THE UNIVERSITY OF ALBERTA

A COMPARISON OF ADAPTIVE CONTROLLERS: ACADEMIC VERSUS INDUSTRIAL

by

BRENDAN J. P. MINTER

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF MASTER OF SCIENCE

IN

PROCESS CONTROL

DEPARTMENT OF CHEMICAL ENGINEERING

EDMONTON, ALBERTA

FALL, 1987

ISBN 0-315-40981-9

THE UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: BRENDAN J. P. MINTER

TITLE OF THESIS: A COMPARISON OF ADAPTIVE CONTROLLERS: ACADEMIC VERSUS INDUSTRIAL

DEGREE FOR WHICH THESIS WAS PRESENTED: MASTER OF SCIENCE

YEAR THIS DEGREE GRANTED: FALL, 1987

(SIGNED) ....

PERMANENT ADDRESS:

#4, 10133 - 123st

Edmonton Alberta

T5N [illegible]

DATED October 2, 1987

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the FACULTY OF GRADUATE STUDIES AND RESEARCH, for acceptance, a thesis entitled A COMPARISON OF ADAPTIVE CONTROLLERS: ACADEMIC VERSUS INDUSTRIAL submitted by BRENDAN J. P. MINTER in partial fulfilment of the requirements for the degree of MASTER OF SCIENCE in PROCESS CONTROL.

....

Supervisor

....

....

....

Date....October 2, 1987.......

To Mum and Dad

Abstract

This thesis evaluates and compares the design, implementation and operation of Clarke and Gawthrop's generalized minimum variance (GMV) adaptive predictive controller with two industrial self-tuning PID controllers: The Foxboro Company's "Exact" and Turnbull Control System's "Auto-Tuning" Controller.

The TCS controller uses a statistically determined predictive model as a basis for continually providing the operator with recommended PID controller constants. Performance on the nonlinear plant was generally unsatisfactory because there was no reliable indication of when to update the PID constants and the controller had fewer practical design features than the other controllers.

The Exact controller's self-tuning algorithm automatically adjusts the PID constants once per transient until user specified values of overshoot and/or damping are met. This generally required 5-10 transients (eg. step changes in set point) for major adjustments. The controller is easy to implement, requires minimal expertise for operation and gave good closed loop performance on a very nonlinear process.

The GMV can be interpreted as a *fixed-parameter* 1/Q control law acting on an adaptive predicted control error. This suggests a PI form for 1/Q to facilitate comparison with the other controllers. Predictive control improved closed loop performance but experimental runs on the

nonlinear temperature process showed that a fixed gain controller was unsuitable. Therefore, a second self-tuning loop was added to the GMV controller which adjusted the controller gain in 1/Q such that a user specified overshoot was maintained for step changes in set point. Note that the Exact controller can be used to tune the PI controller constants inherent in the 1/Q contro[illegible]f the GMV controller.

The experimental studies, plus the literature, show that self-tuning and/or adaptive predictive controllers offer definite advantages in selected industrial applications. However, the user must carefully select the desired design features such as continuous self-tuning, predictive control action, the performance criteria, etc., plus practical features such as nonlinear gain compensation, effective filtering options, pretune capabilities to facilitate startup and default options for safe operation during unusual process conditions. No controller offers all the features in one package.

Acknowledgement

I express my thanks to Dr. D.G. Fisher for both the technical and personal guidance he has offered during the course of this work. The many hours spent in discussion are greatly appreciated. I would also like to thank Robert Barton, of the DACS centre, for his technical help with the computer systems-related problems encountered in this work.

The financial support from both Dr. D.G. Fisher and the Department of Chemical Engineering, University of Alberta is gratefully acknowledged.

To my sister, Linda, and brother, Kerry, my gratitude for their patience, support, and strength during these past years of our lives.

Table of Contents

List of Figures

Figure | Page

List of Tables

Nomenclature

Alphabetic

A	cross-sectional area
$A(z^{-1})$	polynomial corresponding to process output
a_i	coefficient of $A(z^{-1})$, $a_0=1$
$B(z^{-1})$	polynomial corresponding to process input
b_i	coefficient of $B(z^{-1})$
$C(z^{-1})$	polynomial corresponding to measurment noise
c_i	coefficient of $C(z^{-1})$, $c_0=1$
d	system time delay
$\bar{d}$	d.c. bias term
$E[\cdot]$	statistical operator
$E(z^{-1})$	$G(z^{-1})B(z^{-1})$
e_0	first coefficient of $E(z^{-1})$
$e(k+d)$	predicted control error
$\hat{e}(k)$	$y(k)-\hat{y}(k)$
$\bar{e}(k)$	average value of $\hat{e}(k)$
$e'(k)$	$\hat{e}(k)-\bar{e}(k)$
$F(z^{-1})$	general polynomial
$F'(z^{-1})$	$F(z^{-1})/P_d(z^{-1})$
$G(z^{-1})$	general polynomial
$H(z^{-1})$	$C(z^{-1})-1$
$h(s)$	polynomial in Laplace domain
J	costing function

K_c	controller gain
K_i	controller reset time
K_p	process gain
k	time in sample instants
$L(z^{-1})$	general polynomial
l_i	coefficient of $L(z^{-1})$
$M(z^{-1})$	transfer function for closed loop reference model
n	dimension of $\theta(k)$
na	order of $A(z^{-1})$
nb	order of $B(z^{-1})$
nc	order of $C(z^{-1})$
nl	order of $L(z^{-1})$
npd	order of $P_d(z^{-1})$
npn	order of $P_n(z^{-1})$
nqd	order of $Q_d(z^{-1})$
nqn	order of $Q_n(z^{-1})$
nrd	order of $R_d(z^{-1})$
nrn	order of $R_n(z^{-1})$
P(k)	covariance matrix
$P(z^{-1})$	weighting transfer function acting on process output
$P_d(z^{-1})$	denominator polynomial of $P(z^{-1})$
$P_n(z^{-1})$	numerator polynomial of $P(z^{-1})$
q	time delay for measured disturbance
$Q(z^{-1})$	weighting transfer function acting on process input
$Q_d(z^{-1})$	denominator polynomial of $Q(z^{-1})$
$Q_n(z^{-1})$	numerator polynomial of $Q(z^{-1})$
R	valve resistence coefficient

$R(z^{-1})$	weighting transfer function acting on process set point
$R_d(z^{-1})$	denominator polynomial of $R(z^{-1})$
$R_n(z^{-1})$	numerator polynomial of $R(z^{-1})$
s	Laplace operator
T_r	trace of matrix
T_s	sampling interval
T_{95}	95% settling time of closed loop system
U_{ss}	steady state process input
u(k)	process input
u(s)	polynomial in Laplace domain
v(k)	measured disturbance
w(k)	process set point
x(k)	general disturbance term
x^T	I/O regressor vector
Y_{ss}	steady state process output
y(k)	process output
z	forward shift operator

Greek

β	forgetting factor for estimation of $\epsilon(k+d)$
Δ	delta operator, $(1-z^{-1})$
δ	$G(z^{-1})\bar{d}$
$\epsilon(k+d)$	$G(z^{-1})\xi(k+d)$

$\psi(k+d)$	$P(z^{-1})y(k+d)$
$\psi^*(k+d)$	$\psi(k+d)-G(z^{-1})\xi(k+d)$
$\xi(k)$	member of an uncorrelated zero mean random sequence (white noise)
γ	time constant (Laplace domain) for second order Butterworth filter
$\theta(k)$	parameter vector used in RLS/ILS
λ	scalar for control weighting
$\lambda(k)$	forgetting factor for RLS/ILS
$\lambda_u(k)$	forgetting factor for estimation of $\bar{u}(k)$
$\lambda_y(k)$	forgetting factor for estimation of $\bar{y}(k)$
σ^2	variance of $\xi(k)$
τ_d	process delay time
τ_i	controller reset time
τ_p	process time constant

Superscripts

$\bar{x}$	average value
$\hat{x}$	estimated value
$\tilde{x}$	deviation from average value
T	transpose

Subscripts

SS steady state

<u>Abbreviations</u>

A^3	academic, adaptive, algorithm
DOS	disk operating system
ELS	extended least squares
GLS	generalized least squares
ILS	improved least squares
RLS	recursive least squares

<u>Chapter 6 Abbreviations</u>

CLM	change limit for clamping PID parameters
D	self-tuned derivative time
DF	fixed derivative time
DER	switch for directionally varying derivative action
DFCT	derivative factor
DMP	user specified damping
I	self-tuned reset time
IF	fixed reset time
LAG	discrete time constant for Butterworth filter
LIM	limit on cycling of controller output
NB	noise band

NLN	switch for nonlinear gain compensation
OVR	user specified overshoot
P	self-tuned proportional band
PF	fixed proportional band
PKi	i^{th} peak of a transient response
TPKi	time that i^{th} peak was located
T	period of oscillation
WMAX	maximum wait time

Chapter 7 Abbreviations

AD	magnitude of automatic set point perturbations
CF	confidence factor
DT	process delay time - user specified
EL	error limit parameter
ER	control error
ID	maximum allowable deviation in process output during "Initial Test"
IF	measurment filter time constant
IM	identification mode
OD	controller output deviation for "Initial Test"
OP	process input
PV	process variable
RD	recommended derivative time
RI	recommended integral time
RP	recommended proportional bank

RT	recommended tuner sample time
TD	fixed derivative time
TI	fixed integral time
TT	tuner sample time
XP	fixed proportional band

1. Introduction

Interest in the design and application of adaptive control systems has continued since the late 1950's motivated by the work of such authors as Kalman(1958), Aström(1967) and many others. The research effort has been particularly strong since the early 1970's with many published reports of successful applications both on pilot and industrial processes.

The industrial demand for robust adaptive and/or self-tuning controllers is growing as plant engineers tackle the problem of controlling time-varying or unknown processes. Several instrumentation companies now market single loop adaptive PID controllers, some of which are gaining widespread acceptance due in part to their improved performance as well as industry's familiarity with the three term controller.

At the same time, both the industrial and academic communities continue research into the development of more general adaptive control schemes. These advanced control schemes can give better performance than conventional PID controllers and offer greater design flexibility. Most, however, have not been accepted for industrial applications due to their relative complexity and unproven performance. Also, many of the implementations have suffered from a lack of practical design considerations to accomodate real process problems and their effects on controller performance.

1.1 Scope and Objectives.

This author takes the view that before attempts to design and implement any particular adaptive control algorithm are made, the functional requirements for the control system should be developed based on industry's needs and the proposed range of applications. The control engineer can then proceed with a comprehensive design. For a particular adaptive algorithm the engineer must understand the control law's structure and the assumptions involved in its derivation. Adjustable parameters must be identified and their effect on closed loop performance understood. The mechanism used for adaptation of adjustable parameters must then be designed or selected for robust long term operation. Using the above requirements and an understanding of the practical problems one can then proceed with the development of a practical adaptive control system for industry.

In this work, two industrial adaptive PID controller's, The Foxboro Company's "Exact" and Turnbull Control System's "Auto-Tuning" controller, were experimentally evaluated and compared with an "academic" adaptive controller. The purpose of these experiments was to identify the controllers' type of self-tuning mechanism, evaluate practical design features and define their useful range of application. These results are presented in Chapters six and seven.

The next objective was the selection of an academic adaptive algorithm as the basis for a practical adaptive controller. The generalized minimum variance (GMV) adaptive predictive controller of Clarke and Gawthrop (1979) was selected for study because of its relative success in past applications, its design flexibility and its simplicity.

Based on the experience gained from the evaluation of the PID controllers, a review of current work in the adaptive area and industrial requirements, the third objective was to design and implement a practical academic adaptive controller. The theoretical background for GMV is summarized in Chapter three. The practical issues related to GMV design such as offset removal, control over closed loop response characteristics and the control of non-minimum phase processes are discussed. The effects of the various adjustable parameters on closed loop performance were investigated and led to the conclusion that GMV can be interpreted as a *fixed gain*, 1/Q controller acting on an adaptive predicted control error. For demonstration purposes, a second self-tuning loop was implemented to adjust 1/Q to give user specifed overshoot for set point control on a nonlinear process. The design of a robust parameter estimation scheme is critical for long term estimation of predicted outputs. Chapter four summarizes the current problems associated with the use of recursive least sqares (RLS) identification and an Improved Least Squares (ILS) algorithm (Sripada, Fisher 1987) is reviewed. For this particular implicit, positional implementation, the problems of disturbance reconstruction and estimation of measurement noise are also addressed. Refer to section 8.1 for a functional description of the "academic adaptive algorithm" or A^3 controller that summarizes its current design features.

