instantaneous view of the process to characterize the process state. In other words

CASE A	
Sensor Patterns:	Raw
Window:	Window length = 0
IAE:	2.32
Similarity Measure	Euclidean norm
Interpolation	Linear

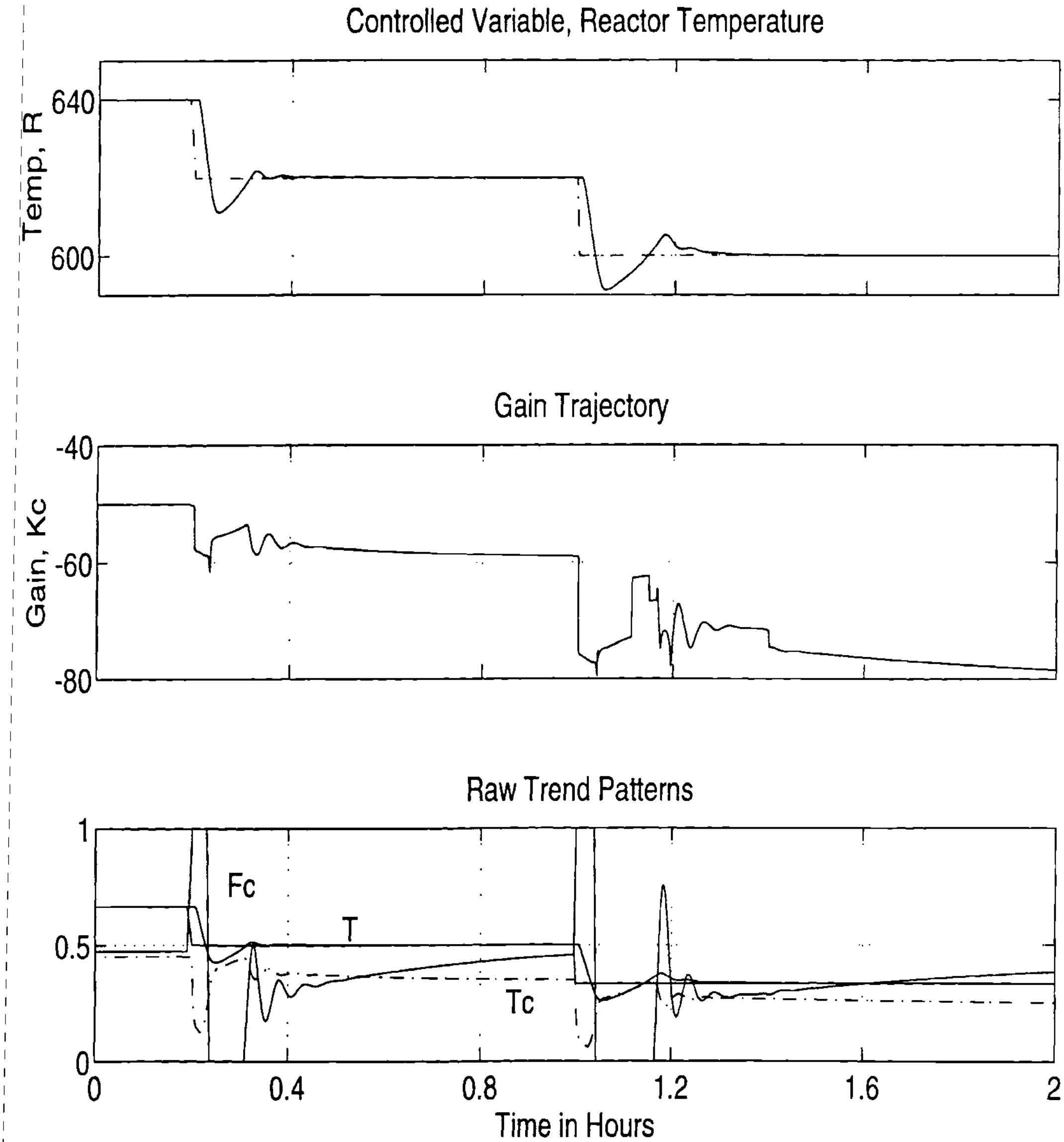


Figure 5.4: Gain scheduled control performance using instantaneous raw sensor values. The controller gain (K_c) picks up wrong values during periods of transition.

the window length is zero. Moreover the sensor readings are not smoothened. At the first step change, the controller settings rapidly change and the controller settings for a process state far from the desired plant operating condition is scheduled. As the fluctuations in process variables die out, the controller settings gradually settle down to the final value. This behavior is seen at both the set point changes. Such a gain schedule can be disastrous as it can schedule 'wrong' controller settings that could lead to an unstable or undesired response.

A time smoothening approach is used to process the sensor signals. Figure 5.5 illustrates how the performance of this controller improves by the use of time smoothened sensor patterns. The arithmetic mean of the past 10 sensor values is used to identify the state of the process. Here again, instantaneous values of the smooth sensor signals are used. Just a simple smoothening leads to large improvement in controller performance. The improvement in controller performance can be attributed to this smoothening technique. Time smoothening should not be confused as a signal processing technique to smooth noisy sensor data. It is actually a way to compress pattern information contained in a time period for which the signal is smoothened. Such smoothening results in a sensor trend which is less sensitive to rapid variations in process during periods of transition. Thus the trend change in the process dynamics can be effectively extracted from such smoothened pattern trends.

Another way in which use of pattern information can be implemented is by observing the process for a finite period of time. In this approach a sliding window is used to extract the features of sensor patterns. The width of this sliding window is approximately equal to the time it takes for the process under investigation to transform

CASE B	
Sensor Patterns:	Time smoothened
Sliding Window:	Window length = 0
IAE:	1.85
Similarity Measure:	Euclidean norm
Interpolation:	Linear

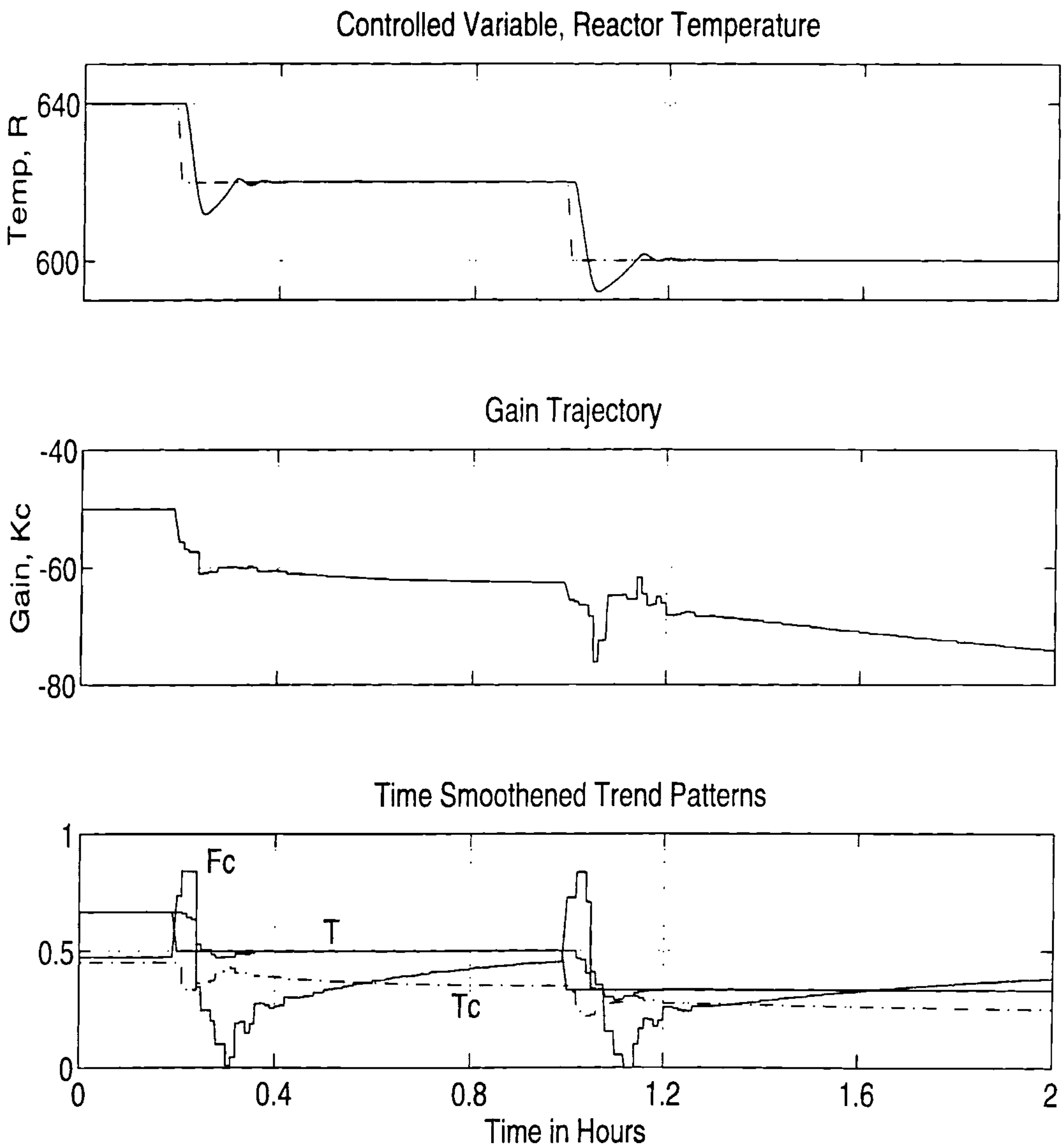


Figure 5.5: Gain scheduled control performance using time smoothened sensor values. The controller gain (K_c) fluctuates but not as much as shown in case A.

from one state to another in response to a step change. Here a six minute sliding window consisting of 10 time samples is used. As expected the gain trajectory smoothens as a result of better process characterization. The controller performance for this case is shown in Figure 5.6.

One result that emerges from these investigations is that use of pattern information is imperative for design of a gain scheduler. Finally a controller which extracts features of time averaged sensor patterns over a window length to characterize the process dynamics, is simulated. Its performance is shown in Figure 5.7. This controller uses a window length of six minutes and 10 sampled values of time smoothened sensor outputs.

The results of all the cases discussed so far in this section are tabulated in Table 5.2. The performance of a gain scheduled controller for different cases are compared by the IAE (integral of absolute error) value of the controller variable. IAE is one of the most popular indices to measure the performance of a controller. Table 5.2 also shows controller performance when the process characterization is done using ART2 based similarity measure. The variation in IAE values as more and more pattern information is used reemphasizes the importance of pattern information for gain scheduling.

CASE C	
Sensor Patterns:	Raw
Sliding Window:	6 minute sliding window
IAE:	1.80
Similarity Measure:	Euclidean norm
Interpolation:	Linear