Based on an experimental evaluation of the A^3 controller and comparison with the two industrial controllers, the final objective was to summarize the most important aspects of current self-tuning and adaptive controllers in terms of adaptive mechanisms, practical features, performance, ease of use and recommendations for future improvements in design. This is done in Chapters 9 and 10.

2. Adaptive Control: A Functional Specification

2.1 Justification for Adaptive Control.

The primary objective in applying a control scheme to a process is the maintenance of some key variables at a specified set point. The process set point may be fixed as in steady state control or change with time such as in servo control operations. The above goal is often difficult to achieve for many reasons. For example, unmeasured and or uncontrolled disturbances can not be predicted. The static process gain may be a nonlinear function of operating point. Many processes have large and sometimes time-varying transport delays. Plant dynamics can change with time.

Good control of processes that exhibit any of the above characteristics generally requires features specific to the particular application. For example, process nonlinearities make controller tuning a difficult task since any selected set of fixed controller parameters may only be suitable for a narrow operating range due to the destablilizing effect of the changes in process gains. Large changes in set point often require retuning of controller parameters to maintain desired performance levels. Often, controllers are purposely detuned, so that closed loop control will remain stable throughout the operating range of the process being controlled. The tradeoff is between a broad range of stable operation versus a significant reduction in disturbance rejection performance and subsequent servo control.

The solution is to design a controller that can maintain user specified closed loop performance when controlling typical process applications. Such controllers would perform automatic self-tuning in order to minimize the amount of supervision needed for normal operation and improve overall performance. A practical self-tuning controller would enable an engineer to maintain specified levels of performance for a greater number of loops, with less personal effort than that required using conventional tuning practices.

2.2 Requirements.

There is a set of minimum requirements that must be met by any controller that will operate in an industrial environment. Closed loop stability is the absolute minimum followed by the ability to guarantee removal of controller offset.

At a higher level, a controller should be structured so that the adjusted parameters directly affect the performance characteristics of the system's closed loop response to either set point or load disturbances. These characteristics must be defined in terms of a practical, useful user-specified performance index. The adaptive controller could then be used to force the closed loop system to behave in the prespecified manner.

In order to increase a controller's scope of application there should be a provision of features or options designed to handle specific process characteristics. A worst case might be a process that exhibited the following characteristics: nonlinearities in static gain; non-minimum phase processes; time-varying dynamics; unknown transport delay (fixed or varying); unknown high order dynamics; disturbances; and high levels of process and or measurement noise.

The controller should have an "easy to use" process operator interface. Options should be clearly defined and require a minimum amount of process control knowledge to use. In the event of unforeseen problems, the controller should have a backup or default mode that will guarantee stable performance. For example, a self-tuning PID controller should have a stable set of backup PID parameters available for use in the event that the adaptive mechanism produces unsatisfactory controller parameters.

An adaptive controller's design should minimize the amount of expertise and effort required for: controller configuration; selection of features; implementation and specification of desired closed loop performance. The operator interface should provide the user with measured levels of closed loop performance so that the tuner's performance can be quickly assessed and decisions made to modify it, if desired.

2.3 Practical Problems

An adaptive predictive controller can be interpreted as normal feedback control loop coupled with some adaptive mechanism that automatically adjusts parameters used to calculate predicted outputs. The block diagram representation in Figure 2.1 illustrates the general structure of the GMV.

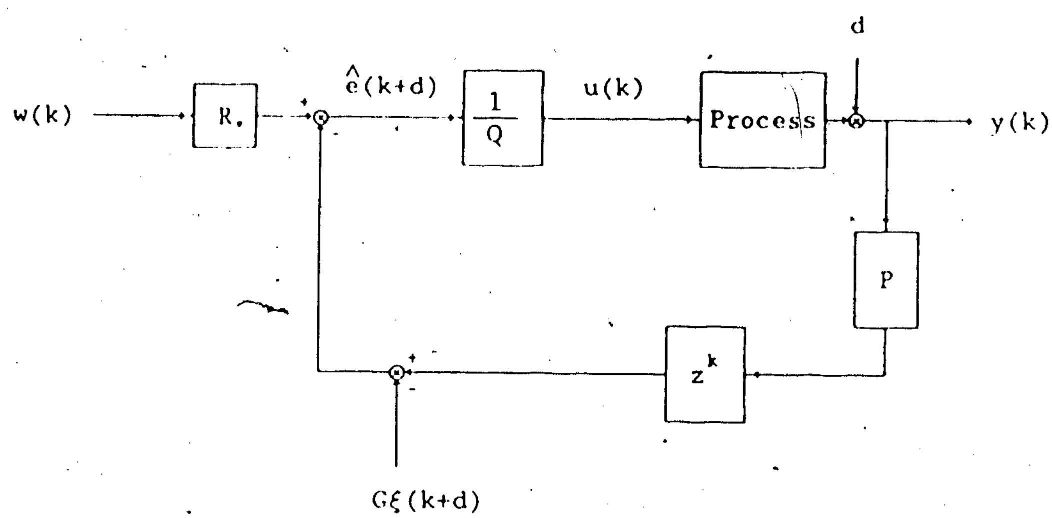


Fig. 2.1 Block Diagram Structure For An Adaptive Predictive Controller

Overall performance is dependent on both the control law structure and the adaptive mechanism. The GMV controller is a predictive form of adaptive controller, ie. the calculation of control action is based on a predicted error which is the difference between the set point and a prediction of the process measurement 'd' steps into the future. This delay term accounts for both process transport and discretization delay. The removal of steady state offset in closed loop control will be accomplished only if two conditions are satisfied. The STC requires the asymptotically correct prediction of the process measurement and a controller structure that will guarantee zero steady offset given that prediction. Steady state offset due to improper control law structure will be termed controller offset. Steady state offset due to a bias in the predicted process output will be called prediction offset. These are the two most important problems to be solved for any adaptive controller: designing a control law structure that will deliver good performance and developing a robust adaptive mechanism suitable for sustained operation over a wide range of process conditions.

Controller Structure.

The generalized minimum variance form of the STC includes discrete transfer functions in the z-domain denoted by $R(z^{-1})$, $P(z^{-1})$ and $Q(z^{-1})$ which permit set point tailoring, filtering of process measurements and filtering of control action. The use of these filtering options dramatically affects the performance of the control law and its useful range of applications. The effect of these weighting or filter functions on controller performance will be summarized below and discussed more fully in Chapter three.

When discrete filtering of control action is not used (ie. $Q(z^{-1})=0$) the controller tends to calculate signals with large variations. The use of $Q(z^{-1})$ can lessen these variations and, if the proper structure is chosen, be used to ensure that there is zero controller offset.

Set point filtering with $R(z^{-1})$ can be used to convert step changes in set point into a form that is more easily followed by the adaptive controller. This leads to slower but smoother changes in operating point and does not require large initial changes in control action.

The use of process measurement filtering, $P(z^{-1})$, can be used to filter the process output prior to calculation of a control signal. Filtering is generally used to force the process to follow a desired reference model given by $1/P(z^{-1})$.

The discrete filters $P(z^{-1})$,$Q(z^{-1})$ and $R(z^{-1})$ should be selected so that: the control law produces reasonable variations in controller output; the necessary conditions for zero controller offset are satisfied; by adjusting parameters within these filters, the characteristics of closed loop dynamic performance can be prespecified and changed at any time, and closed loop stability can be guaranteed.

The performance of adaptive predictive controllers with model based estimation schemes is dependent on their ability to predict the dynamic output, $y(k+d)$, of the process being controlled. The estimation requires input/ouptut data that clearly defines the dynamic behavior of the process. Measurement sampling rates must be selected so that sampled data closely approximates true process dynamics. High frequency cut-off filters should be used to prevent aliasing.

Adaptive controllers generally assume a linearized fixed model structure of low order. In general, as a process becomes more nonlinear, an adaptive controller requires longer periods of identification before the predicted output converges to the true value. Modelling errors can result in biased parameter estimates which in turn lead to inaccurate predictions and possible problems with closed loop stability. To guarantee zero steady state control error, the predicted output must at least asymmptotically approach the true future process output.

Many of the problems found in practical implementations of adaptive controllers are associated with long term steady state operation of the estimation algorithm. The basic recursive least squares algorithm suffers from several practical problems (see Chapter four) which make it unsuitable for practical applications. The estimation algorithm must maintain its ability to adapt to changing process conditions but be able to suspend parameter updates when no useful dynamic information is available in the measured I/O data.

D.c. bias is yet anothe problem for many estimation schemes. D.c. bias is due to nonzero steady state process measurements, unmeasured load disturbances and unmodelled process dynamics. The bias term must be estimated so that accurate predictions can be determined. Its direct inclusion in the parameter identification routine often results in undesirable variations in other parameter estimates and numerical ill-conditioning. The design of the estimation scheme for determining the predicted output should minimize the effects of d.c. bias estimation on other estimated parameters.

The negative effect of unfiltered measurement noise on the predicted output must be minimized or removed. Measurement noise can produce biased parameter estimates which will result in prediction offset.

The recursive estimation schemes make up the adaptive part of most "self-tuning" controllers today and are used to track changing process dynamics so that accurate predictions are maintained. These adaptation mechanisms do not adjust parameters within the discrete filters $P(z^{-1})$, $Q(z^{-1})$ and $R(z^{-1})$ which have a significant effect on closed loop behavior. For this reason a second adaptation mechanism or supervisory scheme is needed to automatically tune parameters within these filters so that the adaptive controller can deliver user specified closed loop performance (refer to section 3.5).

2.4 Features for a Practical Adaptive Controller.

From the preceding discussion of general requirements and common difficulties associated with implementing a practical adaptive control scheme, the following important features can be identified.

1.) The use of a-priori process information should be maximized:

<u>Gain Scheduling</u>: Available steady state input-output data can be used to approximate static process gains. By removing the estimated gain from measured I/O data an adaptive scheme would not have to track large process nonlinearities.

<u>Constraints on Estimated Parameters</u>: If, for a given process it was known that good control performance existed for a particular range of controller parameters, it would be useful if estimated parameters could be constrained within that same region. Closed loop stability and a certain performance level could then be guaranteed. This would be one way to minimize the need for operator supervision.

<u>Open Loop Process Response Data and Signal to Noise Levels</u>: Dominant process time constants can be used by adaptive schemes as an estimate for the time scale of the closed loop system. Known noise levels in measured signals can be useful when choosing filter constants or within estimation algorithms that use ON-OFF criteria.

2.) Features designed to handle specific process conditions should be available:

Detuning Safeguards: When the user specified closed loop performance specifications result in poor control due to unreasonable values or other factors such as modelling errors, controller parameters should be automatically adjusted to improve performance. For example, unreasonable oscillations in controller output or process measurement should force the controller to be detuned. If the controller is unable to detune correctly, the operator should be able to select an option that will immediately suspend adaptation and transfer a set of stable backup parameters to the controller.

Initialization: An option must be provided that will enable the operator to obtain initial values for critical operating parameters and provide for an easy transition from startup to normal adaptive or self-tuning operation. Such initialization data could include initial estimates for controller parameters, process time constant, transport delay and noise levels in measurement signals.

Feedforward Compensation: If measured disturbances are available, an option should be provided that incorporates this information into the overall control scheme.

3.) The acceptance of any controller (adaptive or non-adaptive) in the field will depend on how much effort and skilled knowledge is required to operate it. The controller must be designed to minimize the amount of information and supervision required during normal operation.

3. Implicit Adaptive Predictive Control

Much of current adaptive predictive technology is based on the original minimum variance self-tuning regulator designed by Aström and Wittenmark(1973). The work was extended by Clarke and Gawthrop(1975) to produce the widely known self-tuning controller (STC). This control law received a great deal of attention from the academic community primarily because of its flexibility. Based on the minimization of a general cost function, this formulation allowed the engineer to modify the controller structure using discrete time polynomial transfer functions. This algorithm was again modified by Clarke and Gawthrop(1979) to give a generalized adaptive predictive controller that was more suited to on-line implementation. It is this controller that was used in experimental evaluations and the comparative study with the Turnbull and Foxboro self-tuning PID controllers.

This chapter reviews that controller's structure and the assumptions made in its derivation. The necessary conditions to guarantee offset removal are also discussed in terms of the controller's structure alone. Guidelines for selection of the generalized weighting transfer functions are provided. It is shown that GMV is not a self-tuning controller and requires automatic adjustment of $Q(z^{-1})$ based on some measure of closed loop performance.

3.1 Process Model Assumptions (CARMA model structure)

Given measured input output data for a particular process, an adaptive predictive controller attempts to identify parameters in a dynamic model of the process being controlled. These adapted model parameters are then used to calculate estimates for predicted process outputs.

For this work the process model is given by the following multi-input single-output (MISO) linearized discrete time representation.

$$A(z^{-1})y(k) = B(z^{-1})z^{-d}u(k)+L(z^{-1})z^{-q}v(k)+x(k) \tag{3.1}$$

where

$$A(z^{-1})=1+a_1z^{-1}+a_2z^{-2}+. . . +a_{na}z^{-na}$$
$$B(z^{-1})=b_0+b_1z^{-1}+b_2z^{-2}+. . . +b_{nb}z^{-nb}$$
$$L(z^{-1})=l_0+l_1z^{-1}+l_2z^{-2}+. . . +l_{nl}z^{-nl}$$
$$q \geq d$$

and y(k), u(k) and v(k) are the process measurement, controller output and measured disturbance variables, respectively. Equation (3.1) will be referred to as the correlated, auto-regressive, moving average or CARMA plant model. Process response to changes in controller output is assumed to be delayed by "d" sampling periods and "q" intervals for changes in measured load disturbances. For processes with no inherent delays, a delay of one (1) sampling instant due to discretization is assumed.

The signal x(k) represents a general disturbance or load term which accounts for other effects on the process measurement including: unmeasured load disturbances, steady state bias, stochastic noise and errors due to the linearization of time domain process model. This term, x(k), is defined as:

$$x(k)=C(z^{-1})\xi(k)+\bar{d} \tag{3.2}$$

where

$$C(z^{-1})=1+c_1z^{-1}+c_2z^{-2}+. . . +c_{nc}z^{-nc}$$

and $\xi(k)$ is an uncorrelated random signal (white noise) with zero mean and variance σ^2. It is further assumed that the roots of $C(z^{-1})$ are strictly within the unit circle so that $C(z^{-1})\xi(k)$ is therefore a stationary, moving average process of order "nc".

The second term, $\bar{d}$, represents the previously mentioned factors particularly steady state bias, unmeasured disturbances and the effects of model structure violations.

The inherent assumptions in the model equation are: the process is time invariant, i.e. the parameters in the discrete polynomials $A(z^{-1})$, $B(z^{-1})$, $L(z^{-1})$ and $C(z^{-1})$ are constant; since true processes are rarely linear, it is implicitly assumed that deviations in control signals and process measurements from their mean values are not large, so that the asssumed process model can be considered a local linearization about some fixed operating point; the orders of the discrete polynomials are known and bounded by some finite value and the magnitude of 'd' and 'q' are known.

The above assumptions, since they are violated in most practical applications, significantly affect the applicability of adaptive control schemes.

3.1.1 Comments on the CARIMA Plant Model

As will be shown in section 3.3 the use of the CARMA model structure for process modelling, forces the control engineer to incorporate a special feature for the rejection of unmeasured load disturbances into the control design. From (3.2), $\bar{d}$ must be estimated so that x(k) can be accurately determined. In the implicit schemes, estimates of predicted outputs contain a term which is a function of x(k) (see (3.11)). The estimation of this term, δ, is known as load reconstruction and is a necessary condition for the elimination of prediction offset.

An early solution to the above problem was the formulation of an incremental predictor equation (Clarke, Hodgson and Tuffs 1983) which eliminated the need for load reconstruction to remove prediction offset. This method was followed by that of Tuffs and Clarke (1985) in which the assumed structure of the noise term was given by a nonstationary process, i.e.

$$\Delta x(k)=C(z^{-1})\xi(k)+\Delta\bar{d} \tag{3.3}$$

so that substitution of the above disturbance model into (3.1) gave the controlled autoregressive <u>integrated</u> moving average (CARIMA) plant representation:

$$A(z^{-1})y(k)=B(z^{-1})u(k)z^{-d}+\frac{C(z^{-1})\xi(k)}{\Delta} \quad (3.4)$$

The CARIMA model, through the choice of various forms for $C(z^{-1})$ and assumptions about $\xi(k)$, allows $x(k)$ to be either stochastic or of a random stepwise form. The result of using the above model structure in the implicit GMV scheme can be summarized as follows: no load reconstruction is required to guarantee the elimination of prediction offset; no particular form of $Q(z^{-1})$ filtering is required to guarantee removal of controller offset (see section 3.3) and the resulting integrating STC has excellent disturbance rejection properties, particularly when unmeasured disturbances can be characterized by random steps. It can also be shown that parameter estimates are not affected by step changes in load since data used in recursive estimation is differenced using $\Delta=(1-z^{-1})$.

It is noted that the differencing of measured I/O data introduces additional variance into the recursively estimated parameters. Since the process measurements have zero mean and random variance σ^2 and $Cov[y(k),y(k-1)]=E[y(k)-\bar{y}]\cdot E[y(k-1)-\bar{y}]=0$, it can be shown (Guttman et al. 1982) that:

$$Var(\Delta y(k))=Var(y(k))+ Var(y(k-1)) \cong 2\cdot Var(y(k))$$

The above CARIMA formulation suffers from sensitivity to measurement noise, particularly at steady state operation when nonzero I/O signals are due only to measurement noise.

3.2 Generalized Minimum Variance Predictive Control.

The adaptive predicitve controller used in this study (Clarke et al. 1979) is based on the self-tuning regulator (Aström and Wittenmark 1973) or minimum variance type control. This regulator was based on the minimization of fluctuations in a controlled plant's process measurement as it was affected by random disturbances. The criterion for this regulator, shown below as the minimization of:

$$J=E[(y(k))^2] \tag{3.5}$$

does not include penalties for excessive control action or allow for optimal set point following control.

The self-tuning controller of Clarke and Gawthrop (1975) was based on a cost function incorporating process measurements, set points and controller output shown below:

$$J=E\{[P_n(z^{-1})y(k+d)-R_n(z^{-1})w(k)]^2+[\frac{e_0}{q_0}Q_n(z^{-1})u(k)]^2\} \tag{3.6}$$

The resulting control law allows for optimal tracking of set point changes and/or penalty on controller output using discrete $P_n(z^{-1})$, $Q_n(z^{-1})$ and $R_n(z^{-1})$ polynomials. This controller structure permits the control of stable non-minimum phase processes, without the need for solving a Riccati equation, through the proper choice of control weighting (see section 3.4).

Clarke and Gawthrop's later GMV version (1979) of the self-tuning controller contained several important modifications which further expanded its range of applications. Unlike the 1975 version it allowed for open loop identification of the predictor equation, discrete time transfer functions for $P(z^{-1})$, $Q(z^{-1})$ and $R(z^{-1})$ and their on-line modification.

The self-tuning control law is based on minimization of the cost function:

$$J=E\{[P(z^{-1})y(k+d)-R(z^{-1})w(k)]^2+[\frac{e_0}{q_0}Q(z^{-1})u(k)]^2\} \tag{3.7}$$

where $P(z^{-1})y(k+d)$ represents the future weighted process measurement 'd' sampling steps ahead in time and is defined as:

$$P(z^{-1})y(k+d)=\psi(k+d)$$

and $\psi(k+d)$ can be expressed in terms of known process model parameters using the following Diophantine identity:

$$P(z^{-1})C(z^{-1})=G(z^{-1})A(z^{-1})+z^{-d}\,\frac{F(z^{-1})}{P_d(z^{-1})} \tag{3.8}$$

where

$$G(z^{-1}) = \frac{E(z^{-1})}{B(z^{-1})} = 1+g_1z^{-1}+g_2z^{-2}+\ldots+g_{d-1}z^{-(d-1)}$$

$$F(z^{-1}) = f_0+f_1z^{-1}+f_2z^{-2}+\ \ldots+f_{ni-1}z^{-(ni-1)}$$

$$ni=\max(na+npd,nc+npn-d+1)$$

and

$$P(z^{-1})=\frac{P_n(z^{-1})}{P_d(z^{-1})} \qquad Q(z^{-1})=\frac{Q_n(z^{-1})}{Q_d(z^{-1})} \qquad R(z^{-1})=\frac{R_n(z^{-1})}{R_d(z^{-1})}$$

$$P_n(z^{-1})=P_{n_0}+P_{n_1}z^{-1}+\ldots+P_{n_{npn}}z^{-npn}$$

etc...

The use of this Diophantine identity allows for the separation of the stochastic part of the process model into two sequences of terms: noise terms up to and including time 'k', and future unknown and unpredictable terms. Multiplying (3.1) by $P(z^{-1})z^{+d}$ (and neglecting the measured disturbance term) yields:

$$\psi(k+d)=\frac{PB}{A}u(k)+z^{d}\frac{PC}{A}\xi(k)+\frac{P}{A}\bar{d} \qquad (3.9)$$

Then multiplying (3.8) by $\xi(k)$ and substituting into the above equation gives:

$$\psi(k+d)=\frac{PB}{A}u(k)+z^{d}G\xi(k)+\frac{F}{AP_d}\xi(k)+\frac{P}{A}\bar{d}$$

or

$$\psi(k+d)=\frac{PB}{A}u(k)+\frac{F}{AP_d}\frac{[Ay(k)-Bu(k-d)-\bar{d}]}{C}+\frac{P}{A}\bar{d}+G\xi(k+d)$$

$$\psi(k+d)=Bu(k)[\frac{P}{A}-\frac{Fz^{-d}}{P_dAC}]+\frac{F}{P_dC}y(k)+[\frac{P}{A}-\frac{Fz^{-d}}{P_dAC}]\bar{d}+G\xi(k+d)$$

Using (3.8)

$$C\psi(k+d)=Eu(k)+\frac{F}{P_d}y(k)+G\bar{d}+G\xi(k+d) \quad (3.10)$$

where $E(z^{-1})=G(z^{-1})B(z^{-1})$

At steady state, from the Final Value Theorum (Stephanopoulos 1984) $\delta=G(z^{-1})|_{z=1}\bar{d}$ and $y'(k)=y(k)/P_d(k)$ (3.10) becomes:

$$C\psi(k+d)=Eu(k)+Fy'(k)+\delta+G\xi(k+d) \quad (3.11)$$

or

$$\psi(k+d)=Eu(k)+Fy'(k)+H\psi(k+d)+\delta+G\xi(k+d) \quad (3.12)$$

where

$$H(z^{-1})=c_1z^{-1}+c_2z^{-2}+\ldots+c_{nc}z^{-nc}$$

As can be seen from (3.12) $\psi(k+d)$ is not realizable since the term $G\xi(k+d)$ is unknown. $\psi(k+d)$ is therefore approximated by a prediction of the weighted process measurement as:

$$\psi(k+d)=\psi^*(k+d)|_k+G\xi(k+d) \tag{3.13}$$

where

$$\psi^*(k+d)|_k=Eu(k)+Fy^{n}(k)+H\psi^*(k+d)|_k+\delta \tag{3.14}$$

Substitution of (3.13) into the cost function of (3.7) gives:

$$J=E\{[\psi^*(k+d)|_k+G\xi(k+d)-Rw(k)]^2+\frac{e_0}{q_0}[Qu(k)]^2\} \tag{3.15}$$

which can not be easily minimized as written. Solution of (3.15) requires knowledge of the unconditional probability density function of u(k) (MacGregor 1977). Since this is not known, (3.15) is minimized based on information available up to time 'k'. Rewriting the cost function to account only for control actions taken up to time 'k' and their effect on measurements no farther than 'd' steps ahead in time:

$$J=E\{[\psi^*(k+d)|_k-Rw(k)]^2+\frac{e_0}{q_0}[Qu(k)|_k]^2]\}+\sigma^2 \tag{3.16}$$

and minimizing with respect to u(k) gives:

$$\frac{\partial J}{\partial u(k)}=2[\psi^*(k+d)|_k-Rw(k)]e_0+2\frac{e_0}{q_0}[Qu(k)]q_0=0 \tag{3.17}$$

which can be simplified to give the self-tuning control law

$$Qu(k)=[Rw(k)-\psi^*(k+d)|_k] \quad (3.18)$$

3.3 Requirements for Removal of Controller Offset.

The GMV's control law (3.18) has a fixed structure based on the mathematical minimization of an objective function that contains three, as yet unspecified, discrete transfer functions $P(z^{-1})$, $Q(z^{-1})$ and $R(z^{-1})$. The minimization and therefore the control law is only useful if proper designs of P, Q and R are chosen so that the specified requirements of section 2.2 can be achieved. This issue of proper controller design is independent of the problem of estimating a predicted output term which is itself related to model estimation and discussed in Chapter four.