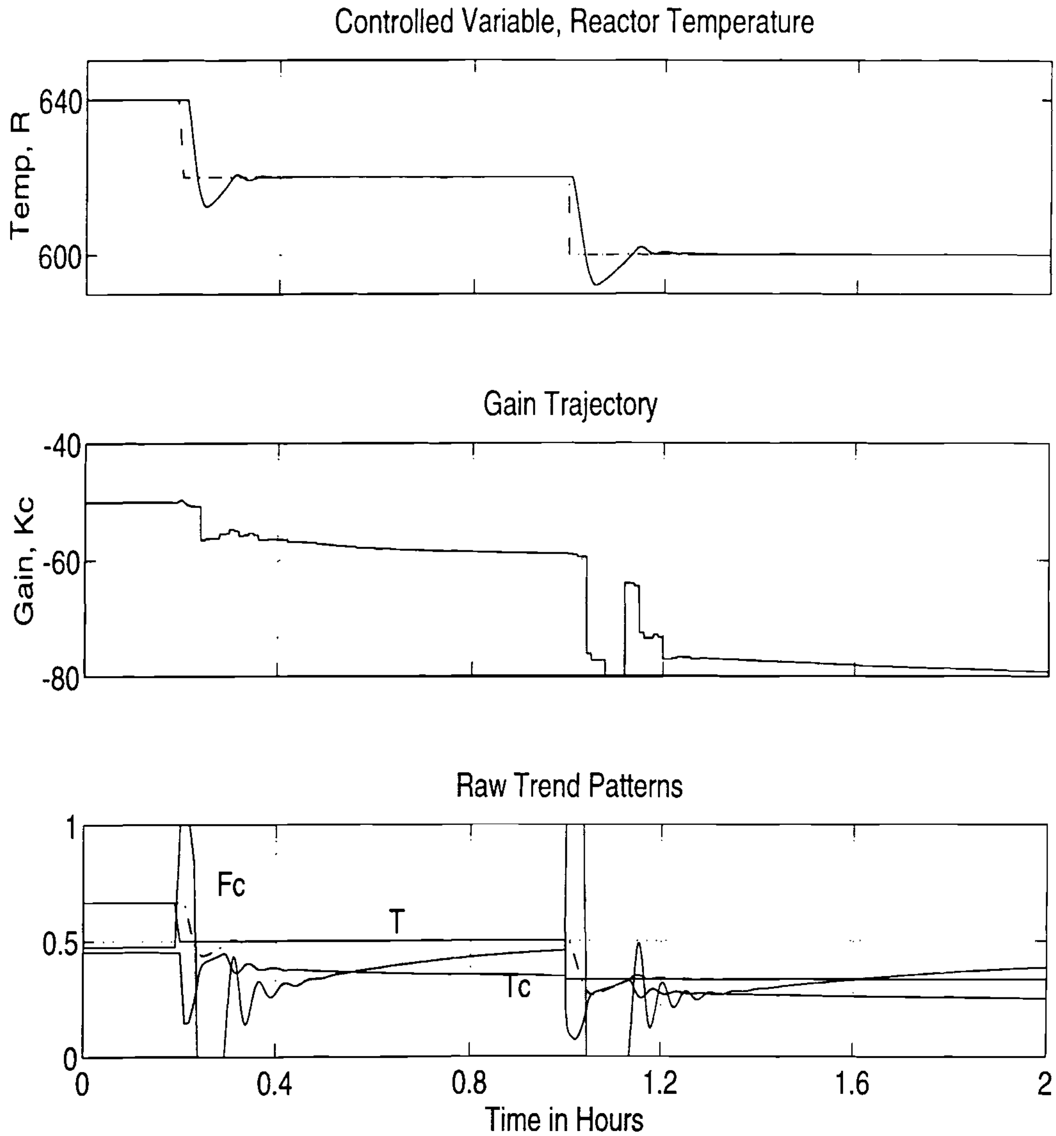


Figure 5.6: Gain scheduled control performance using raw sensor values and an on-line sliding window. The controller gain fluctuates but not as much as shown in case A.

CASE D	
Sensor Patterns:	Time smoothened.
Sliding Window:	6 minute sliding window
IAE:	1.80
Similarity Measure:	Euclidean norm
Interpolation:	Linear

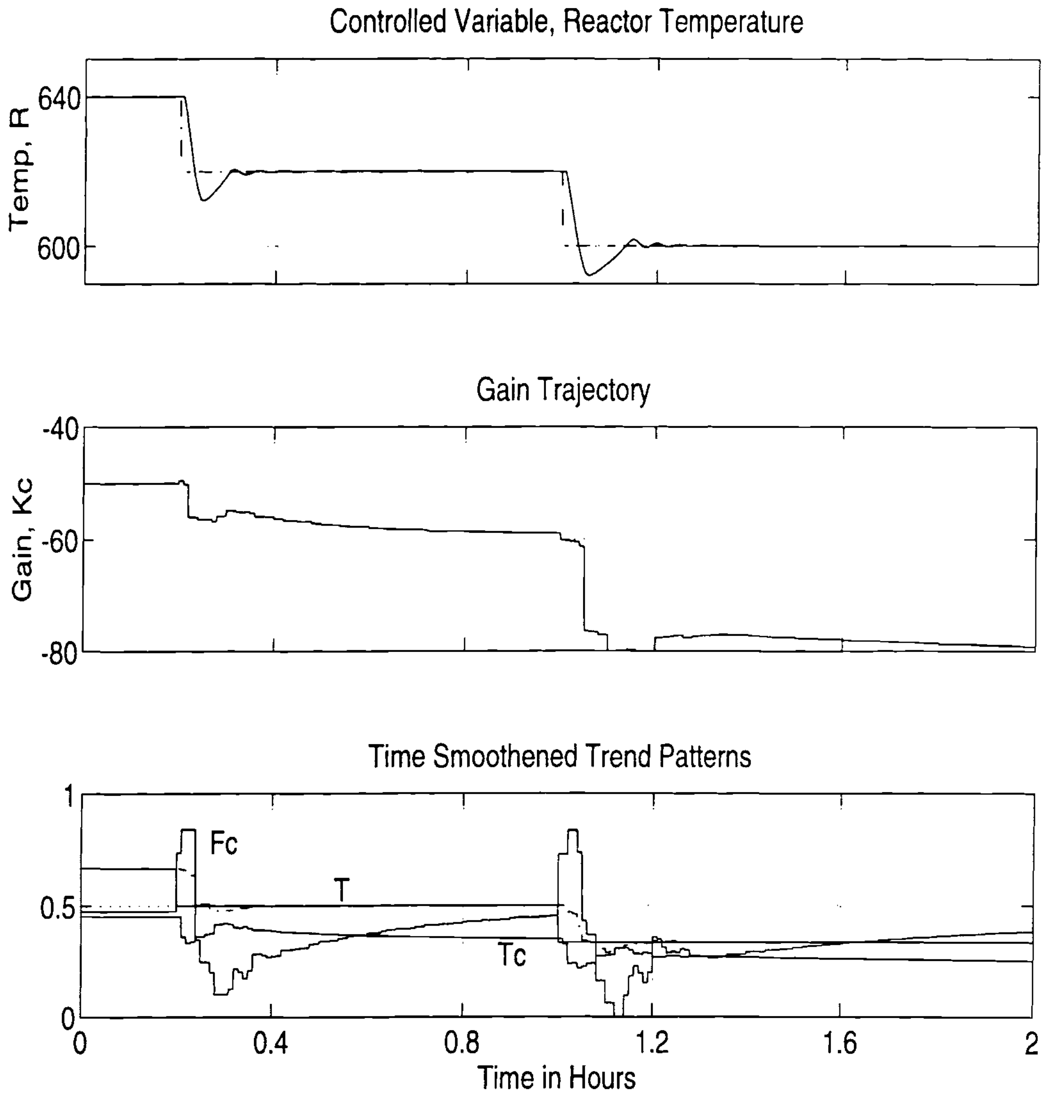


Figure 5.7: Gain scheduled control performance using time smoothened sensor values and a on-line sliding window. The controller gain settles down to its final value very fast.

Table 5.2

Importance of Pattern-Based Information for Gain Scheduling.

CASE	**Pattern-Based Information**	**IAE**
A	Raw sensor patterns. No window information. Linear interpolation:	2.32
B	Time smoothened sensor patterns. No window information. Linear interpolation.	1.85
C	Raw sensor patterns. Six minute finite window. Linear interpolation.	1.80
D	Time smoothened sensor patterns. Six minute finite window. Linear interpolation.	1.80

Note: Corresponding IAE values when ART2 similarity measure was used are 1.98, 1.81, 1.98 and 1.72 for cases A, B, C and D respectively.

5.4.2 Number of Nominal Conditions

In the previous section the controllers were scheduled using linear interpolation based on the four nearest neighbors. Chapter IV illustrated that gain scheduling performance varies depending on how many nearest neighbors are used for interpolation purposes. In this section, the effect of a different number of nominal conditions used to interpolate is shown. The pattern similarity is measured using Euclidean norm and the final controller settings are calculated using linear interpolation.

The number of nominal conditions used for interpolation is a key issue for successful implementation of this strategy. Controller performance with interpolation using 2, 4 and 7 prototypes are shown. In all cases, sensor patterns are time smoothened and a six minute sliding window is used. Euclidean norm is used as similarity measure.

Thus any difference in performance can entirely be attributed to the number of prototypes used for interpolation.

Figure 5.8 shows the controller performance when only the two nearest neighbors are used for interpolation. A look at the gain trajectory immediately reveals that during periods of transition the process state is not correctly identified and results in a wrong schedule. The process takes almost 25 minutes to settle down to its final value. This is because when the process is in transition it passes through regions where it could be distinctly closest to one prototype ('the closest neighbor') but equally closer to two or more prototypes ('the next closest neighbors'). As the process evolves its interpolation is constantly done using two or more pairs of prototypes. This results in a jagged gain trajectory.

Next, the four closest neighbors are used to arrive at the final controller setting. The resulting controller performance is shown in Figure 5.9. This means that the process state is observed from four different points in pattern space and all the four of the similarity measures are used to characterize the state of the process. This gives much smoother gain trajectory and a drastic improvement in controller performance. Using 7 prototypes further smoothens the gain trajectory (Figure 5.10). Also the IAE value is reduced by approximately 5 %. The effect of the number of prototypes used for interpolation is summarized in Table 5.3.

CASE E INTERPOLATION USING 2 PROTOTYPES	
Sensor Pattern:	Time smoothened
Window :	6 minute sliding window
Similarity Measure:	Euclidean norm
Interpolation	Linear
IAE:	3.14

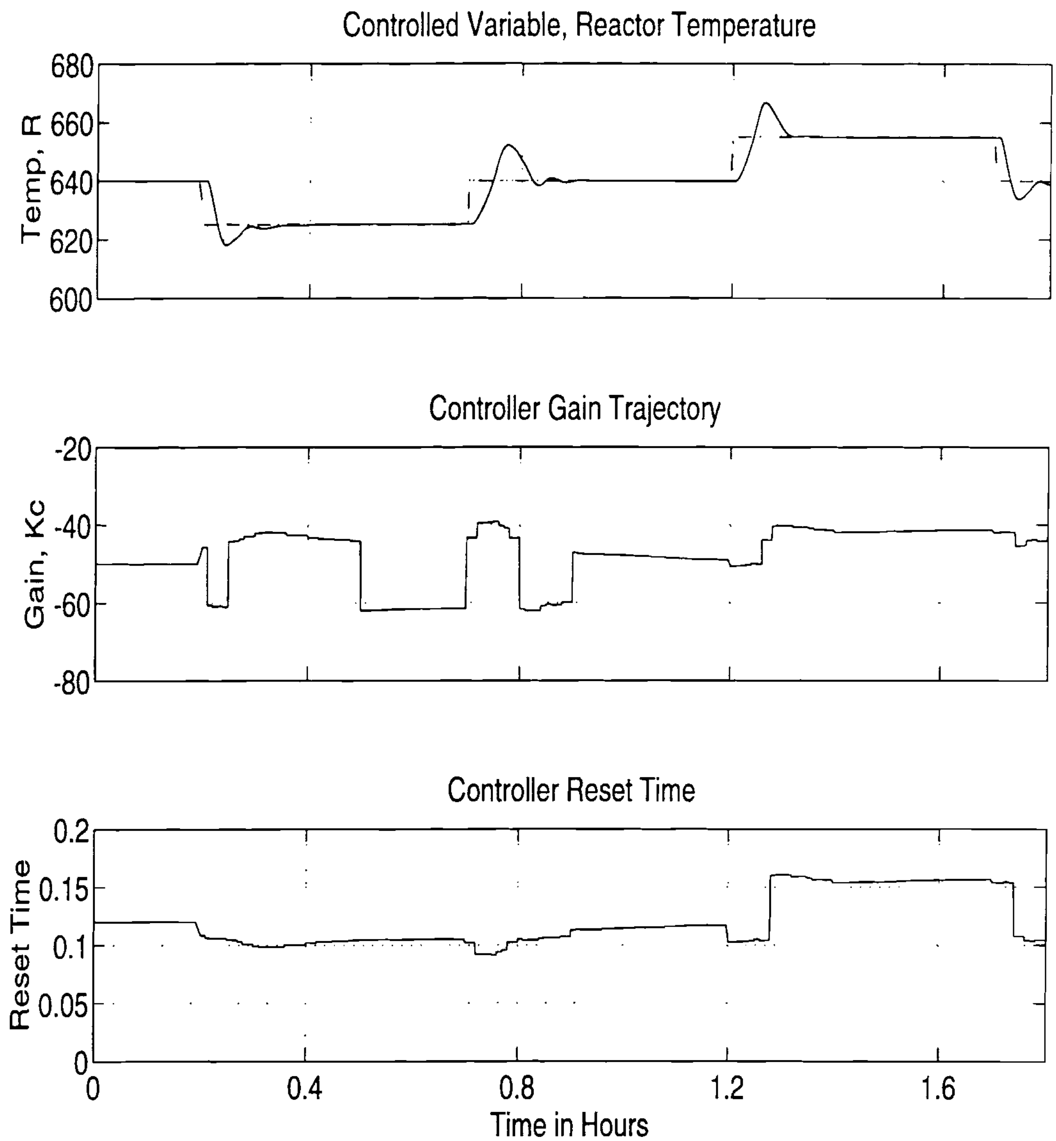


Figure 5.8: Controller performance with interpolation using only the 2 closest neighbors. Gain trajectory is very jagged indicating that transient pattern information was misinterpreted causing a wrong schedule.