The control law for the GMV is illustrated using the block diagram structure shown in Figure 3.1.

In this section, the closed loop equation will be derived and the necessary conditions for the removal of controller offset summarized. Consider the general CARMA model structure given as:

$$A(z^{-1})y(k)=B(z^{-1})z^{-d}u(k)+x(k), \quad x(k)=C(z^{-1})\xi(k)+\bar{d} \quad (3.19)$$

and the control law:

$$Qu(k)=[Rw_o(k)-\psi^*(k+d)|_k] \quad (3.20)$$

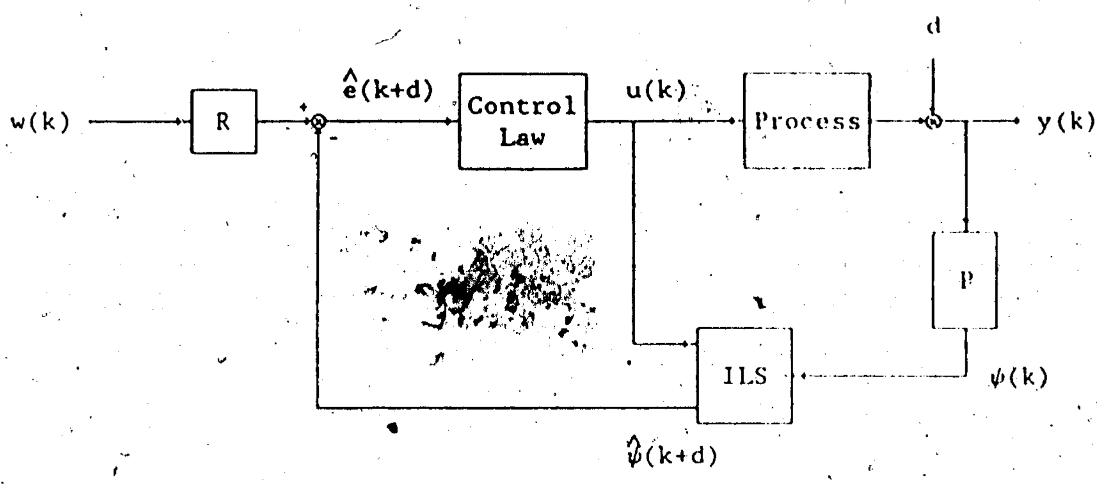


Fig. 3.1 Block Diagram Structure for a GMV Adaptive Predictive Controller

along with the prediction model:

$$\psi(k+d)|_k=\frac{F}{P_d}y(k)+Eu(k)+G\bar{d}+G\xi(k+d) \qquad (3.21)$$

where $C(z^{-1})$ has been assumed=1 for simplicity.

Assuming that the process model is known, substitution of (3.19) and (3.20) into (3.21) yields the following closed loop equation:

$$y(k+d)[PB+QA]=RBw(k)+\xi(k+d)[Q+E]-Q\bar{d}z^{-d} \qquad (3.22)$$

In practical applications, the removal of controller offset implies that the process measurement asymptotically equals the desired set point. From (3.22) it can be shown that this will be the case for all unmeasured $\bar{d}$ only if:

$$Q(z^{-1})|_{z=1}=0.0$$

so that (3.22) simplifies to:

$$y(k)Pz^{d}=Rw(k)+G\xi(k) \tag{3.23}$$

or at steady state

$$\bar{y}P(1)=R(1)\bar{w}$$

It is apparent that two conditions must be satisfied for the proper choice of controller design:

1. $Q(1)=0.0$
2. $R(1)=P(1)$

Note that the predicted output term $\psi(k+d)|_k$ is itself a function of u(k) which has been ignored in the above discussion since the issue of controller offset is being addressed. The issue of prediction offset discussed in section 4.3 imposes additional constraints on the discrete transfer functions which are summarized below.

3. $R(1)=P(1)=1$
4. $P_n(1)=1$

3.4 P, Q and R Weighting - Practical Interpretations.

The basic control law of (3.18) contains three discrete transfer function filters which can be designed to affect the closed loop performance of the controller. This section reviews several problems common to most academic controllers: the stabilization of non-minimum phase processes and the tailoring of closed loop transients through the proper design of discrete filters.

3.4.1 Non-minimum phase Systems.

Non-minimum phase processes are those processes that contain transport delay or right half plane zeros in the 's' domain. In the following discussion, non-minimum phase will refer to those discrete time processes that contain zeros outside the unit circle.

The implementation of digital control requires discrete sampling of plant I/O data. In an effort to improve control, many applications have attempted to use faster sampling rates. Unfortunately, this may unexpectedly result in unstable control. Minimum phase plants, when discretely sampled, may become non-minimum phase. Aström, Hagander and Sternby (1984) demonstrated that when the number of continuous time poles exceeds the number of continuous time zeros by two or more, there always exists a small enough sampling interval that will result in a non-minimum phase discrete system. In practical terms this means that any plant that exhibits phase lag > 180° at high frequencies may become non-minimum phase if sampled too quickly. Goodwin et al. (1986) proposed the use of a "delta" operator to overcome the problem of creation of non-minimum phase discrete systems.

General guidelines suggest a sampling interval based on an open loop response to a step change in manual controller output, i.e.:

$$T_s = 0.1\tau_p$$

Similarly, Isermann (1982) suggests that control performance is not very sensitive to the choice of sample time and recommends:

$$\frac{1}{15}T_{95} \leq T_s \leq \frac{1}{4}T_{95}$$

where T_{95} is the 95% settling time of the process transient response.

The selection of longer sampling intervals may prevent the "creation" of discrete non-minimum phase zeros, but it usually does so at the expense of control performance. A proper solution to the problem is the selection of discrete filters that will enable faster sampling to be used for a given process.

3.4.2 $Q(z^{-1})$ - Weighting on Control Action.

The stability of any closed loop system is determined by the poles of its characteristic equation. The closed loop equations of interest are:

$$y(k+d)[CPB+QA]=RBw(k)+C\xi(k+d)[Q+E]-Q\bar{d}z^{-d} \qquad (3.24)$$

$$u(k)[CPB+QA]=ARw(k)-\frac{F}{P_d}C\xi(k)-PC\bar{d} \qquad (3.25)$$

From (3.24) above the characteristic equation is:

$$CPB=-QA \qquad (3.26)$$

If P=Q=1, the closed loop system may become unstable if either $B(z^{-1})$ or $A(z^{-1})$ contain roots outside the unit circle. However, closed loop stability can be assured through the proper choices for $P(z^{-1})$ and/or $Q(z^{-1})$. Assuming that the open-loop process is stabilizable (i.e. unstable modes are controllable and observable), Clarke (1984) suggests using $Q(z^{-1})$ for stabilization and $P(z^{-1})$ for model following purposes (see discussion on the use of $P(z^{-1})$). If the unstable roots of $B(z^{-1})$ were due to the effects of fast sampling, one alternative would be to decrease the sample rate and then adjust $Q(z^{-1})$ to compensate for the unstable roots, if any, of $A(z^{-1})$. Another alternative suggested by Clarke (1984) is the use of a factorization technique that allows for the maintenance of closed loop stability when very small amounts of $Q(z^{-1})$ weighting are used. The trade off is a considerable increase in complexity of the control calculation.

As previously discussed, a necessary condition for the removal of controller offset was that $Q(z^{-1})|_{z=1}=0$. The simplest choice of $Q(z^{-1})=0$ gives minimum variance control with set point following when $P(z^{-1})=1$ and $R(z^{-1})=1$. The controller attempts to drive the process measurement to the desired set point in no less than 'd' sample intervals. The closed loop equation of (3.24) reduces to:

$$y(k)= w(k-d)+\xi(k)G(z^{-1})$$

or

$$y(k+d)= w(k)+G(z^{-1})\xi(k+d) \tag{3.27}$$

and shows that any controller offset will be due to the effects of stochastic noise. If the process gain is large then (3.25) also indicates that set point changes will produce large deviations in control action particularly at higher sample rates where the parameters in $B(z^{-1})$ will be even smaller in magnitude. From (3.26) the characteristic equation reduces to:

$$C(z^{-1})B(z^{-1})=0.0 \tag{3.28}$$

which indicates that closed loop stability will depend on the roots of $B(z^{-1})$. This demonstrates that minimum variance controllers can not stabilize plants that are non-minimum phase either in the continuous time domain or the discrete time domain due to fast sampling rates. In the latter case the engineer's only alternative is to decrease the sample rate until the roots of $B(z^{-1})$ lie within the unit circle. With PID controllers, the same applies. Either sampling time must be increased so that the controller does not "see" the wrong way response or controller gain must be decreased so that u(k) will not change during the non-minimum phase behavior.

When nonzero control weighting of the form:

$$Q(z^{-1})=\lambda \tag{3.29}$$

is included, the characteristic equation becomes:

$$C(z^{-1})B(z^{-1}) = -\lambda A(z^{-1}) \tag{3.30}$$

and if $B(z^{-1})$ contains unstable roots, λ can then be increased so that the R.H.S. term will dominate so that the roots of the characteristic equation will lie within the unit circle. Of course, there will be controller offset, but the system will be stable. The closed loop equation reduces to:

$$y(k)[CB+\lambda A]=Bw(k-d)+C\xi(k)[Q+E]-\lambda\bar{d}z^{-d}$$

or

$$y(k+d)=\frac{B}{CB+\lambda A}w(k)+\frac{C\lambda}{CB+\lambda A}\xi(k+d)+\frac{CE\xi(k+d)}{CB+\lambda A}-\frac{\lambda\bar{d}}{CB+\lambda A} \qquad (3.31)$$

which, for large λ becomes:

$$y(k+d)\cong\frac{B}{A\lambda}w(k)+\frac{C}{A}\xi(k+d)+\frac{CE}{A\lambda}\xi(k+d)-\frac{\bar{d}}{A} \qquad (3.32)$$

Equation (3.32) shows that λ will act as a filter on changes in set point. For open loop stable processes it intuitively makes sense that the closed loop system will be easier to stabilize if only small changes in u(k) are allowed (cf. low controller gain). This illustrates that $Q(z^{-1})$ weighting can be used to decrease the variance of controller output.

The next obvious choice for $Q(z^{-1})$ that would satisfy the necessary condition for offset removal and allow for the stabilization of non-minimum phase systems is an integrator of the form:

$$Q(z^{-1})=\lambda(1-z^{-1}) \qquad (3.33)$$

Simulation results by Clarke (1984) and preliminary experimental work by this author indicate that the use of this integrating $Q(z^{-1})$ does remove controller offset and allow for stabilization of non-minimum phase systems. An undesirable effect, however, was the sometimes oscillatory control action and accompanying overshoot (see Section

8.2.3).

From the predictive control law:

$$u(k)=\frac{[Rw(k)-\psi^{*}(k+d)|_{k}]}{Q(z^{-1})} \qquad (3.[illegible])$$

and an interpretation of the numerator of the R.H.S. term as filtered, predicted control error, GMV can be considered an adaptive predictive, fixed parameter integral controller. The above interpretation suggests that $1/Q(z^{-1})$ have the structure of a PI controller so that the resulting control weighting will decrease the oscillatory behavior of u(k) and give faster closed loop responses. The desired form of $Q(z^{-1})$ is therefore:

$$Q(z^{-1})=\frac{\lambda(1-z^{-1})}{q_{d0}-q_{d1}z^{-1}} \qquad (3.35)$$

The corresponding proportional gain and integral constants are easily determined by comparing (3.35) with the velocity form of a PI controller corresponding to:

$$u(k)=u(k-1)+K_c\left(1+\frac{T_s}{\tau_I}\right)e(k)-K_c e(k-1) \qquad (3.36)$$

The effective gain and integral constant for (3.35) are:

$$K_c=\frac{q_{d1}}{\lambda}$$

$$\tau_I=\left[\frac{q_{d1}}{(q_{d0}-q_{d1})}\right]\cdot T_s$$

Simulated results (Clarke 1984) indicate improved dynamic performance when inverse PI filtering of u(k) is used over the integrating form. Clarke's formulation used fixed values of 1.0 for q_{d0}, 0.5 for q_{d1} and λ was manually adjusted to demonstrate that smaller values resulted in less weighting or filtering of controller output.

The above $Q(z^{-1})$ design increases the GMV's scope of application and its usefulness as a practical controller. It meets the necessary conditions for removal of controller offset, contains an adjustable parameter that can be used to stabilize non-minimum phase systems and has advantages over the integrating form in that larger λ values (less penalty on control) can be used with less overshoot for transient responses.

Because process dynamics change with operating region and time, it is possible that for a given sampling interval a once stable system can become unstable due to the appearance of non-minimum phase roots in $B(z^{-1})$. In order to improve the robustness of the controller it would be ideal if parameters in $Q(z^{-1})$ were automatically adjusted in order to maintain stability when such problems arose.

One approach (Tham 1985) considers the characteristic equation when $C(z^{-1})=R(z^{-1})=P(z^{-1})=1.0$ and $Q(z^{-1})$ is given by (3.35) as:

$$B + QA = 0.0$$

or

$$BQ_d + Q_nA=0.0 \tag{3.37}$$

For the implicit formulation, $A(z^{-1})$ and $B(z^{-1})$ can be related to estimated parameters of $F(z^{-1})$ and $E(z^{-1})$ so that (3.37) becomes:

$$Q_dE + Q_n[1.0-Fz^{-d}]=0 \quad . \tag{3.38}$$

Since the actual calculation of u(k) involves division by $[\hat{e}_0 + Q]$, Tham suggested that q_{d0} be chosen as $1/\hat{e}_0$ so that the leading coefficient would always be unity, thereby decreasing the sensitivity of the controller to small values of $\hat{e}_0$. The value of q_{d1} was then determined by solving (3.38) for a desired dominant pole "p". As pointed out by Tham, the specification of the closed loop pole was exact only when $\deg(Q_dB)=\deg(Q_nA)=1$, which is rarely the case.

Another approach is to maintain K_cK_p at a constant value by adjusting controller gain in response to changes in estimated process gain. With explicit schemes, $A(z^{-1})$ and $B(z^{-1})$ are recursively identified so that process gain estimates can be determined by:

$$\hat{K}_p=\frac{B(1)}{A(1)}$$

at every sampling interval. Because parameters in $A(z^{-1})$ are generally much smaller than those in $B(z^{-1})$ (due to magnitude of static process gain), K_p can be sensitive to variations in $A(z^{-1})$ or $B(z^{-1})$. Without perfect identification it is difficult to guarantee that the above approach will give reasonable estimates for K_p.

In this work, a nonlinear gain compensation option was implemented to maintain K_cK_p at a constant value K'_c. However, process gains were approximated using user specified steady state input/output data, i.e.:

$$K_p = \frac{\Delta y_{ss}}{\Delta u_{ss}}\Big|_{w(k)}$$

To minimize computational effort, K_p was determined for current set point (steady state compensation) rather than process output (dynamic gain compensation).

3.4.3 $P(z^{-1})$ - Model Reference Control.

Model reference adaptive controllers (M.R.A.C.) attempt to force the closed loop system to follow the dynamic behavior of a prespecified reference model. An adaptive mechanism monitors the difference between the transient response of the reference model and the closed loop system, and automatically adjusts its controller constants to minimize that difference. A block diagram for a typical M.R.A.C. is shown in Figure 3.2.

Through the proper choice of discrete filters the GMV can

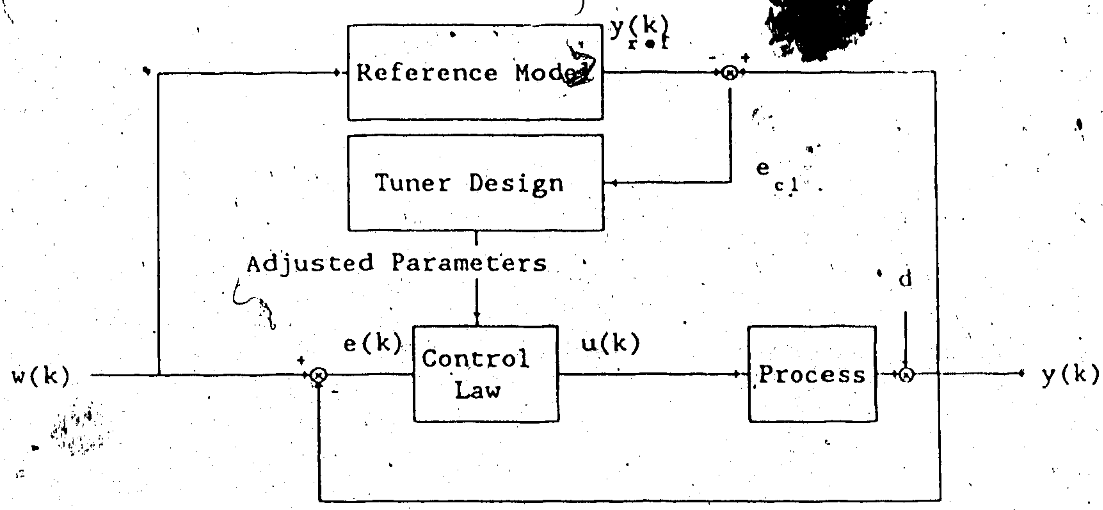


Fig. 3.2 Block Diagram for a Model Reference Adaptive Controller

also take the form of a M.R.A.C. Consider the case when Q=0, G=1 and P=1/M, where $M(z^{-1})$ is some prespecified reference model. The closed loop equation can now be written as:

$$y(k)=\frac{w(k-d)}{P}+\frac{G}{P}\xi(k)$$

or

$$y(k)=M(z^{-1})w(k-d)+GM(z^{-1})\xi(k) \qquad (3.39)$$

Equation (3.39) offers an important advantage over many M.R.A.C. schemes. When $P(z^{-1})$ is chosen as an inverse model, disturbance rejection also follows the reference model. The disturbance affects the system as the moving average prediction error passing through the filter $1/P(z^{-1})$ (neglecting the effect of noise).

Gawthrop (1977) suggested using a discretized model of a unity gain first order process as a reference model $M(z^{-1})$. The first order time constant would be of the order of the open loop time constant but somewhat faster. Since $M(z^{-1})$ is essentially a process with phase lag, $P(z^{-1})$ adds phase advance to the system which should have a stabilizing influence.

It should be noted that like minimum variance control, this scheme involves the cancellation of plant zeros. If $B(z^{-1})$ contains unstable roots this form of M.R.A.C. would not be practical.

Gawthrop (1977) demonstrated through simulations the practical problem of pure model following control for a second order process with no plant zeros given by:

$$y(s)=[(s+0.1)(s+1.0)]^{-1}[e^{-s}u(s)+\xi(s)] \tag{3.40}$$

When reference models were used which contained less excess of poles over zeros than the actual process, closed loop control was oscillatory and suffered from intersample ringing. In practical applications where model orders are unknown, pure model following may not be feasible since it will be impossible to guarantee that the specified reference model's excess of poles over zeros will be at least equal to that of the real plant.

Since model following can be interpreted as a form of differentiation, the use of $P(z^{-1})$ might not be suitable for measurement signals with high noise levels.

3.4.4 $R(z^{-1})$ - Set Point Filtering.

An alternative to the above model reference type of control is the use of $R(z^{-1})$ to implement set point tailoring. For the special case $Q(z^{-1})=0.0$, $C(z^{-1})=P(z^{-1})=1.0$, the closed loop equation of (3.24) reduces to:

$$y(k)=R(z^{-1})w(k-d)+G(z^{-1})\xi(k) \tag{3.41}$$

which is much like (3.39) for pure model following except that the rejection of disturbances will no longer follow the reference model. Also note that if $P(z^{-1})=1.0$ there will be no way for the engineer to stabilize non-minimum phase systems other than to decrease the sampling rate. Because $R(z^{-1})$ does not enter the characteristic equation, its use has no effect on closed loop stability. From (3.8) the order of $R(z^{-1})$ can be modified on-line with no effect on the number of estimated parameters. After startup, however, $\bar{P}(z^{-1})$'s order can not be changed, although its coefficients can be modified. Recall that the order of $F(z^{-1})$, ni, is given as: " max[na+npd,nc+npn-d+1] ".

The primary reason for the use of set point filtering is that it makes excursions in control action smaller and smoother since the effective set point change for a given sample instant is decreased. The smoother transition between operating points will improve the recursive estimation of parameters since data in the I/O regressor vectors will not contain large variations. These large variations can result in undesirable changes in estimated parameters and even ringing control action, particularly when data differencing or averaging is used.

3.4.5 GMV - A Fixed Gain Adaptive Predictive Controller

Discussion in the previous sections has shown that the GMV controller's effect on closed loop performance varies with the choice of P, Q and R. In theory, minimum variance or one step ahead control gives the fastest closed loop response to set point or load disturbances, with no penalty on control action considered in J. Such control is not usually practical. Calculated controller output is sensitive to the variance of parameter estimates and the accuracy of predicted outputs. Long periods of open loop identification are required to get converged estimates of parameters used to calculate predicted outputs, so that the closed loop system will remain stable. Closed loop stablility may be affected by sampling rates and may suffer from intersample ringing. The lack of filtering on u(k) usually results in unacceptably vigorous control.

By including a penalty on control, the cost function J is minimized to give a simple control law structure (R=1=P):

$$Qu(k)=[w(k)-\psi^{*}(k+d)|_{k}] \qquad (3.42)$$

which can be interpreted as a 1/Q control law acting on predicted error, e(k+d). This practical implementation of GMV requires: the choice of $Q(z^{-1})$ to get the desired control law and the accurate estimation of predicted control error, e(k+d).

Considerations have shown that $Q(z^{-1})|_{z=1}=0$ is a requirement. A simple integrating form given by $\lambda(1-z^{-1})$ is adequate but involves a direct tradeoff between speed of closed loop response and degree of damping as in classical integral only control. 1/Q=PI control is the next logical step. An inverse PI structure for $Q(z^{-1})$ gives a variable rate of weighting on control. The effective proportional action gives fast response to set point changes and the integral term guarantees controller offset elimination. User specified gain and integral time are explicitly related to parameters in $Q(z^{-1})$, allowing for easy on-line modification.

For the simple case of proportional action on predicted error, $Q=1/\lambda$, the GMV effectively gives a proportional derivative control law given by:

$$u(k)=\frac{1}{\lambda}[w(k)-y(k+d)]$$

$$=\frac{1}{\lambda}[e(k)]+\frac{1}{\lambda}[\Delta_d e(k)]$$

Using this argument, it follows that the use of a predictive PI controller offers phase lead over the classical PI control law and should give better control performance, much like PID.

The use of higher order weighting would give additional control design flexibility at the expense of computation time and effort. Furthermore, as the order of $Q(z^{-1})$ is increased, the degree of differentiation increases. This may not be practical for measured signals that are corrupted by noise.

Overall performance is dependent upon the accuracy of prediction and therefore the predicted set point error. For nonlinear or time varying processes, adaptive modelling is required to track changing dynamics. Such estimation will give better performance by providing the control equation with correct values for predicted output.

The use of $Q(z^{-1})$ for weighting on control action and $R(z^{-1})$ for filtering of user specified set point changes allow for the "control" of closed loop performance. $Q(z^{-1})$ directly affects how the controller output responds to changes in predicted set point error and can affect closed loop stablility. Set point filtering affects closed loop performance indirectly, by preventing large $\Delta w(k)$ from entering the system directly. Since GMV is a derivative of 'd' step ahead control, the filtering of the set point relaxes the requirement of driving the process output to desired level in 'd' steps. This allows for smoother transitions to new operating levels. For larger filter time constants, a form of model following results for set point transients while smaller values simply modify the closed loop response to set point changes.