CASE F INTERPOLATION USING 4 PROTOTYPES	
Sensor Pattern:	Time smoothened
Window :	6 minute sliding window
Similarity Measure	Euclidean norm
Interpolation:	Linear
IAE:	3.06

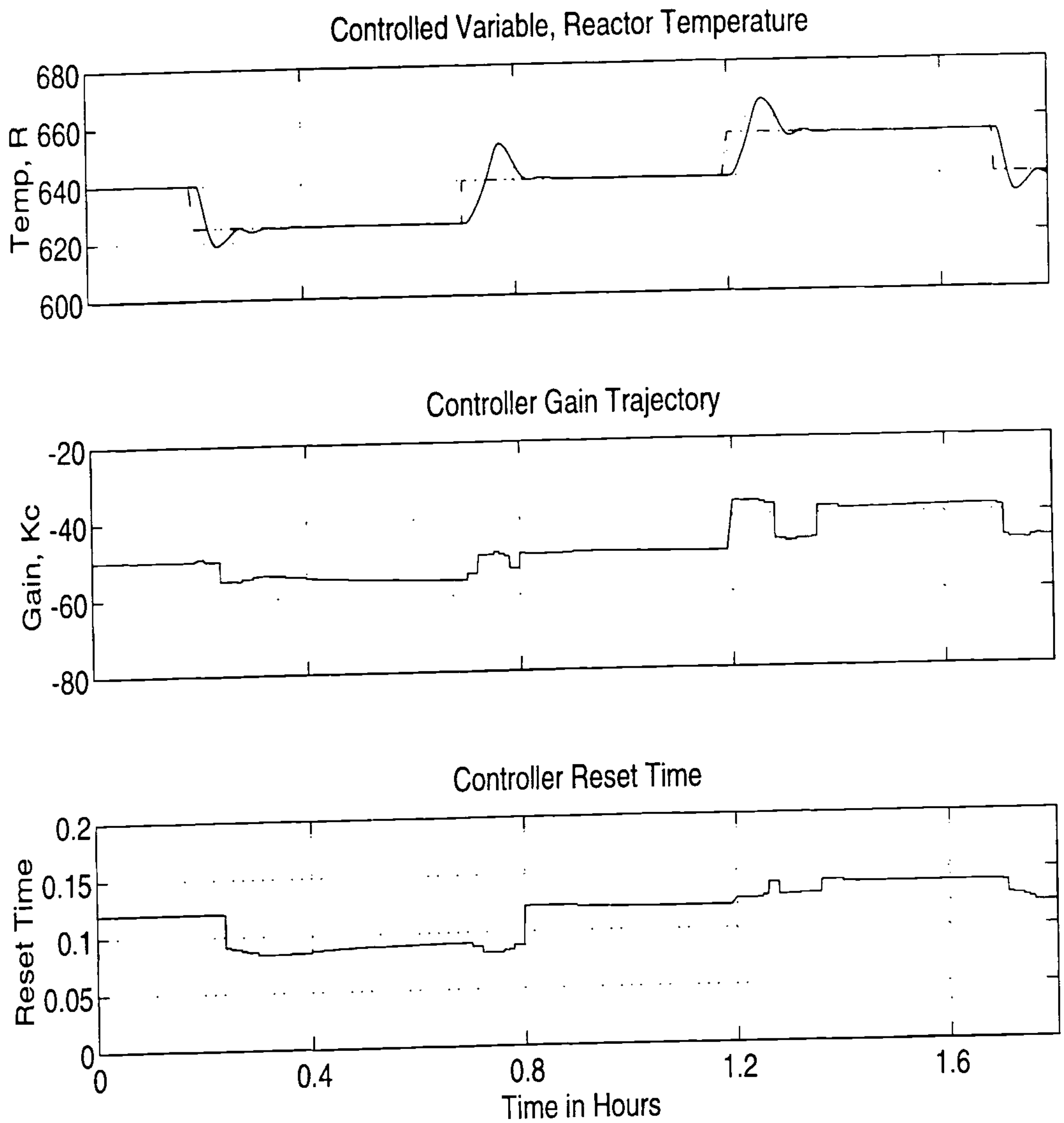


Figure 5.9: Controller performance with interpolation using only the 4 closest neighbors. Gain trajectory is quite smooth.

CASE G INTERPOLATION USING 7 PROTOTYPES	
Sensor Pattern:	Time smoothened
Window :	6 minute sliding window
Similarity Measure	Euclidean norm
Interpolation	Linear
IAE:	2.97

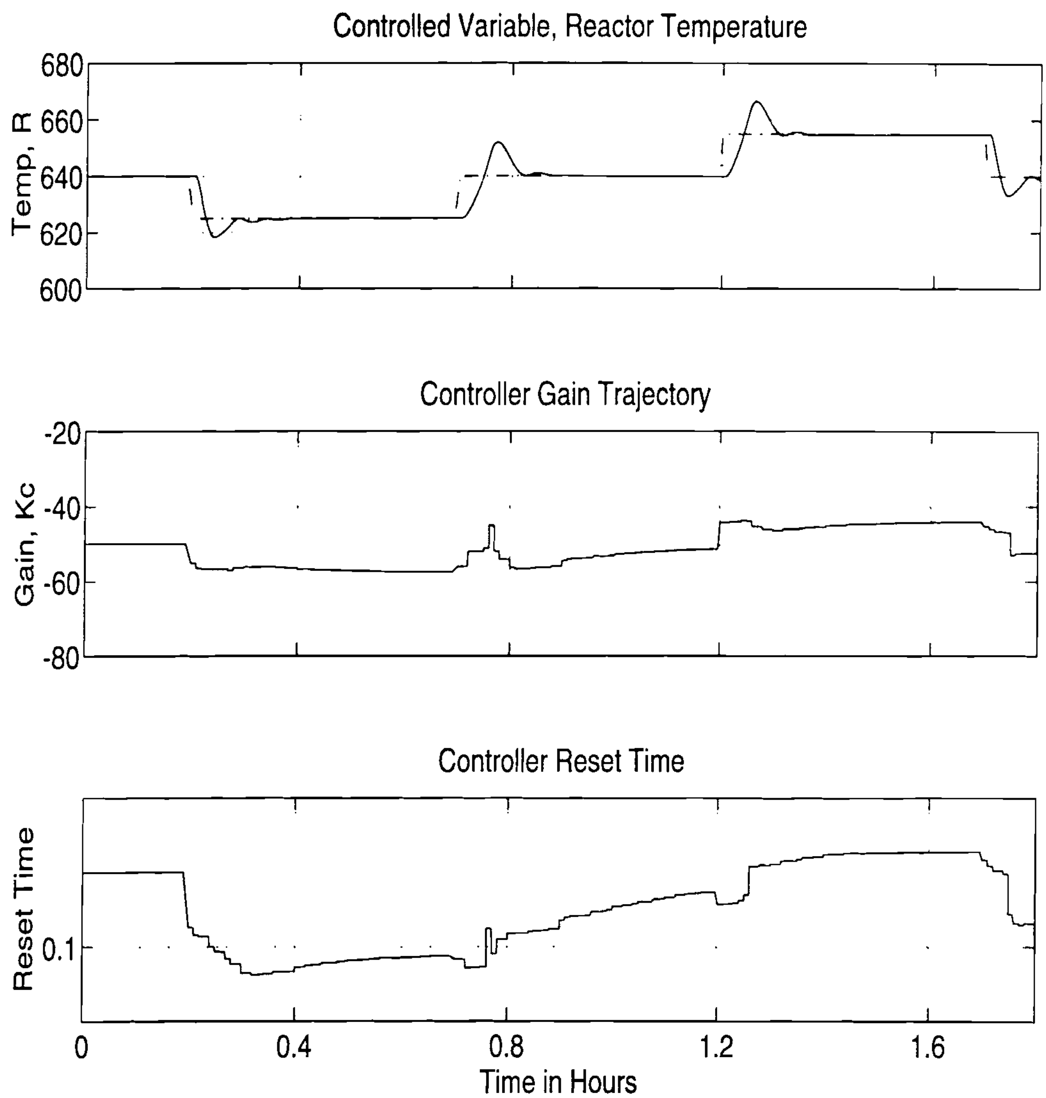


Figure 5.10: Controller performance with interpolation using 7 closest neighbors.

CASE G INTERPOLATION USING 7 PROTOTYPES	
Sensor Pattern:	Time smoothened
Window :	6 minute sliding window
Similarity Measure	Euclidean norm
Interpolation	Linear
IAE:	2.97

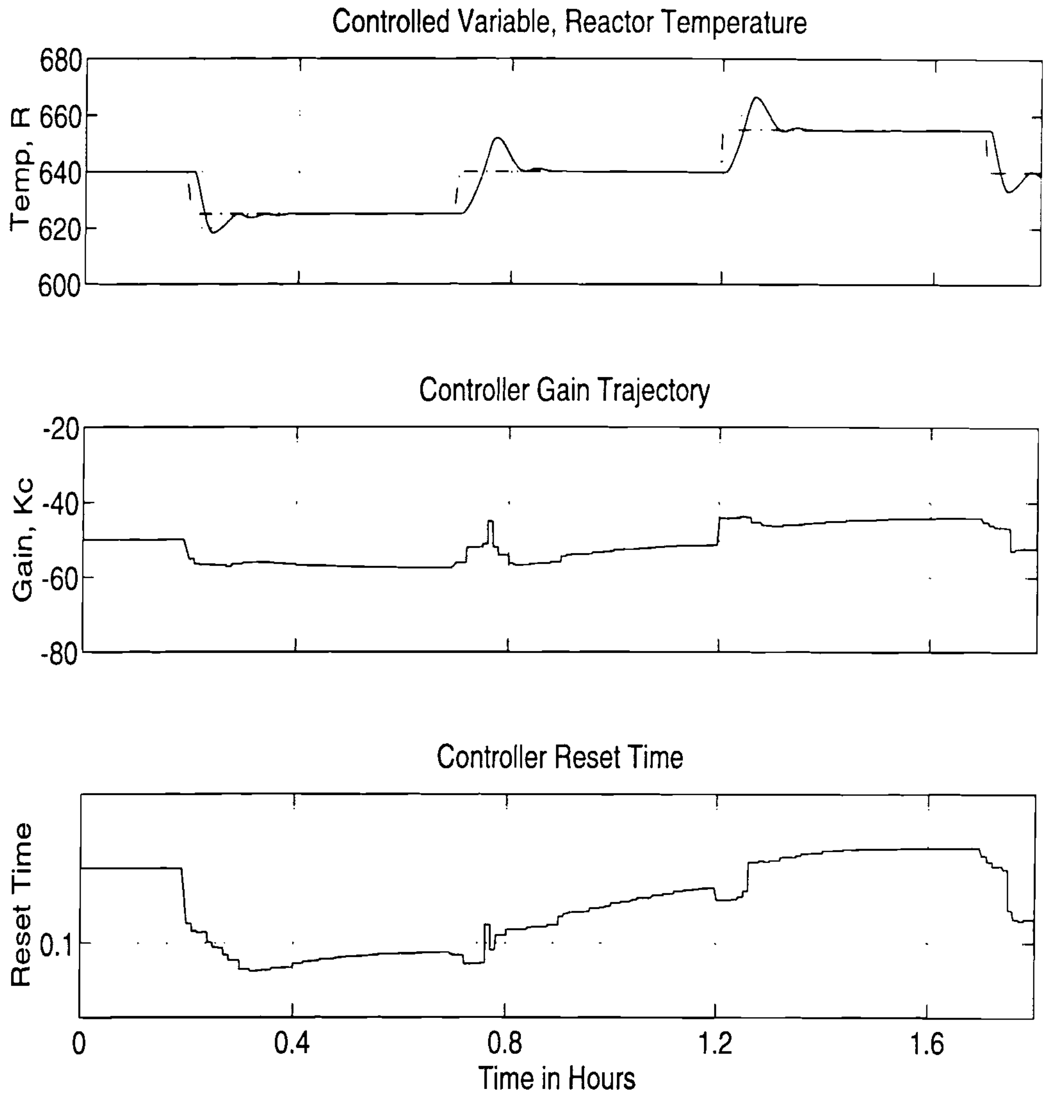


Figure 5.10: Controller performance with interpolation using 7 closest neighbors.