The above interpretation of a $1/Q(z^{-1})$ control law acting on a predicted control error term leads one to conclude that GMV is a fixed structure, constant gain control law with adaptive prediction used to estimate y(k+d). The controller is not self-tuning. The GMV is adaptive only in the sense that parameters used to calculate y(k+d) are recursively estimated and changes in their values due to changing model dynamics, tracked with time.

For the case $1/Q(z^{-1})=PI$, P=R=1, GMV reduces to a PI, adaptive predictive controller differing from classical PI control in two respects. It uses predicted error rather than current measured values. Parameters used to calculate that predicted error are recursively updated to account for changing process dynamics.

3.5 Automatic Closed Loop Adaptation of $Q(z^{-1})$ - PI Formulation

The use of filtering for controller output directly affects closed loop stability and dynamic closed loop performance. In practical applications, fixed $Q(z^{-1})$ is not appropriate due to the effects of process nonlinearities on closed loop control. When an inverse PI structure is chosen for $Q(z^{-1})$, it intuitively makes sense that if static process gain increases, controller gain must be decreased to maintain a certain stability margin. When $Q(z^{-1})$ is fixed, GMV is a non self-tuning 1/Q controller acting on an adaptive predicted set point error. In this work, λ of (3.35) was unity so that:

$$q_{d1}=K_c$$

$$q_{d0}=K_c[1 + \frac{T_s}{\tau_I}] \tag{3.43}$$

For a given sample time, T_s, $Q(z^{-1})$ contains two adjustable parameters, q_{d0} and q_{d1}, that are explicitly related to specified PI controller parameters. Using this formulation allows for on-line modification of $Q(z^{-1})$ using conventional tuning rules for a PI controller.

For demonstration purposes, a closed loop adaptive scheme was implemented to automatically adjust $Q(z^{-1})$ to give user specified overshoot for set point transients. Using a fixed reset time, K_c was adjusted in response to changes in process dynamics or user specified overshoot. Likewise, $Q(z^{-1})$ was initialized using the Cohen and Coon process reaction curve method to determine effective controller gain and reset time. Refer to Chapter 8 for a discussion of that algorithm's performance.

3.6 Feedforward Compensation

The process model given by (3.1) includes a term that accounts for the effects of a measurable disturbance on open loop dynamics given by:

$$L(z^{-1})z^{-q}v(k) \tag{3.44}$$

Using the same arguments as in section 3.2 it can be shown that the predicted output equation (3.12) becomes:

$$\psi(k+d)=Eu(k)+F'y(k)+H\psi(k+d)+G\bar{d}+G\xi(k+d)+GLv(k+d-q) \tag{3.45}$$

From (3.45), feedforward compensation is simply the accounting of measurable disturbances and their effect on process dynamics into the predictor equation. Note that for (3.45) to be realizable, the condition $q \geq d$ must be met or predicted outputs will be a function of future disturbance terms which are not known. There is a corresponding increase in the number of parameters to estimate which will increase the number of updates of $\theta(k)$ required for the adaptive mechanism to give a converged prediction. The actual control law is unaffected and additional measurable disturbances can be included in (3.45) if available.

As mentioned in section 3.1 the assumed delay of measured disturbances must be greater than or equal to that of the manipulated variable or (3.45) will not be realizable since "future" measured disturbances are not available.

3.7 Certainty Equivalence Principle

The discussion in Chapter three has dealt with the subject of controller design and choice of discrete filters for a known process model given by (3.1) and (3.2). For the implicit GMV, the predictive equation was derived as:

$$C\psi(k+d)=Eu(k)+Fy'(k)+\delta+G\xi(k+d) \qquad (3.46)$$

which is a weighted prediction of future process measurements based on information up to and including time 'k'.

In practice the true parameters of the assumed model are unknown and possibly time-varying so that they must be approximated on-line using some parameter estimation technique. The Certainty Equivalence Principle involves the assumption that those estimated parameters are the "true" values and can be directly used within the adaptive predictive scheme. For implicit control the unknown parameters of $E(z^{-1})$,$F(z^{-1})$ and $C(z^{-1})$ along with δ are estimated and then used to calculate an approximate value for $\psi(k+d)$ given by:

$$\hat{\psi}(k+d)=\hat{E}u(k)+\hat{F}y'(k)+\hat{H}\psi(k+d)+\delta+\epsilon(k+d) \qquad (3.47)$$

where $\hat{E}$,$\hat{F}$,$\hat{H}$ and δ must be obtained from the following regression model:

$$\psi(k)=\hat{E}u(k-d)+\hat{F}y'(k-d)+\delta+\hat{A}\psi(k-1)|_{k-d-1}+\hat{\epsilon}(k) \qquad (3.48)$$

Rewriting (3.48) in vector canonical form as:

$$\hat{\psi}(k)|_{k-d}=x^T(k-d)\theta(k)+\hat{\epsilon}(k) \qquad (3.49)$$

$$\hat{\psi}(k)|_{k-d}=x_0^T(k-d)\theta_0(k)+\hat{e}_0u(k-d)+\hat{\epsilon}(k) \qquad (3.50)$$

where

$$\theta(k)=[\hat{e}_1,\hat{e}_2,\ldots,\hat{e}_{nE},\hat{f}_0,\hat{f}_1,\ldots,\hat{f}_{nF},\hat{c}_1,\hat{c}_2,\ldots,\hat{c}_{nc},\delta,\hat{e}_0] \qquad (3.51)$$

$$\begin{aligned}x(k-d)=[&u(k-d-1),u(k-d-2),\ldots,u(k-d-nE),\\&y'(k-d),y'(k-d-1),\ldots,y'(k-d-nF),\\&\hat{\psi}(k-1)|_{k-d-1},\hat{\psi}(k-2)|_{k-d-2},\ldots,\hat{\psi}(k-1-nc)|_{k-d-nc-1},\\&1,u(k-d)]\end{aligned} \qquad (3.52)$$

and

$nE=d-1+nb$

$nF=ni-1$

$ni=\max(na+npd,nc+npn-d+1)$

defines in a compact notation the identification problem to be solved. In subsequent discussions, x and θ will be referred to as the input/output regressor vector and estimated parameter vector, respectively.

The actual implementation of GMV using the estimated predicted output becomes:

$$Rw(k)-\hat{\psi}(k+d)|_k=Qu(k) \tag{3.53}$$

where $\hat{\psi}(k+d)|_k$ is determined using $x(k)$ and $\hat{\theta}(k)$. Note that $x(k)$ contains the unknown value of $u(k)$ so that $\hat{\psi}(k+d)|_k$ is not explicitly calculated until after (3.54) yields the current controller output.

$$u(k)=\frac{[Rw(k)-x^T{}_0(k)\hat{\theta}_0(k)]}{[\hat{e}_0+Q]} \tag{3.54}$$

4. On-Line Estimation Techniques

It was shown in section 3.6 that some form of parameter estimation is required to estimate the coefficients of polynomials in the implicit prediction equation, (3.14), which in turn are a function of the process model (3.1). Because many process plants are nonlinear and time-varying the chosen estimation scheme must be able to track variations in process dynamics over long periods of operation.

Most adaptive control schemes that use a parametric model representation use a recursive form of parameter estimation update known as least squares (RLS). In this chapter the least squares algorithm will be presented. The common problems associated with its practical use will be summarized and many of the ad hoc solutions presently used to overcome them, reviewed.

A new "Improved Least Squares" (ILS) algorithm by Sripada and Fisher (1987a,b) will be discussed that solves many of the above practical problems yet still maintains the numerical properties of the RLS algorithm. Practical guidelines for its use are also given.

Finally, the conditions required for the removal of prediction offset will be summarized.

A detailed discussion of recursive least squares identification including its derivation, practical problems and current solutions is provided in (Fisher and Minter 1987).

4.1 Recursive Least Squares Identification (RLS).

4.1.1 RLS Formulation.

With recursive least squares identification, estimates of θ of (3.49) are calculated every sampling instant based on previous estimates and the latest measured I/O data. In practice, RLS identification allows for the specification of initial estimates for θ given by $\theta(0)$. These initial estimates may be determined a priori using off-line modelling procedures.

The least squares method of solution for (3.49) is based on the minimization of the costing function:

$$J_s = \frac{1}{2}\sum_{k=1}^{N} [y(k) - x^T(k)\theta(k)]^2 + \frac{1}{2}(\theta - \theta(0))^T P(0)^{-1}(\theta - \theta(0)) \quad (4.1)$$

where the second term of (4.1) accounts for the error in assumed initial estimates. The initial magnitude of the covariance matrix, P(0), indicates the degree of uncertainty in the initial parameter estimates, $\theta(0)$.

The RLS algorithm is given by the following update equations:

$$\theta(k) = \theta(k-1) + \frac{P(k-1)x(k)}{[1 + x^T(k)P(k-1)x(k)]} [y(k) - x^T(k)\theta(k-1)] \quad (4.2)$$

and

$$P(k)=P(k-1)-\frac{P(k-1)x(k)x^T(k)P(k-1)}{1+x^T(k)P(k-1)x(k)} \tag{4.3}$$

where y(k) is assumed to be given by:

$$y(k)=x^T(k)\hat{\theta}(k)+\xi(k) \tag{4.4}$$

The least squares algorithm presented is one of a number of parameter estimation routines whose recursive update takes the form of (4.2). RLS will give unbiased estimates for the parameter elements of θ provided that (4.4) is satisfied and $\xi(k)$ is an uncorrelated sequence of terms with zero mean and variance σ^2.

4.1.2 Summary of RLS Problems and Current Solutions.

The basic RLS algorithm will asymptotically give unbiased estimates for θ provided that:

1. process model structure is correctly assumed.
2. the noise term $\xi(k)$ is uncorrelated with measured I/O data.
3. measured I/O data is persistently excited.

In practical applications, virtually all of these conditions are violated to some degree. Many controlled plants are difficult to model and therefore model structures are only approximations, and linear at that. Adaptive controllers such as the GMV are a combination of feedback control law and self-tuner. External signals such as measurement noise become correlated with controller output due to feedback of corrupted process measurements. The resulting estimation errors are therefore not independent of measured I/O data and so biased parameters estimates should be expected. The primary objective of most control schemes is the regulation of some plant output about a desired set point at steady state. This results in a lack of persistent excitation in measured I/O data which can result in failure of the RLS algorithm.

Other problems also arise due to the effects of finite word lengths on numerical conditioning and algorithmic stability. The covariance matrix can become non-positive definite, a theoretical impossibility, when parameter estimates converge and Tr[P(k)] becomes small. Large time delays and fast sampling times can make the condition number of P(k) become very large, resulting in poor solutions for θ.

The limitations of the RLS algorithm in its basic form are listed below along with the techniques currently used to overcome them. Refer to (Fisher and Minter 1987) for a more detailed discussion and literature review.

Basic RLS Algorithm.

Problem: Identification turns off due to Tr[P(k)] $\rightarrow 0.0$.

Identification can become unstable due to the formation of a negative definite P(k) (Carlson 1973). The negative definite problem is due to the effects of numerical round off errors during long periods of steady state operation.

Solution: Discount old I/O data by using a forgetting factor $\lambda(k)$ to prevent P(k) from getting too small (Clarke and Gawthrop 1975). Use special factorization techniques for P(k) to minimize the effects of numerical round off errors on the accuracy of solutions (Potter 1963, Bierman 1976). Vogel and Edgar (1982) suggested the addition of a term to P(k) whenever changes in parameter estimates were detected. Periodic resetting of P(k) to prespecified values was suggested by Goodwin and Sin (1984) to maintain adaptability.

RLS with $\lambda(k) \leq 1$

Problem: Fixed forgetting factors with values <1 result in exponential growth of P(k) during periods of low excitation (steady state). Large covariance matrices can result in unwanted parameter drift and sensitivity to changes in process conditions.

Solution: Use a variable forgetting factor whose value is based on the amount of information content entering the ID problem (Fortescue 1981). During periods of low excitation, $\lambda(k) \to 1$. This ensures, for the stochastic case, that $P(k)$ will not grow exponentially.

RLS with Variable Forgetting Factor $\lambda(k)$

Problem: When measured I/O signals are corrupted by noise, $Tr[P(k)]$ can still grow exponentially. Fortescue's algorithm will maintain $\lambda(k)<1$ since RLS estimation errors will never be exactly zero due to noise. As a result, $\theta(k)$ will drift which can lead to bursting phenomena and poor closed loop control (Anderson 1985).

Solution: Place hard limits on $Tr[P(k)]$ and use covariance resetting to maintain boundedness. Remove measurement noise with analog filtering of $y(k)$. Disable identication when process I/O data is predominantly noise.

Any of the above:

Problems: Correlated noise sequences result in biased $\theta(k)$.

Solution: ELS (Panuska, 1969), GLS(Clarke 1967, Hasting-James and Sage 1969) used to ident $C(z^{-1})$. This requires an estimate of $\xi(k)$ which is unknown and generally estimated using high pass filtering of previous estimation errors, $\hat{e}(i)$.

Any of the above:

Problem: Biased $\theta(k)$ due to nonzero mean data. Requires estimation of the nonzero steady state mean output, $\bar{d}$. The "1 in the data vector" technique suffers from practical problems related to poor conditioning due to the relatively large magnitude of $\bar{d}$. RLS can produce unwanted variations in $\theta(k)$ due to the effects of unmeasured step disturbances entering the system.

Solution: Use independent estimation of the means and conditioning of measured I/O data.

Any of the above:

Problem: Lack of persistently exciting measured I/O signals results in random drifting of $\theta(k)$ sometimes into unstable regions, which can result in bursting.

<u>Solution:</u> Maintain excitation by adding external perturbations or disable identification when no useful dynamic information is present in I/O signals. Use filtering to remove noise from I/O signals to prevent parameter drift.

<u>Any of the above:</u>

<u>Problem:</u> Large process time delays plus fast sampling results in poor conditioning and more parameters to estimate.

<u>Solution:</u> Careful selection of sampling interval used for identification.

From the previous discussion it should be clear that one of the most important problems associated with RLS is its continued use with nonpersistently excited input signals and the possibility of bursting. The incorporation of such an identification algorithm within an adaptive controller could result in sudden undesirable closed loop performance and possible instability.

While the use of an external signal to satisfy conditions for persistent excitation is a possible solution it is not the most desirable alternative. It would be more desirable to incorporate some type of ON-OFF criteria into the RLS algorithm so that identification would be disabled when the system was not persistently excited. The improved least squares algorithm presented in the next section (Sripada, Fisher 1987a) meets this objective.

4.2 Improved Least Squares (ILS)

An improved least squares (ILS) algorithm (Sripada and Fisher 1987b) for recursive parameter estimation addresses the problem of identifying θ during periods of low excitation. The ILS scheme includes: an ON-OFF criteria that disables update of θ when $x(k)$ is not persistently excited and thus prevents parameter drift and bursting; a variable forgetting factor that maintains $Tr[P(k)]$ at a user specified value; filtering and normalization of I/O data to improve accuracy of $\hat{\theta}(k)$, and scaling of the I/O regressor vector to minimize the condition number of $P(k)$.

The ILS algorithm retains the exponential rate of convergence of $\hat{\theta}(k)$ to θ and is able to track slowly time varying processes due to its constant trace feature. Internal checks on numerical conditioning are performed that improve overall performance of the estimator.

4.2.1 Important Properties of ILS

The initial data conditioning steps used in this algorithm are a necessary part of any identification scheme in improving numerical conditioning. The scaling of x(k) can be used to minimize numerical ill-conditioning and therefore provide more accurate solutions.

The constant trace feature implemented by varying the amount of forgetting avoids the need for special factorization techniques to guarantee positive definite P(k). The user can effectively modify the algorithm's tracking ability by inflating or deflating P(k) at the expense of higher variance in $\theta(k)$ for the former case. The problem of estimator turn-off is avoided and the need for ad hoc solutions such as covariance resetting avoided.

The ON-OFF criteria can solve the problems of parameter drift and bursting due to continued identification with nonpersistently exciting signals. Using a user specified lower bound on $\|P(k-1)x(k)\|$ and an upper bound on C[P(k)], the ILS algorithm will only allow parameter estimates to be updated if enough new information is present and the problem is not too ill-conditioned.

4.3 Rejection of Prediction Offset

In the implicit GMV scheme the most critical requirement for the recursive estimation scheme is the exact prediction of future process outputs, at least asymptotically. Exact predictions during transient responses are only possible when the exact process model is known, noise is uncorrelated and the plant is not subjected to unmeasured disturbances, ie. simulation studies. In practical applications these factors degrade the performance of any identification scheme. In particular, feedback of measurement noise, results in prediction errors which become correlated with subsequent calculated control signals. Likewise, unmeasured disturbances if not estimated result in prediction offset. This section addresses the problems of load reconstruction and the a-priori estimation of nonzero mean noise.

4.3.1 Load Reconstruction

Conditions required to guarantee the removal of controller offset were summarized in Chapter three based on the assumption that y(k+d) was exactly known at time step k. However, in practice the disturbance term is not known and hence must be estimated. Consider the block diagram in Figure 4.1 that incorporates the prediction equation (3.12) into its structure.

The closed loop equation for this system is:

$$y(k)\left[\frac{(Q+E)Az^{d}}{B}+\frac{F}{P_d}\right]=Rw(k)+\frac{z^{d}(E+Q)}{B}x(k)-G\bar{d} \qquad (4.5)$$

where $x(k)=C(z^{-1})\xi(k)+\bar{d}$. In order to eliminate prediction offset, i.e. ensure that:

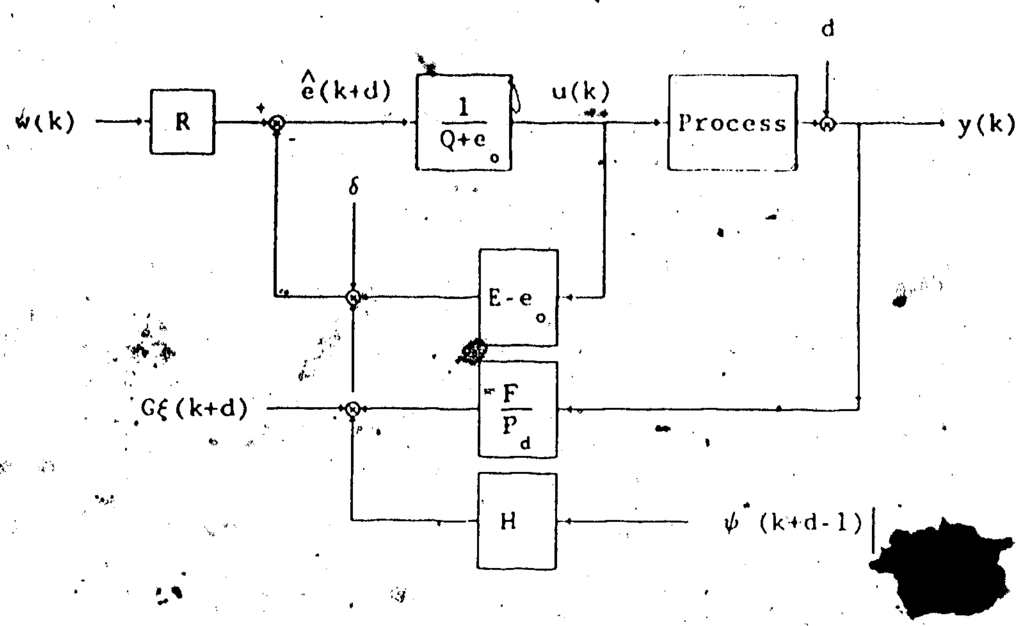


Fig. 4.1 Block Diagram of GMV Control Law

$$\lim_{k\to\infty} y(k)=\bar{y}(k)=\lim_{k\to\infty} w(k)=\bar{w}(k) \tag{4.6}$$

The steady state closed loop equation is written as:

$$\bar{y}\left[\frac{(Q(1)+E(1))A(1)}{B(1)}+\frac{F(1)}{P_d(1)}\right]=R(1)\bar{w}+\frac{E(1)+Q(1)}{B(1)}\bar{d}-G(1)\bar{d} \tag{4.7}$$

From section 3.3 the conditions for removal of controller offset were:

1. $Q(1)=0.0$
2. $R(1)=P(1)$ (4.8)

Substitution of (4.8) into the steady state closed loop equation gives:

$$\bar{y}\left[\frac{\hat{E}(1)A(1)}{B(1)}+\frac{F(1)}{P_d(1)}\right]=R(1)\bar{w}+\frac{E(1)}{B(1)}\bar{d}-G(1)\bar{d} \qquad (4.9)$$

The last term in (4.9), $G(1)\bar{d}$ is unknown but in order to satisfy (4.6) $\hat{E}(1)/B(1)\ \bar{d} = \delta$ implies that an exact estimate of $G(1)\bar{d}$ is required. The second condition is that:

$$\frac{E(1)A(1)}{B(1)}+\frac{F(1)}{P_d(1)}=R(1)$$

or

$$G(1)A(1)+\frac{F(1)}{P_d(1)}=R(1) \qquad (4.10)$$

Comparing (4.10) above with (3.8) shows that this relationship will be satisfied since $R(1)=P(1)$.

4.3.2 A-Priori Estimation of Nonzero Mean Noise

The exact prediction of the future weighted output term $Py(k+d)$ is given by (3.12) and repeated for clarity as:

$$\psi(k+d)=Eu(k)+\frac{F}{P_d}y(k)+H\psi(k+d)+G\bar{d}+G\xi(k+d) \qquad (4.11)$$

Since $\xi(k+d)$ is not known, $\psi(k+d)|_k$ can only be approximated using:

$$\psi^*(k+d)|_k=\psi(k+d)|_k-G\xi(k+d) \qquad (4.12)$$

The calculated prediction, $\hat{\psi}^{*}(k+d)|_{k}$ is based on estimates of $E(z^{-1})$, $F(z^{-1})$ and $H(z^{-1})$ and $G(z^{-1})\bar{d}$ obtained using some least squares estimation scheme on:

$$\psi(k)|_{k-d}=\hat{E}u(k-d)+\frac{\hat{F}}{P_d}y(k-d)+\hat{H}\psi(k)|_{k-d}+G\bar{d}+G\xi(k) \qquad (4.13)$$

Assuming that exact estimates for E,F,H and $G\bar{d}$ are available, (4.13) can be written in vector form as

$$Py(k)=\theta^{T}x(k)+e(k) \qquad (4.14)$$

where e(k), the prediction error will be equal to $G\xi(k)$. In order to ensure that parameter estimates will be unbiased, e(k) must be approximated prior to parameter update at each time step by e'(k). Note that $\bar{e}(k)$ is the mean value of $\hat{e}(k)$, the prediction error, given by:

$$\hat{e}(k)=Py(k)-\theta^{T}x(k) \qquad (4.15)$$

Note that $\bar{e}(k)$ is unknown and must be estimated. Two possible solutions are provided:

1. Assume that e(k)=0.0
2. Estimate e(k) using previous prediction errors. This would involve low pass filtering of prediction errors using:

$$\bar{e}(k)=\beta\bar{e}(k-1)+(1-\beta)\hat{e}(k) \tag{4.16}$$

and $e'(k)=\hat{e}(k)-\bar{e}(k)$ would be the estimate for $G\xi(k)$. Note that if $\beta=0$ then $e'(k)=0$ which corresponds to (1) above. The adjustable parameter, β, is a forgetting factor. During long periods of steady state operation when prediction errors are due to measurement noise, a β value 0.990-0.999 is recommended. This range is not appropriate when prediction errors are due to actual changes in process dynamics, however, since if $\beta=0.999 \rightarrow \bar{e}(k) \simeq \hat{e}(k)$ which will prevent parameter updates.

4.4 Use of Mean Deviation Data - Implementation Details

This section summarizes important implementation details related to the use of zero mean data with the positional implicit GMV used in this work. As discussed in section 4.2 the use of zero mean data in recursive least squares identification improves numerical conditioning and allows for faster rejection of unmeasured disturbances. Drifting disturbances can be independently estimated, thereby decreasing unwanted variations in $\theta(k)$ by using an estimation technique such as the one in the data vector technique.

Steady state bias is removed from I/O data prior to identification so that the zero initial condition assumption of (3.1) is not violated and $\bar{d}$ is not included in $\theta(k)$.