Table 5.3

Effect of Number of Operating Conditions on Interpolation

CASE	Number of Operating Points Considered for Interpolation	Qualitative Analysis of the Gain Trajectory	IAE
E	2	Gain trajectory is very jagged indicating that transient information is misinterpreted leading to a "wrong" schedule.	3.14
F	4	Gain trajectory is quite smooth and responds smoothly during transient periods.	3.06
G	7	Gain trajectory is very smooth and very smooth changes in controller settings are seen in presence of transient condition.	2.97

5.4.3 *Interpolation Strategy*

The results shown in the previous two subsections demonstrates the importance of pattern-based information and use of more than two nearest neighbors for gain scheduling. This subsection investigates performance of a gain scheduled controller using different interpolation strategies. The aim of an interpolation strategy is to approximate the gain surface. The trajectory of controller parameters as the process moves from one operating condition to another is used to evaluate how well the interpolation strategy approximates the gain surface. First, the system is simulated using linear interpolation, next a quadratic interpolation based on an ART2 similarity measure is simulated. Finally, the process is simulated using fuzzy interpolation.

The system is initially at the operating temperature of 640^{o}R and inlet concentration is 1.0 lb. mole/ft^3. Step changes are made in reactor temperature to increase the reactor temperature to a maximum possible temperature of 675^{o}R. Then the temperature is decreased thus covering the entire operating range. For all the simulations a six minute sliding window and a sampling frequency of 10/6 $min.^{-1}$ is used.

Controller performance using a Euclidean based process state identifier is shown in Figure 5.11 and with an ART-2 based identifier in Figure 5.12. Figures 5.11 and 5.12 show a comparison in controller performance for step changes in set point. It is worth noting that though the IAE are comparable, the gain trajectory is a bit smoother for the case in which an ART2 based similarity measure is used. This could be attributed to the fact that ART2 based similarity not only measures the distance between the pattern vectors but also takes into consideration its orientation in the pattern space.

CASE H LINEAR INTERPOLATION	
Sensor Pattern:	Time smoothened
Window	6 minute sliding window
Interpolation:	Linear using the 4 closest neighbors
Similarity Measure:	Euclidean norm
IAE:	5.31

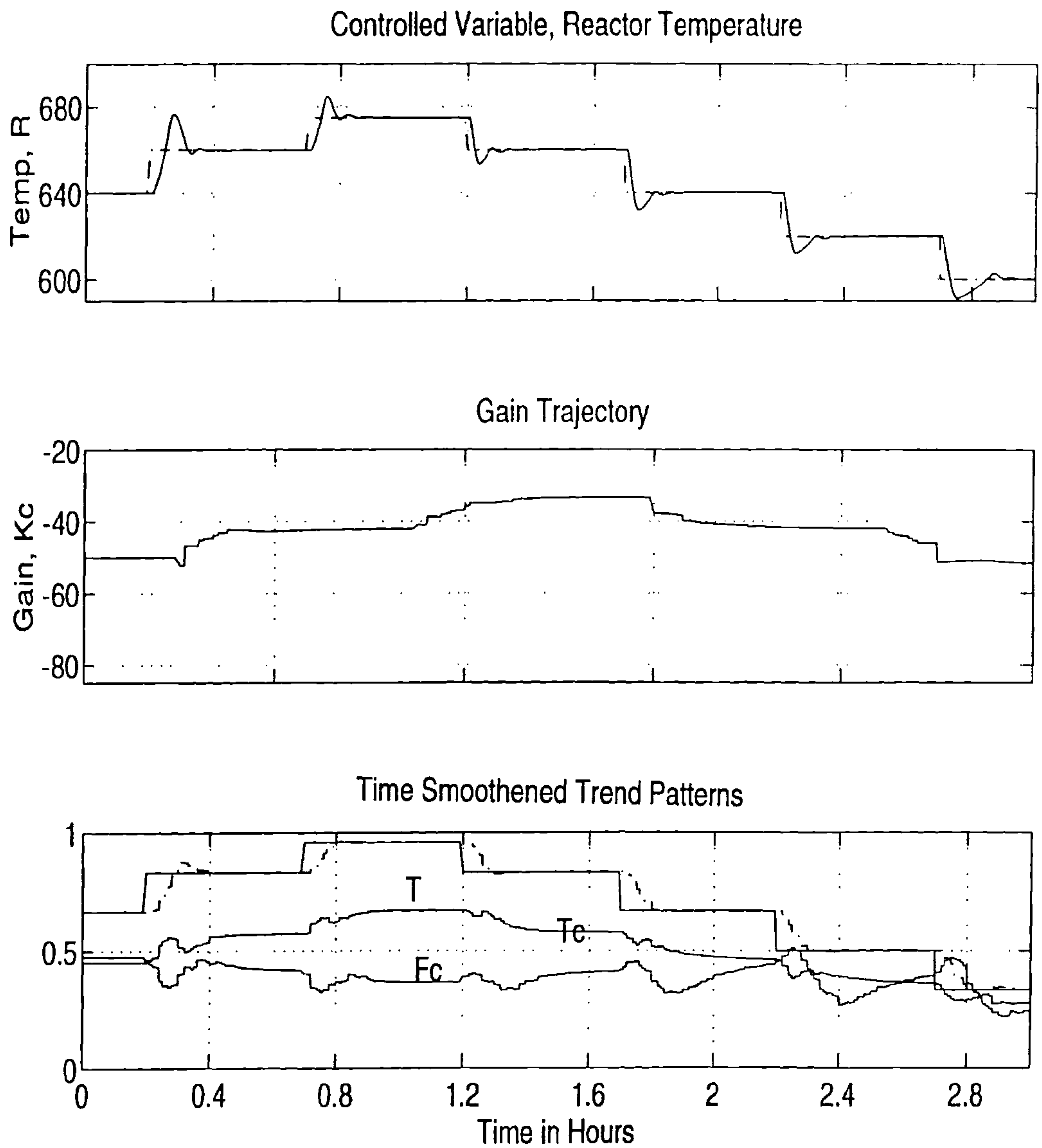


Figure: 5.11: Overall servo performance using Euclidean norm as a similarity measure.

CASE I INTERPOLATION USING ART2 SIMILARITY MEASURE	
Sensor Pattern:	Time smoothened
Window	6 minute sliding window
Interpolation:	Quadratic using the 4 closest neighbors
Similarity Measure:	ART2
IAE:	5.16

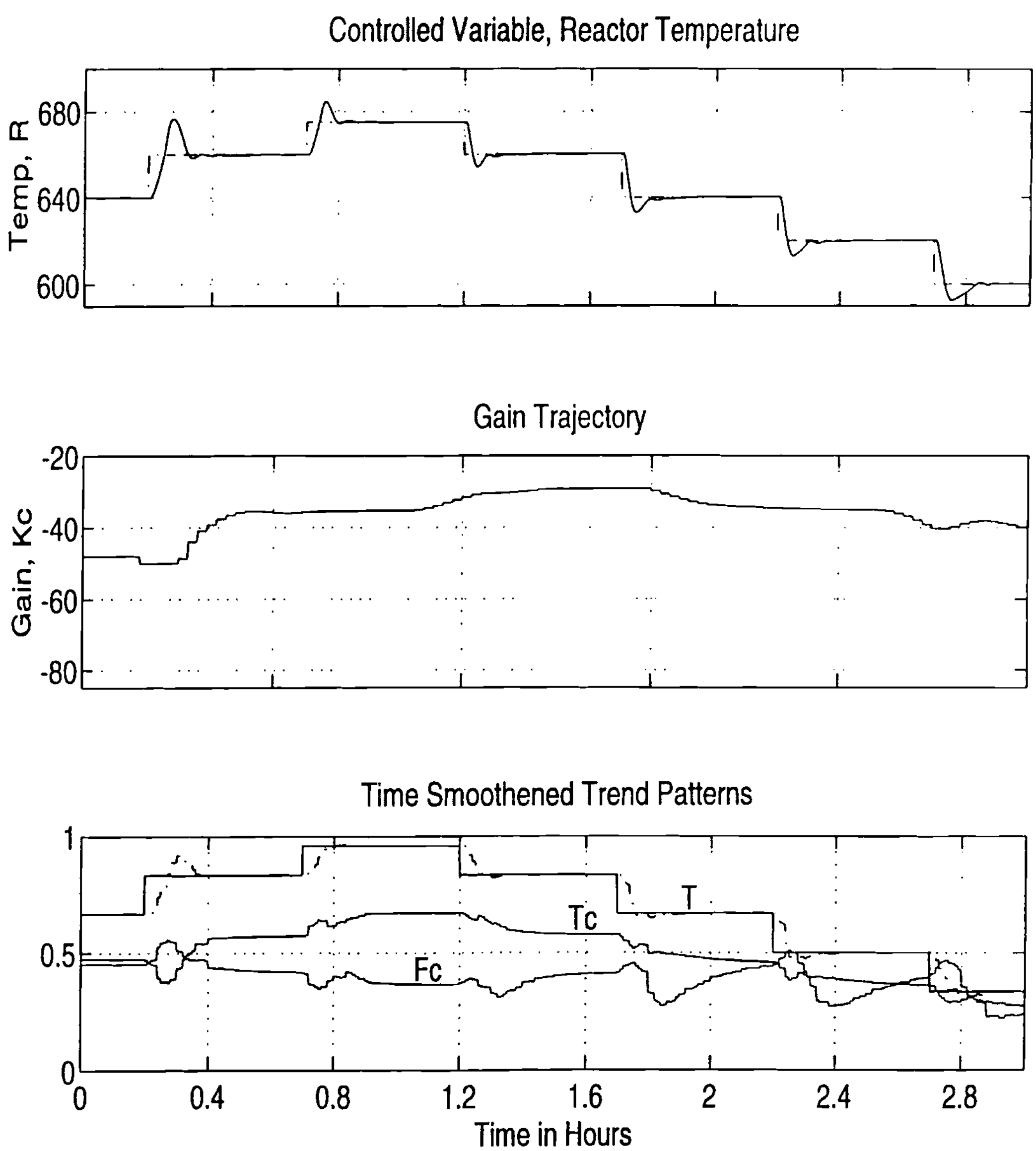


Figure 5.12: Overall servo performance using the ART-2 based similarity measure.

Moreover, the gain surface approximated using a quadratic interpolation technique can be inferred to be less discontinuous than the case using linear interpolation. This is based on an assumption that any discontinuity in gain trajectory is an indication of a discontinuity in the approximated gain surface.

Fuzzy interpolation is implemented using a triangular fuzzy set to represent controller settings at each nominal condition. The span of a triangular fuzzy set representing the controller gain is 20 ($ft^3hr^{-1}\ ^oR^{-1}$) while the one representing the reset time (τ_I) is 0.02 hr. A center of area defuzzification is used as described in Chapter IV. The gain trajectory is much more smoother than the previous two methods as shown in Figure 5.13. This indicates that fuzzy interpolation results in much smoother approximation of the gain surface.

The performance of fixed gain PI controllers is also shown for comparison purpose. Figure 5.14 shows controller performance with controller settings for operating condition number 3 (see Table 5.1). Fixed controllers having controller settings corresponding to operating conditions 4 and 7 are shown in Figures 5.15 and 5.16, respectively. It is clear from these simulations that fixed gain controllers perform effectively only very close to the nominal operating condition. On the other hand, a simple gain scheduling algorithm superimposed on the PI control structure greatly enhances the controller performance. The results of the overall servo performances for the different interpolation strategies are tabulated in Table 5.4.

CASE J FUZZY INTERPOLATION	
Sensor Pattern:	Time smoothened
Window:	6 minute sliding window
Interpolation:	Fuzzy using the 4 closest neighbors
Similarity Measure:	Euclidean the norm
IAE:	5.22

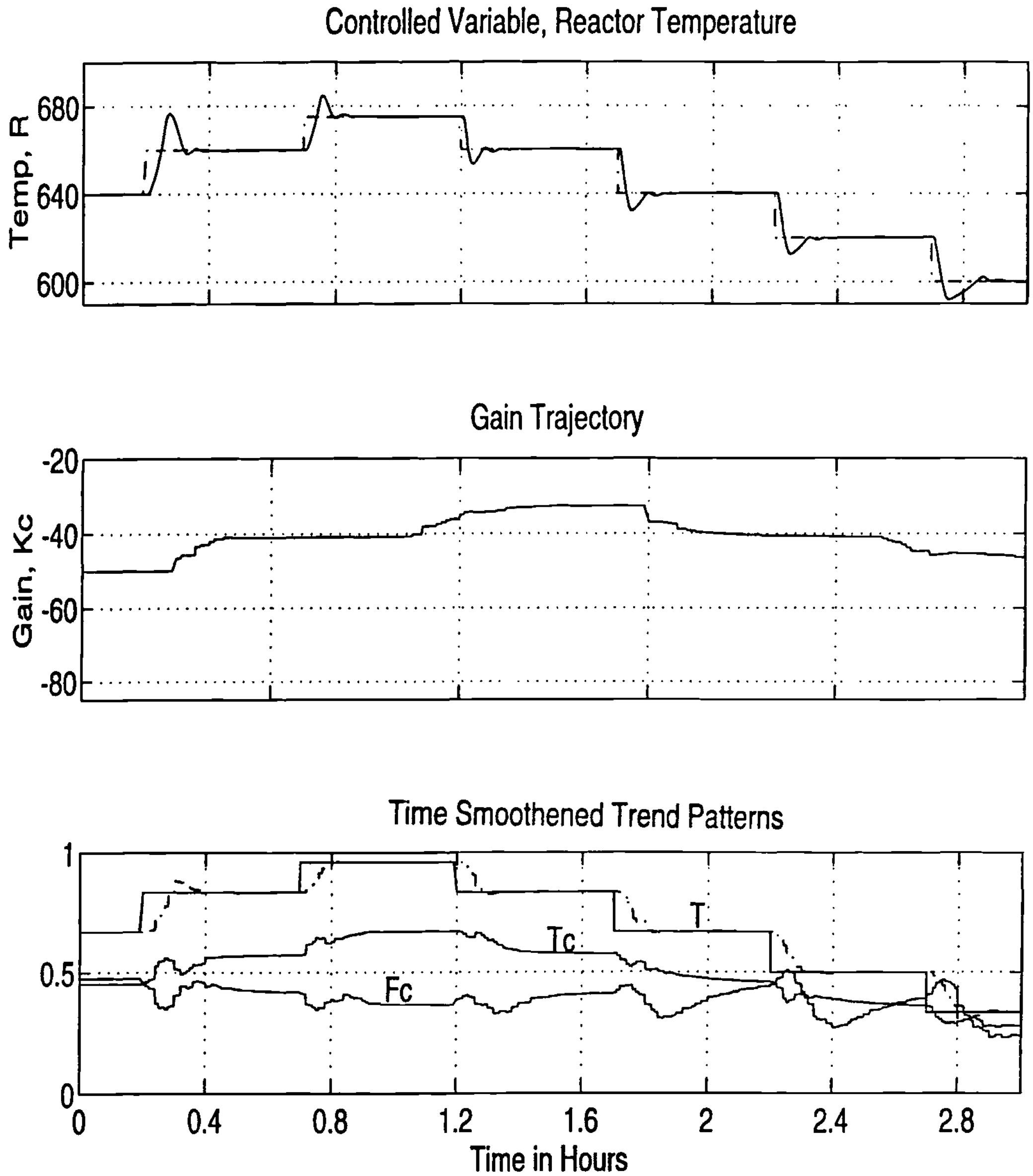


Figure 5.13: Overall servo performance using center of area (COA) fuzzy interpolation.

CASE K FIXED PI AT NOMINAL CONDITION 3	
Sensor Pattern:	Time smoothened
Fixed Parameters :	K_C= -35; τ_I = 0.08
IAE:	5.31
Comment:	Poor performance away from the design conditions

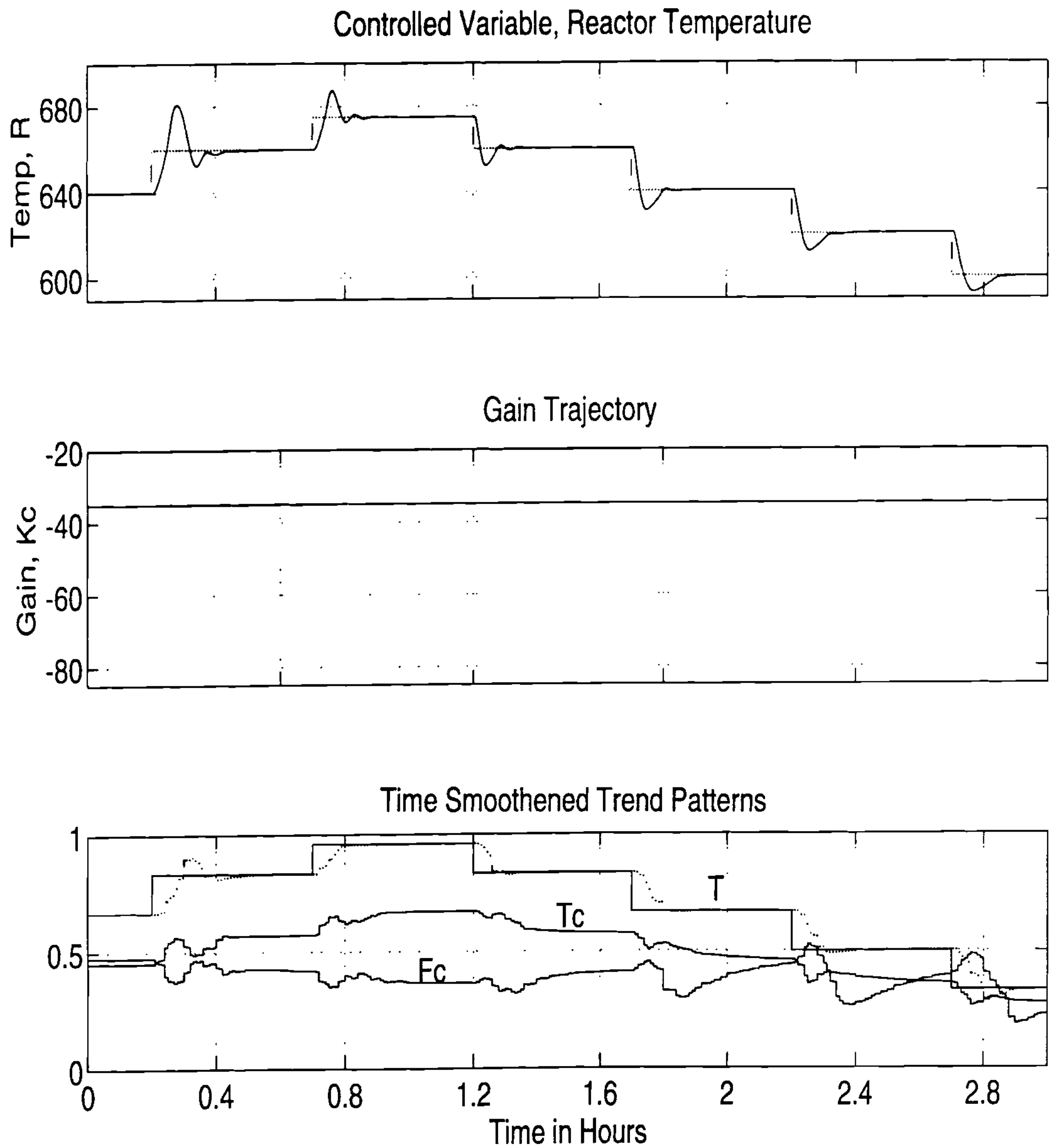


Figure 5.14: Overall servo performance of a fixed PI controller. Controller settings corresponding to nominal operating condition number 3.

CASE L FIXED PI AT NOMINAL CONDITION 4	
Sensor Pattern:	Time smoothened
Fixed Parameters:	K_C= -30; τ_I = 0.15
IAE:	10.55
Comment:	Limit cycles.

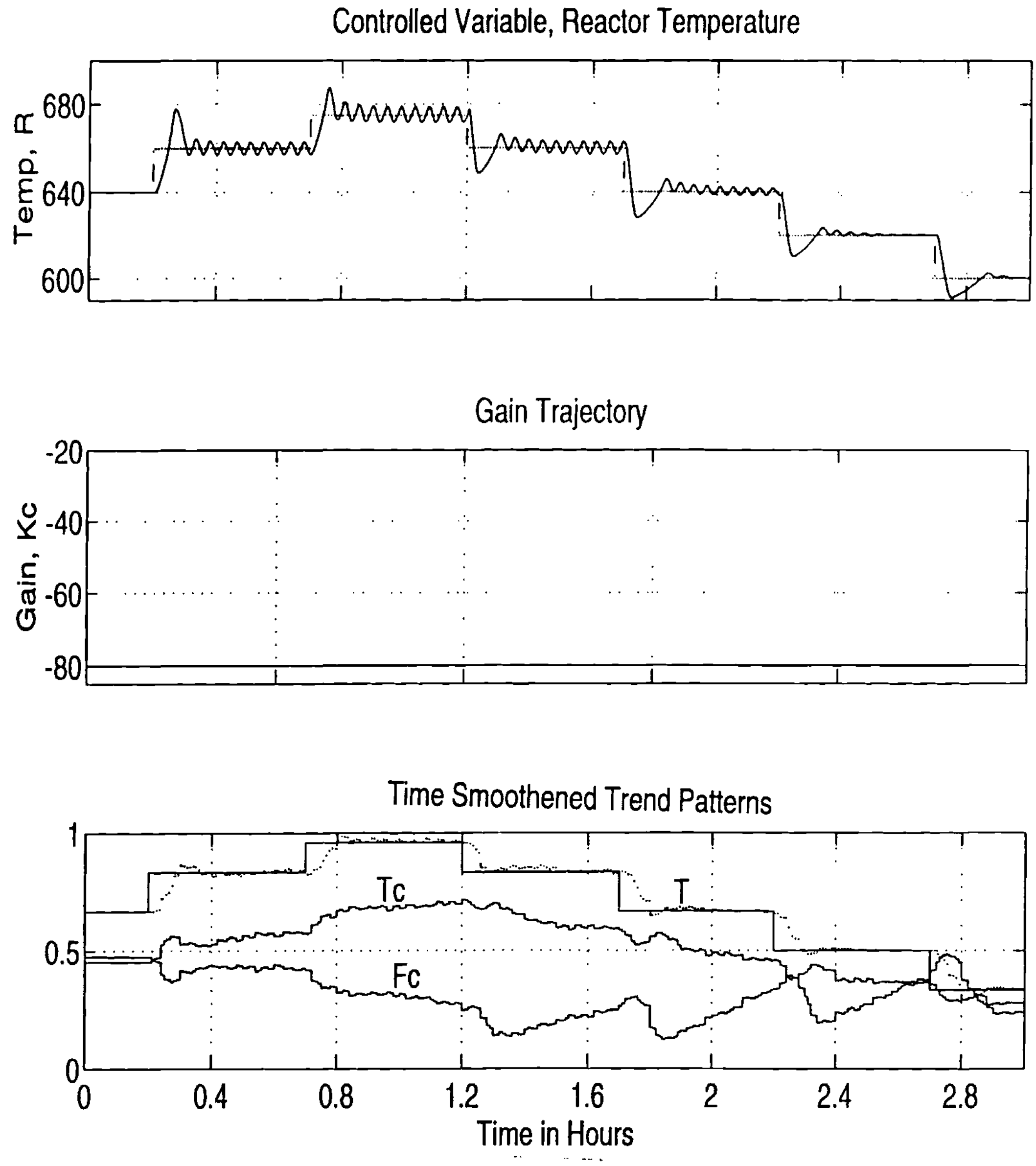


Figure 5.15: Overall servo performance of a fixed PI controller. Controller settings corresponding to the nominal operating condition 4.

CASE M FIXED PI AT NOMINAL CONDITION 7	
Sensor Pattern:	Time smoothened
Fixed Parameters:	K_C= -50; τ_I = 0.12
IAE:	5.26
Comment:	Large settling time away from the design operating condition.

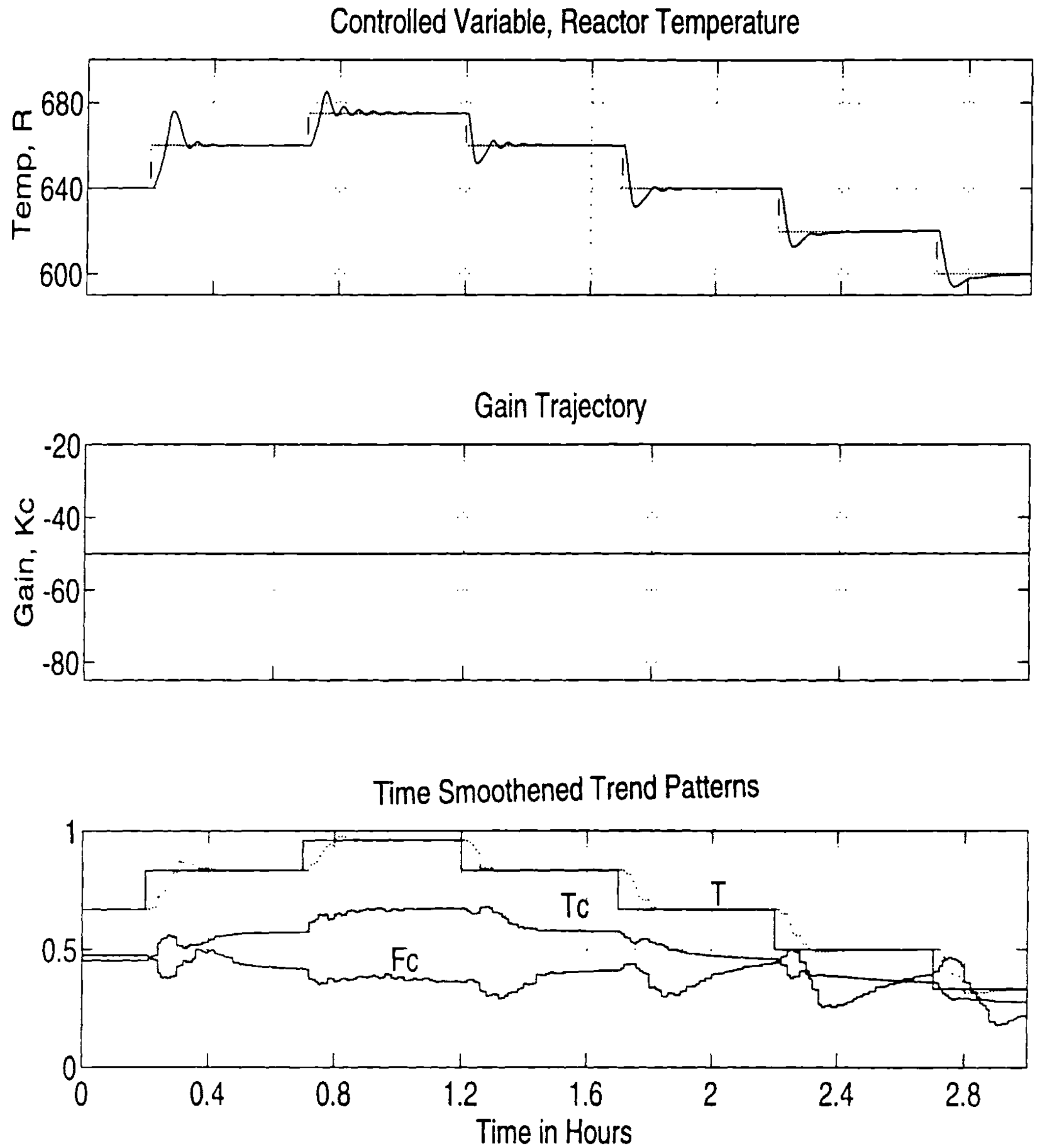


Figure 5.16: Overall servo performance of a fixed PI controller. Controller settings corresponding to the operating condition number 7.

Table 5.4

Overall Servo Performance of Fixed PI Controllers and Gain Scheduled Controllers Using Different Interpolation Strategies.

Case	Similarity Measure and Interpolation Strategy Used	IAE
H	Linear Interpolation using the 4 closest nominal conditions.	5.31
I	Using ART-2 based pattern similarity measure and quadratic interpolation using the highest 4 similarity values.	5.16
J	Using the Euclidean norm for pattern recognition and a center of area (COA) fuzzy interpolation.	5.22
K.	Fixed PI controller, parameters corresponding to those of nominal operating Condition 3.	5.31
L	Fixed PI controller, parameters corresponding to those of nominal operating condition 4	10.55 (Limit Cycles)
M	Fixed PI controller, parameters corresponding to those of nominal operating condition 7	5.26.

5.5 Concluding Remarks

From the results shown in this chapter it is clear that the successful implementation gain scheduling is dependent on accurate characterization of the process dynamics. Use of multi-sensor trend patterns is critical for accurate process state characterization. Use of time smoothening is also very important to monitor the change in process trend towards a different operating condition.

Another key issue is use of more than two nominal conditions to schedule the final controller parameters. The similarity measure itself does not play a very important role. Thus a simple Euclidean Norm based similarity measure can be thus for final design. Though the use of quadratic and fuzzy interpolation strategies does not result in a large improvement in IAE values they definitely result in a much smoother gain trajectory. This is an indication that the real gain surface can be better approximated by continuous smooth nonlinear surfaces as discussed in Chapter IV.

CHAPTER VI

CONCLUSIONS

The main contribution of this thesis is that it has led to an improved understanding that characterization of process dynamics is the key issue in implementation of a gain scheduled control system design for controlling chemical processes. How the controller parameters can be scheduled for any possible operating condition based on information regarding the controller settings at a few discrete operating points is also formalized. Such an analysis is necessary if gain scheduling is to evolve as a control technique to compensate for nonlinearites in chemical plants.

Gain scheduling is a standard way of designing flight control systems for aircraft which operate at wide ranges of altitude and speeds. It is gaining popularity for process control applications. The main bottleneck in its application in chemical process industries is that there is no technique to extract information regarding the state of the process from on-line sensor data. Thus, the emphasis of this thesis has been to develop a methodology to predict the state of the process from the on-line multi-sensor trend data.

The importance of pattern-based information is vital for accurate characterization of process dynamics. This conclusion can be made based on the results in Chapter V. It is shown that an instantaneous view of the process can lead to a wrong schedule. A pattern-based process characterization is proposed. This method of process characterization relies heavily on extracting the steady state trends from transient sensor data. The use of time smoothening as a way to compress pattern information is also demonstrated. It has been demonstrated that a simple pattern-based technique such as time smoothening can considerably improve the gain scheduling.

The next critical issue is to be able to use the results of this process characterization scheme to schedule controller settings in an on-line fashion. How the interpolation is performed is of vital importance in this adaptive control scheme. Interpolation plays a very vital role because a process rarely operates at any one nominal condition. A formal methodology to perform interpolation based on a few nominal operating conditions is developed in this study. One main result that emerges from simulation results shown in Chapter V is that for the given system, at least four nominal conditions are needed to perform interpolation. The number of nominal conditions required to be used for interpolation depends on the distribution of the nominal operating conditions in the pattern-based gain map.

The analysis regarding the interpolation strategy in Chapter IV has revealed that interpolation is a way to approximate the gain surface. The characteristic of the gain trajectory can indicate how well the gain surface is approximated. Our investigations show that fuzzy interpolation results in a much smoother gain trajectory. This is desirable since rapid parameter variations can often lead to instability. Similarity measure based on a simple Euclidean norm between pattern vectors is adequate for gain scheduling purposes.

The main conclusions are summarized in the list below.

1. Characterization of the process dynamics is the key issue in successful implementation of a gain scheduled controller.
2. Patterns of multi-sensor data should be used to accurately characterize the process dynamics.
3. Time smoothening as a way to compress pattern-based information results in considerable improvement in the performance of gain scheduled systems.
4. More than two nominal conditions should be used for interpolation purposes. The exact number of nominal conditions to be used for gain scheduling depends on the distribution of nominal conditions in the pattern representation space.
5. Euclidean Norm as a pattern similarity measure is adequate for gain scheduling purposes.
6. Fuzzy interpolation results in a much smoother gain trajectory.

6.1 *Future Work*

The attention is now directed towards future research. The entire thesis is built on the fact that the relation between process dynamics and controller parameters are known for a few operating conditions. In this work, the controller parameters for the demonstration system were determined by use of process reaction curve and continuous cycling methods. All these methods are based on empirical rules relating the controller parameters to the sensitivity of the manipulated variable to the controlled variable. The main advantage of an adaptive control strategy is that it can accommodate dynamics of process variables other than the controlled variable by adjusting the controller parameters. Hence it is important to investigate the effect of other process variables on the controller parameters. Such a study can give invaluable insight on the characteristic of the gain surface relating controller parameters to process dynamics. Such a analysis can also throw light on the interpolation technique that shows the same characteristic as the real gain surface.

Another implementation issue that was addressed in this thesis but needs further investigation is the way to predict the process state that the process is evolving to, during periods of transition. In this expectation-based gain scheduling, more and more weight would be given to the controller parameters corresponding to the nominal condition at which the process would finally settle. We have addressed this by incorporating the desired set point as a scheduling variable. It is possible to extract information regarding the trend change in the process by looking at smoothened transient data. With advances in pattern recognition it is possible to classify different transient trajectories and develop a method to predict the state towards which the process is headed. The main bottleneck for such an investigation is the lack of good transient data in process industries.

Analysis of the stability of such control methodology is needed if this method is to ever evolve as a commercial technique for process control application. This is a formidable task because there is no function to map the controller parameters to the process state variables. Absolute stability theory using Popov criteria (Popov, 1962) can be used to establish stability margin and establish boundaries on parameters that will yield stable controllers for a nonlinear system. Investigations performed by Shamma (1988) show that main source of instability in such a design scheme is due to rapid variations in the system parameters. Though this work has developed a method to minimize rapid variations by use of time smoothened pattern information and a better interpolation strategy, a theoretical analysis of stability is missing. Such a study would be based on the linear parameter variation (LPV) analysis (Desoer and Vidyasagar, 1975, Shamma, 1993) of control systems.

From a design perspective, this work can be used as a starting point in development of a gain scheduled controller for chemical processes. This work presents a scheme to leverage process information from on-line sensor readings for use in designing an adaptive controller. Unlike other existing methods proposed to perform gain scheduling this method does not disrupt the smooth plant operation in order to characterize its state. One drawback is that the design of a such an adaptive controller is time consuming. The tuning parameters need to be calculated at the nominal operating conditions and the scheduling variables need to be identified before the gain scheduled controller can be implemented. The calculation of controller settings based on the auxiliary and process variables is often computationally intensive. This is not a major problem with the advancement of micro-processor technology. It is not difficult to implement such a methodology on the platform offered by the modern distributed control systems.

REFERENCES

Anderson, J. J., "Pattern Based Gain Scheduling," *MS Thesis, Oklahoma State University* (1993).

Anderson, J. J. and J. R. Whiteley, "Application of ART-2 Neural Network for Nonlinear Process Control," *Proceedings of the 7th Oklahoma Symposium on Artificial Intelligence.* 18-27 (1993).

Andreiv, N., "Prccess Controller that Adapts to Signal and Process Conditions," *Control Engineering* **24,** 38-40 (1977).

Åström, K. J., "Theory and Applications of Adaptive Control -a Survey," *Automatica* 471-486 (1985).

Bristol, E. H., "The design of Industrially Useful Adaptive Controllers," *ISA Transactions* **22,** 17-25 (1983).

Cao, R. and T. McAvoy, "Evaluation of a Pattern Recognition Adaptive PID Controller," *1990 International Federation of Automatic Control* 797-801 (1990).

Cardello, R. J. and K. San, "The Design of Controllers for Batch Bioreactors," *Biotechnology and Bioengineering* **32,** 519-526 (1988).

Carpenter, G. A. and S. Grossberg, "ART-2: Self Organization of Stable Category Recognition Codes for Analog Input Patterns," *Appl. Opt.* 4919-4930 (1987).

Cooper, D. J. ;. A. M. Lalonde and R. Pae, "Automated Gain Schedule Generation for Nonlinear Adaptive Control," *Chem. Eng. Comm.* **81,** 109-121 (1989).

Desoer, C. A. and M. Vidyasagar, "Feedback Systems: Input Output Properties," *Academic Press* (1975).

Driankov, D. , H. Hellendroom, M Reinfrank, "An Introduction to Fuzzy Control," *Springer- Verlag* (1990)

Hinde, R. F. and D. J. Cooper, "Adaptive Process Control," *J. Proc. Cont.* **1,** (1991).

Jury, F. D., "Fundamentals of Three-Mode Controllers," *Fisher Controls Company Technical Monograph* (1973).

Kallstrom, C. G. and K. J. Åström, "Adaptive Autopilots for Tankers," *Automatica* **12,** 241-254 (1979).

Kraus, D. J. and T. J. Myron, "Self tuning PID Controllers uses Pattern Recognition Approach," *Control Engineering* 106-111 (1984).

Leuba, M., C.M. Abdalla, C.E. Ford and L. Guimareas, "A Hybrid Fuzzy-PI Adaptive Control for U-Tube Steam Generators," *Control Theory and Advanced Technology* **8,** 567-575 (1992).

Luyben, W. L., "Process Modeling, Simulation and Control for Chemical Engineers," *McGraw Hill, New York* (1973).

Magen, L. and D. J. Cooper, "Neural Network Based Adaptive Control via Temporal Pattern Recognition," *Canadian Journal of Chemical Engineering* **70,** 1208-121 (1992).

Mellichamp, D. A., D. R. Coughanowr and L. B. Koppel, "Characterization and Gain Identification of Time Varying Flow Processes," *A.I.Ch.E. Journal* **12,** 75-82 (1966a).

Mellichamp, D. A., D. R. Coughanowr and L. B. Koppel, "Identification and Adaptation in Control Loops with Time Varying Gain," *A.I.Ch.E. Journal* **12,** 83-89 (1966d).

Popov, V. M., "Absolute Stability of Nonlinear Control Systems of Automatic Control," *Automation and Remote Control* **22,** (1962).

Rugh, W. J., "Analytical Framework for Gain Scheduling," *IEEE Control Systems* **11,** 79-84 (1991).

Seborg, D. E., T. F. Edgar and D. A. Mellichamp, "Process Dynamics and Control," *Wiley, New York* (1989).

Seborg, D. E., T.F. Edgar and S. L. Shah, "Adaptive Control Strategies for Process Control: A Survey," *AIChE Journal* **32,** 881-913 (1986).

Shamma, J. S., "Design and Analysis of Gain Scheduled Control Systems," *Ph.D. Thesis, Massachusetts Institute of Technology* (1988).

Shamma, J. S., "The Necessity of the Small Gain Theorem for Time-Varying and Nonlinear Systems," *IEEE Trans. Automat. Contr.* **36,** 1138-1147 (1991).

Shamma, J. S. and M. A. Athans, "Analysis of Gain Scheduled Control for Nonlinear Plants," *IEEE Trans. Automat. Contr.* **35,** 898-907 (1990).

Shamma, J. S. and D. Xiong, "Stabilization of Linear Parameter Varying Systems," *Proceedings of 32nd Conference on Decision and Control* (1993).

Smith, C. A. and A. B. Carripio, "Principles and Practice of Automatic Process Control," *Wiley,New York* (1985).

Soule, L. M., "Tuning a Three Mode Controller," *Chemical Engineering* 100-104 (1969).

Stephanopulous, G., "Chemical Process Control: An Introduction to Theory and Practice," *Prentice-Hall, Englewood Cliffs, New Jersey* (1984).

Stien, G., "Adaptive Flight Control - A Pragmatic View," *Applications of Adaptive Control (K.S. Nagendra, Ed),Academic, New York* (1980).

Whatley, M. J. and D. C. Pott, "Adaptive Gain Improves Reactor Control," *Hydrocarbon Processing* **63,** 75-78 (1984).

Whiteley, J. R. and J. F. Davis, "Qualitative Interpretation of Sensor Patterns," *IEEE Expert* **8,** 54-63 (1993).

Wong, S. K. and D. E. Seborg, "Control Strategies for Single Input Single Output For Nonlinear Systems with Time Delay," *International Journal of Controls* **48,** 2303-2307 (1988).

Zeigler, J. H. and N. B. Nichols, "Optimum Settings for Automatic Controllers," *Trans. ASME* **64,** 759 (1942).

Zhao, Z.-Y., M. Tomizuka and S. Isaka, "Fuzzy Gain Scheduling of PID Controllers," *IEEE Transactions on Systems, Man, and Cybernetics* **23,** 1392-1398 (1993).

Zimmerman, H. J., "Fuzzy Set Theory," *Kluwer Academic Publishers* (1991).

APPENDIX

Model Development and Simulation

A1.1 Introduction

The appendix can be divided in two parts. In the first section the model development of the CSTR is presented followed by the discussion of the dynamic characteristic of the CSTR. The second section documents the tuning techniques used to establish the controller parameters. Finally, the simulation of the demonstration system under a gain scheduled controller using a dynamic simulator, Simulink, is presented.

A1.2 Nonisothermal CSTR

Nonisothermal reactors are often the most difficult units to control in a chemical plant, particularly if the reaction is exothermic. Small deviations in temperature can significantly change the reaction rate and, consequently, the yield. Furthermore, the increase in reaction rate with increasing temperature tends to make the reactor unstable. The system discussed here has been adapted from Luyben (1990).

An irreversible exothermic reaction (chemical "A" reacting to form another chemical "B") is carried out in a perfectly mixed CSTR. The reaction is first order and exothermic in nature. Negligible heat losses and constant density are assumed. The reactor has a coolant jacket and the flowrate of coolant is used to control the temperature of the reactants.

Simple first order reaction is expressed as:

$$A \longrightarrow B \tag{A1.1}$$

where A is the reactant and B is the product.

Arrhenius function governing the reaction rate is expressed as:

$$k(T) = k_O \exp(-E/RT) \tag{A1.2}$$

where k_O, E and R are constants and T is reactor temperature in ^oR.

The ODEs describing the system are:

Total continuity:

$$\frac{dV}{dt} = F_o - F \tag{A1.3}$$

where F_o and F are inlet and outlet flowrates in ft^3/hr

Reactor Component A continuity :

$$\frac{d(VC_A)}{dt} = F_o C_{AO} - FC_A - VkC_A \qquad (A1.4)$$

where C_{AO} and C_A are the concentration of reactant A in feed and reactor.

Reactor Energy Equation:

$$\rho \frac{d(VT)}{dt} = \rho(F_o - F) C_P - \Delta H_{rxn} VkC_A - UA_{HX}(T-T_J) \qquad (A1.5)$$

where C_P is the heat capacity of the reacting mixture and ΔH_{rxn} is the exothermic heat of the reaction. U is the heat transfer coefficient and A_{HX} is the heat transfer surface area.

Jacket energy equation:

$$\rho_J V_J C_{PJ} (\frac{dT}{dt}) = \rho_J F_J C_{PJ} (T_{JO} - T_J) + UA_{HX} (T-T_J) \qquad (A1.6)$$

C_{PJ} is the heat capacity of the coolant.

The CSTR model parameters are shown in Table A1.1. These are the CSTR parameters at a steady condition operating at 600^0R.

Table A1.1
Nonisothermal Reactor Parameters

Volume $V_J = 3.85$ ft^3	Rate Constant : $k_O = 7.08 \times 10^{10}$ hr^{-1}
Activation Energy : E = 30,000Btu/lb.mol	Gas constant R = 1.99 Btu/ lb.mol ^oR
Heat Transfer Coeff :U= 150 BTU/hr ft^2 ^oR	Heat transfer Area : $A_{HX} = 250$ ft^2
Coolant Inlet Temp. $T_{JO} = 530$ ^oR	Heat of Rxn:ΔH_{RXN}=-30,000 Btu/ lb.mol^oR
React. Specific Heat: $C_p = 0.75$ BTU/lb$_m$ ^oR	Coolant Specific Heat:C_{PC}=1.0 BTU/lb$_m$$^oR$
Reactant Density : $\rho = 50$ lb$_m$/ft^3	Coolant Density $\rho_J = 62.3$ lb$_m$/ ft^3
Inlet Concentration : 1.0 lb.mol A/ ft^3	Set Point Temperature : $T^{SET} = 600$ ^oR

A1.2.1 Open Loop Dynamic Characteristics

The main source of nonlinearity is the Arrhenius dependence of the reaction rate. For this example, even a 1^{o}R rise in temperature increases the reaction rate by more than 10 percent, enough to cause significant change in conversion. Furthermore, the increase in rate with increasing temperature tends to make the reactor unstable. The effect of this reactor going unstable will be further analyzed based on a steady state analysis later in this subsection.

Figure A1.1 shows that the reactor in absence of a controller runs away at certain temperature ranges. At 680^{o}R the process is stable in open loop but does not show a response which can be modeled as a first order plus dead time (FOPDT). The process runs away at 600^{o}R and settles down to a new steady state. Such a response cannot be

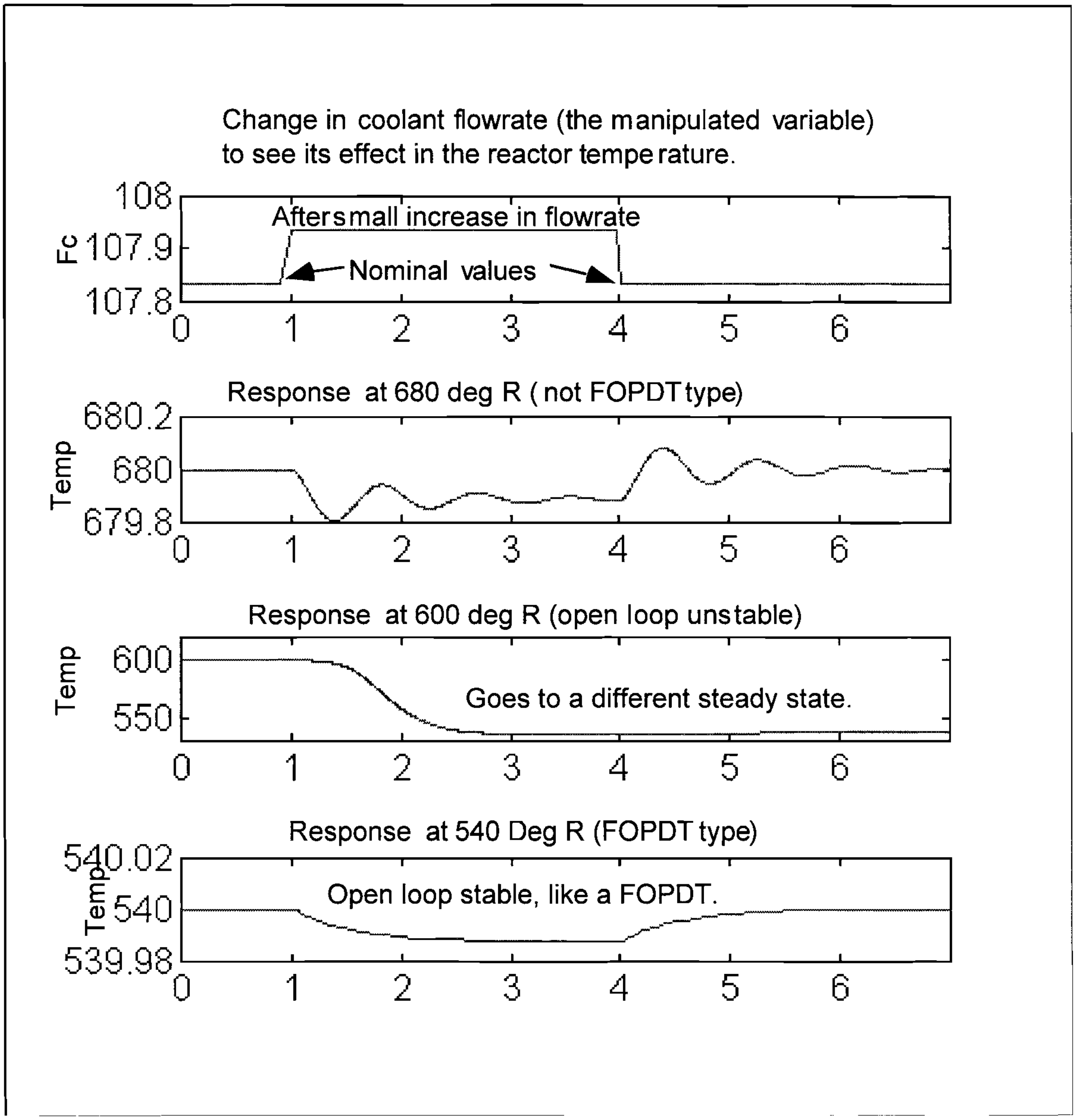


Figure A1.1: Open loop behavior of the demonstration system.

modeled by a FOPDT since the process is actually operating between two steady state points. The reactor shows a FOPDT type response at lower temperatures (540^{o}R).

What makes the dynamics of CSTR especially interesting is that for any value of a manipulated variable there can exist more than one steady state. Whether this CSTR shows multiplicity will be explored next. There is one source of heat generation , the heat of reaction, and two ways in which this heat is removed. Some of the heat is removed as sensible heat in the product stream and the rest is transferred to the coolant in the cooling jacket. The heat generated by the chemical reaction and the heat removed by the jacket and product stream are plotted together against the reactor temperature,as shown in Figure A1.1. This plot is shown for a fixed coolant jacket flowrate.

The heat generated in the CSTR is the heat of reaction at the given steady state conversion corresponding to each reactor temperature. The heat generated is calculated as follows:

$$Q_{GEN} = \frac{-\Delta H_{RXN}\tau C_{AO}}{\rho C_P}\left(\frac{k(T)}{\tau.+k(T)}\right) \qquad (A1.7)$$

where $\tau = V/F$

The generation rate is high at first because of the exponential effect of temperature but the slope eventually decreases because of the decrease in reactant concentration. Thus the graph becomes asymptotic to the heat released at complete conversion. The heat removed graphs are linear as shown in the expression below.

$$Q_{REM} = UA(T-T_J) + F_J \rho C_P (T - T_O) \tag{A1.8}$$

The heat generated and the heat removed are plotted as a function of reactor temperature. The slope of the plot of the heat removed depends on the flowrate. There can be three cases based on the coolant flowrate. For Cases 1 and 3 (see Fig. A1.2) there can be only one equilibrium point for which the rate of heat removal is same as the rate of heat generated. For case 2 there can be three steady states. The middle steady state ("B") is clearly a unstable steady state even though heat balance equation is satisfied. A slight increase in temperature makes the rate of generation of heat more than the rate of heat removal which results in the reactor "running away" to the upper steady state ("C"). Similarly a decrease in temperature results in the reactor "running away" to steady state ("A").

Thus controlling this reactor at unstable steady states becomes a challenging control problem. The amount of self regulation provided by the reactor dynamics depends entirely on the condition of the steady state operation. For steady states which are "stable" even a no control action can ensure that the reactor will be stable. But for conditions when it is "unstable" as in condition "B" (shown in the Figure A1.2) tight control is necessary. Often commercial reactors are operated at nearly unstable conditions for economic reasons. There have been many cases of runaway reactions leading to plugged kettles, melted reaction tubes and explosions.

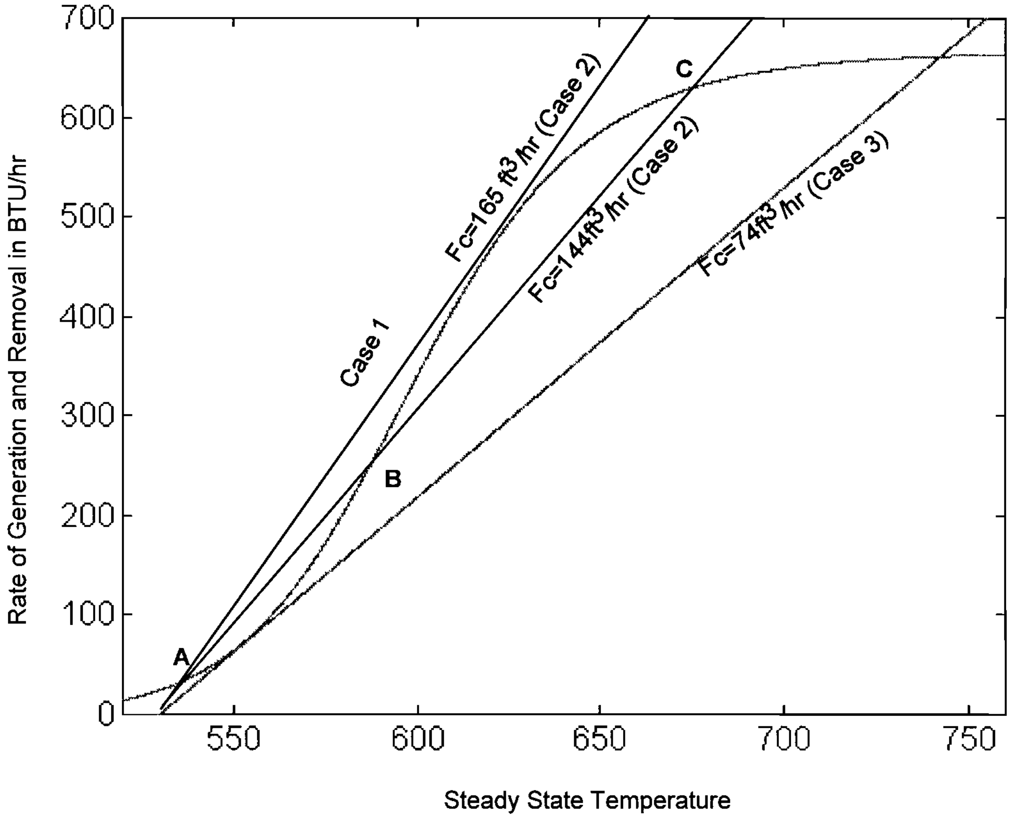


Figure A1.2 : Existence of multiple steady states

A1.3 Estimating Controller Settings at Nominal Operating Points.

The gain scheduling controller can be visualized as made up of many different linear controllers. The performance of this global compensator depends on the quality of each local controller tuning. A good set of controller settings should be able to perform satisfactorily for a small change in a setpoints disturbance and be stable for even a reasonably large perturbation in process conditions.

It is always a subject of controversy as what the performance criteria should be. A simple performance criteria is based on some characteristic feature of the closed loop response of the system (Stephanopoulos 1990). The most often used criteria are

- overshoot.
- rise time (time needed for the response to reach the desired value).
- settling time (the time needed for the response to settle within 5% of the desired value).
- error integrated over time(IAE or ITAE).

Selection of different performance criteria result in different controller settings. The process dynamics are often inferred from the open-loop response of the process to a small change in a manipulated variable. The open loop response is then approximated as a first order plus time delay (FOPTD) model. The resulting model parameters, along with empirical tuning correlations are used to determine the initial controller settings.

One challenge in designing a controller for this nonisothermal CSTR is that most of the conventional tuning methods fail. This is because the process response cannot be approximated by a FOPDT model.

For open loop unstable cases, tuning is reduced to a trial and error technique. The initial guess can be obtained by forcing the process to operate at conditions where the process dynamics can be modeled as a FOPDT. Using the empirically correlated controller parameters as the initial guess, the controller parameters can be established. For process operating conditions when the response is open loop stable but cannot be approximated by a lower order dynamic model, continuous cycling method (Zigler and Nichols, 1942) can be used. Continuous cycling method involves observing the system dynamics under closed loop conditions. The controller is put in a P (proportional) only mode and the controller gain (K_C) is increased untill the process shows continuous oscillations of constant amplitude. The controller gain resulting in these continuous oscillations is called the ultimate gain (K_U). K_U can be used to empirically establish the controller parameters.

The final controller parameters were established after trial and error tuning. The transient closed loop response after trial and error tuning resembled the one shown in Figure A2.3 for all nominal conditions. An effort was also made to arrive at controller settings which gave minimum IAE (integral of absolute error) for both set point and load changes at any given operating condition. Furthermore these controller settings were tested with reasonably large load and set point changes to check if they yielded a stable response.

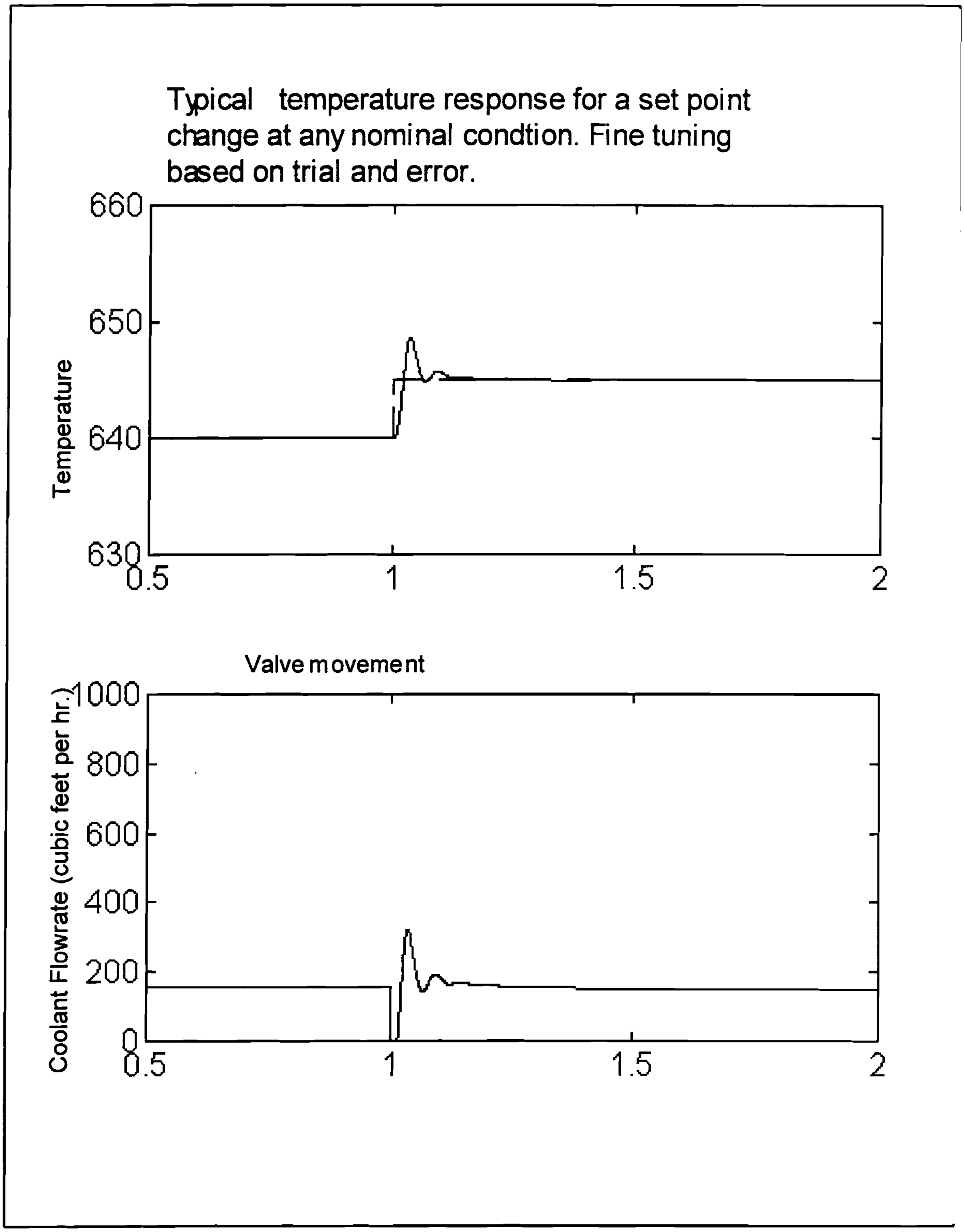


Figure A2.3: Typical controller performance at any nominal operating point.

The CSTR dynamics were simulated using a package called "Simulink" by MathWorks Inc. This package is an extension of MATLAB and is excellent for studying the dynamics of a system and the effect of different control strategies. Simulink has built-in integration subroutines which makes simulation a relatively easy task for a user.

VITA

Manish Sinha

Candidate for the Degree of

Master of Science

Thesis: PATTERN-BASED PROCESS CHARACTERIZATION AND GAIN SCHEDULING FOR NONLINEAR CHEMICAL PROCESSES.

Major Field: Chemical Engineering

Biographical:

Personal Data: Born in Gaya, Bihar, India, September 20, 1970, son of Madhuri and Anjani Sinha.

Education: Graduated from St. Xaviers School, Bokaro Steel City, June 1989; received Bachelors of Technology (B-Tech) in Chemical Engineering from Indian Institute of Technology (IIT) at Bombay, India, June 1993; completed the requirement for Master of Science degree at Oklahoma State University in July 1995.

Professional Experience: Summer Intern at Adarsh Chemicals,India, from May 1992 to July 1992; R&D Engineer at Adarsh Chemicals, India from May 1993 to July 1993; Graduate Research Assistant , Department of Chemical Engineering, Oklahoma State University, August 1993 to December 1993; Teaching Assistant, Department of Chemical Engineering, Oklahoma State University, January 1994 to December 1994; Computer Lab Consultant, CEAT Design and Development, Oklahoma State University, January 1995 to May 1995.