4.4.1 Modified Predicted Output Equation

Consider the linear process model:

$$A(z^{-1})y(k)=B(z^{-1})u(k)z^{-d}+C(z^{-1})\xi(k)+\bar{d} \tag{4.17}$$

which at steady state becomes:

$$A\bar{y}=B\bar{u}+\bar{d} \tag{4.18}$$

The d.c. bias term is eliminated by subtracting (4.17) from (4.18) to give:

$$A(y-\bar{y})=B(u-\bar{u})z^{-d}+C\xi(k)$$

or

$$A\hat{y}=B\hat{u}z^{-d}+C\xi(k) \tag{4.19}$$

The process model (4.19) above can be used in explicit control schemes or used to derive a modified predictor equation suitable for implicit identification as follows:

Multiply (4.19) by $P(z^{-1})z^{d}$ to yield:

$$P\hat{y}(k+d)=\frac{PB}{A}\hat{u}(k)+\frac{PC}{A}\xi(k+d) \tag{4.20}$$

Now multiply the Diophantine identity:

$$PC=GA+z^{-d}\frac{F}{P_d} \quad \text{where } G=\frac{E}{B}$$

by $\xi(k)z^{d}$ and substitute into (4.20) to give:

$$P\tilde{y}(k+d)=\frac{PB}{A}\tilde{u}(k)+\frac{F}{AP_d}\xi(k)+G\xi(k+d)$$

or upon substitution of (4.19) for $\xi(k)$

$$P\tilde{y}(k+d)=\frac{PB}{A}\tilde{u}(k)+\frac{F}{AP_dC}[A\tilde{y}(k)-B\tilde{u}(k)z^{-d}]+G\xi(k+d) \quad (4.21)$$

Equation (4.21) can be simplified to yield the predicted deviation in weighted process measurement from some future average value:

$$CP\tilde{y}(k+d)=E\tilde{u}(k)+\frac{F}{P_d}\tilde{y}(k)+G\xi(k+d) \quad (4.22)$$

The formulation of (4.22) allows for estimation of $E(z^{-1})$, $F(z^{-1})$ and $C(z^{-1})$ with no risk of bias due to nonzero steady state levels. For estimation purposes, (4.22) is rewritten as:

$$\psi^{*}(k)|_{k-d}=E\tilde{u}(k-d)+\frac{F}{P_d}\tilde{y}(k-d)+H\psi^{*}(k)|_{k-d} \quad (4.23)$$

so that the actual deviation is:

$$\psi(k)|_{k-d}=\psi^{*}(k)|_{k-d}+G\xi(k) \quad (4.24)$$

As outlined in section 4.3 values for $G\xi(k)$ must be approximated so that parameter estimates will not be biased.

4.4.2 Calculation of Absolute Prediction

When zero mean data are used for identification in the implicit GMV, the data used in the I/O regressor vector $(\bar{u}(k-d)\ldots, \bar{y}(k-d)\ldots, \bar{\psi}(k-1))$ can not be used to directly calculate the absolute prediction $\psi(k+d)$.

It can be shown that:

$$\psi(k+d)|_k=\bar{\psi}(k+d)|_k+\tilde{\psi}(k+d)|_k \tag{4.25}$$

where

$$\bar{\psi}(k+d)|_k=P\bar{y}(k+d)=\frac{F}{P_d}\bar{y}(k)+E\bar{u}(k)+H\bar{\psi}(k+d)|_k+G\bar{d} \tag{4.26}$$

and $G\bar{d}=\delta$ is the load term that must be approximated to guarantee rejection of prediction offset. Since $\tilde{\psi}(k+d)$ can be calculated using the same information in the I/O regressor used for identification, $\psi(k+d)$ can be determined by calculation of $\bar{\psi}(k+d)|_k$ and summation with $\tilde{\psi}(k+d)|_k$. There are two options available for the calculation of $\bar{\psi}(k+d)|_k$:

1. $\bar{\psi}(k+d)$ can be determined using:

$$\bar{\psi}^*(k+d)|_k=\frac{F}{P_d}\bar{y}(k)+E\bar{u}(k)+H\bar{\psi}(k+d)|_k+\delta \tag{4.27}$$

where δ, an estimate of $G\bar{d}$, is obtained using a-priori estimation:

$$\delta(k)|_{k-1}=\bar{\psi}(k)-\bar{\psi}^*(k)|_{k-d} \tag{4.28}$$

This option is in theory the correct method to use but would require considerable additional storage and computation to maintain vectors of filtered mean data, plus the load reconstruction step (4.28) to guarantee no offset.

2. The second method, an approximation, is based on the assumption that $\bar{\Psi}(k)$ changes very slowly. Therefore, $\bar{\Psi}^*(k+d)$ can be approximated by $\bar{\Psi}(k)$, the actual value of the weighted average process measurement. Using this approximation significantly reduces computational requirements and eliminates the need for load reconstruction since $\bar{\Psi}(k)$ contains unmeasured disturbances. The trade-off is the degradation in the quality of absolute predictions during transients when the assumption that $\bar{\Psi}(k) \simeq \bar{\Psi}(k+d)$ is weakened.

Preliminary experimental runs indicated that the approximation does not significantly affect performance. In practice, the forgetting factors used for mean estimation are $\simeq$ 0.75 to 0.9 (window lengths of 4-10) and, as a result, calculated means do not change a significant amount over the period of "d" steps. Clearly if a process has a very large transport delay, method 2 would not be valid, and $\bar{\Psi}(k)$ would have to be predicted using (4.27) and load reconstruction. It should be noted that in all of the experimental runs performed in this work, process measurement noise was assumed to be random so that $C(z^{-1})=1$. Based on initial open loop runs for both pilot plants used in this study, it was observed that process measurements exhibited only random variations about mean levels.

4.4.3 Calculation of Controller Output

The control law equation (3.18) upon substitution of the expression for $\psi(k+d)$ becomes:

$$Qu(k)=-[\bar{\psi}^*(k+d)|_k+\psi^*(k+d)|_k]+Rw(k) \tag{4.29}$$

or using the approximation for $\bar{\psi}^*(k+d)$:

$$Qu(k)=-[\bar{\psi}(k)+\psi^*(k+d)|_k]+Rw(k) \tag{4.30}$$

Noting that $\psi^*(k+d)$ is itself a function of $\tilde{u}(k)$, ie.:

$$\psi^*(k+d)|_k=\hat{F}\frac{\tilde{y}(k)}{P_d}+(\hat{E}-e_0)\tilde{u}(k)-H\psi(k+d)|_k+\hat{G}\xi(k+d)+e_0\tilde{u}(k) \tag{4.31}$$

where $\tilde{u}(k)$ is given by:

$$\tilde{u}(k)=u(k)-\bar{u}(k)$$

or upon substitution of (4.6):

$$\tilde{u}(k)=u(k)-\lambda_u(k)\bar{u}(k-1)-[1-\lambda_u(k)]u(k)$$

or

$$\tilde{u}(k)=\lambda_u(k)[u(k)-\bar{u}(k-1)] \tag{4.32}$$

Equation (4.30) upon substitution of (4.31) and (4.32) can be formulated to give an explicit expression for u(k) as:

$$u(k)=[-\bar{\psi}(k)-\frac{\hat{F}}{P_d}\hat{y}(k)-(\hat{E}-\hat{e}_0)\hat{u}(k)+\hat{A}\hat{\psi}(k+d)|_k - \hat{G}\xi(k+d)+e_0\lambda_u(k)\bar{u}(k-1)+Rw(k)]/[q_0+\hat{e}_0\lambda_u(k)] \quad (4.33)$$

The reader should note that the calculated controller output u(k) in (4.33) may be sensitive to changes in $\lambda_u(k)$, depending on the relative magnitudes of q_0 and $\hat{e}_0(k)$.

A particular problem related to the use of the calculated forgetting factor λ(k) from the ILS algorithm in estimation of $\bar{u}(k)$ was observed. Following a transient response to a set point change that demonstrated good control performance (no offset and slight overshoot), data in the I/O regressor vector quickly approached zero since at steady state $y(k)=\bar{y}(k)$ and $u(k)=\bar{u}(k)$. The least squares problem then became numerically ill-conditioned, but due to inappropriate specification of limits on Tr[P(k)], ‖P(k-1)x(k)‖ and C[P(k)], identification was not suspended. As a result, the constant trace algorithm for λ(k) produced forgetting factors that varied by relatively large amounts over each sample interval. These variations in λ(k) resulted in random perturbations in u(k). The problem was corrected by specifying a smaller limit for Tr[P(k)] and more conservative limits on C[P(k)] and ‖P(k-1)x(k)‖. The use of λ(k) from ILS identification for "independent" estimation of $\bar{u}(k)$ and $\bar{y}(k)$ effectively "couples" the identification with the control law. If the above problem can not be solved through the choice of limits on the adjustable parameters, $\lambda_u(k)$ and $\lambda_y(k)$ should be specified using some other technique or at least bounded by lower limits.

5. Preparation for Experimental Evaluations

To perform real-time experimental evaluations of adaptive controllers, important groundwork must first be carried out. This Chapter contains a description of the pilot scale plants used in this work and discusses why they were selected for testing of the adaptive controllers. The use of a cascaded control scheme for one of the plants is discussed and the performance of the inner loop is described. The majority of the experimental evaluation work involved use of a personal computer system with a real-time multi-tasking operating system. General details of the personal computer system, and applications software used, are presented.

5.1 Process Equipment

Many of the important issues in adaptive control are related to how true plant processes differ from the simplified models typically used within adaptive control schemes. Process models are almost always linear, lower order approximations of the true plant and hence result in unmodelled dynamics. Additional factors such as unmeasured disturbances and process measurement noise compound the problem further. The two pilot scale plants used in this work were selected because they demonstrate some of these nonideal features.

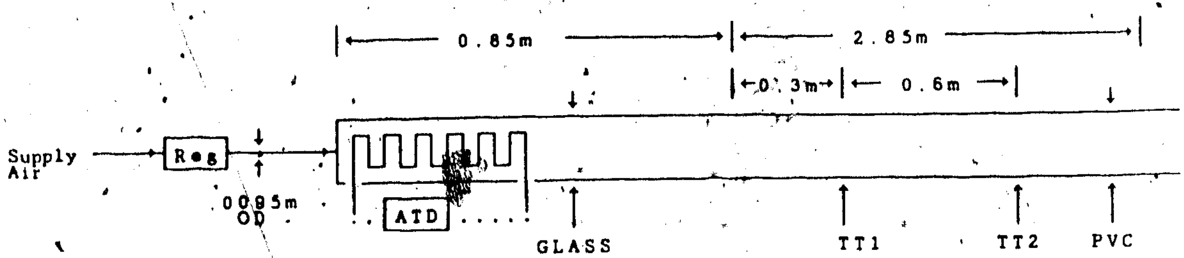
5.1.1 Nonlinear Process - Temperature Controlled Pilot Plant

Temperature controlled plants are typically nonlinear because heat transfer coefficients are nonlinear functions of temperature and velocity, which results in variable process gains and directionally varying dynamics.

For the majority of this work, the pilot plant shown in Figure (5.1) was used. The downstream air temperature was controlled by manipulating the air flow rate past the heating coil as shown. The heating coil was supplied by a 120 volt autotransformer operating at 60 volt (50%) or 84 volt (70%) output. The air flow rate was regulated using a 0.0127m ID reverse acting, equal percentage Foxboro flow control valve. Temperature was measured using a type J thermocouple inserted through polyvinyl chloride plastic (PVC) wall. The supply pressure of air to the system was regulated to 7 kPa gauge. This process was selected because it was highly nonlinear and also included very slow dynamics which showed up as bias or drift.

5.1.1.1 Cascade Control Scheme

Initially the above process was to be controlled using a simple feedback control loop in which the flow rate of air was manipulated to effect changes in downstream temperature. It was difficult to determine process steady state gain values due to the effects of a sticking valve stem. Figure (5.2) shows the open loop response of this process to a series of manual increases in air flow rate (reverse acting valve). With each



LEGEND
ATD - Autotransformer (60 volt supply)
PVC - 0.0381m ID polyvinyl chloride pipe
REG - Pressure regulator (0-200 kPa range)
TTi - Type J thermocouple

Fig. 5.1 Schematic of Temperature Controlled Pilot Plant

increase, the process temperature rose up to a maximum value after which, further increases in air flow rate resulted in decreasing temperature. Note that the process gain reverses sign at a particular operating point. With these initial runs, the process air supply pressure was approximately 30 kPa gauge. By decreasing this supply pressure to 7.0 kPa the operating range of the process was maintained within a region of positive

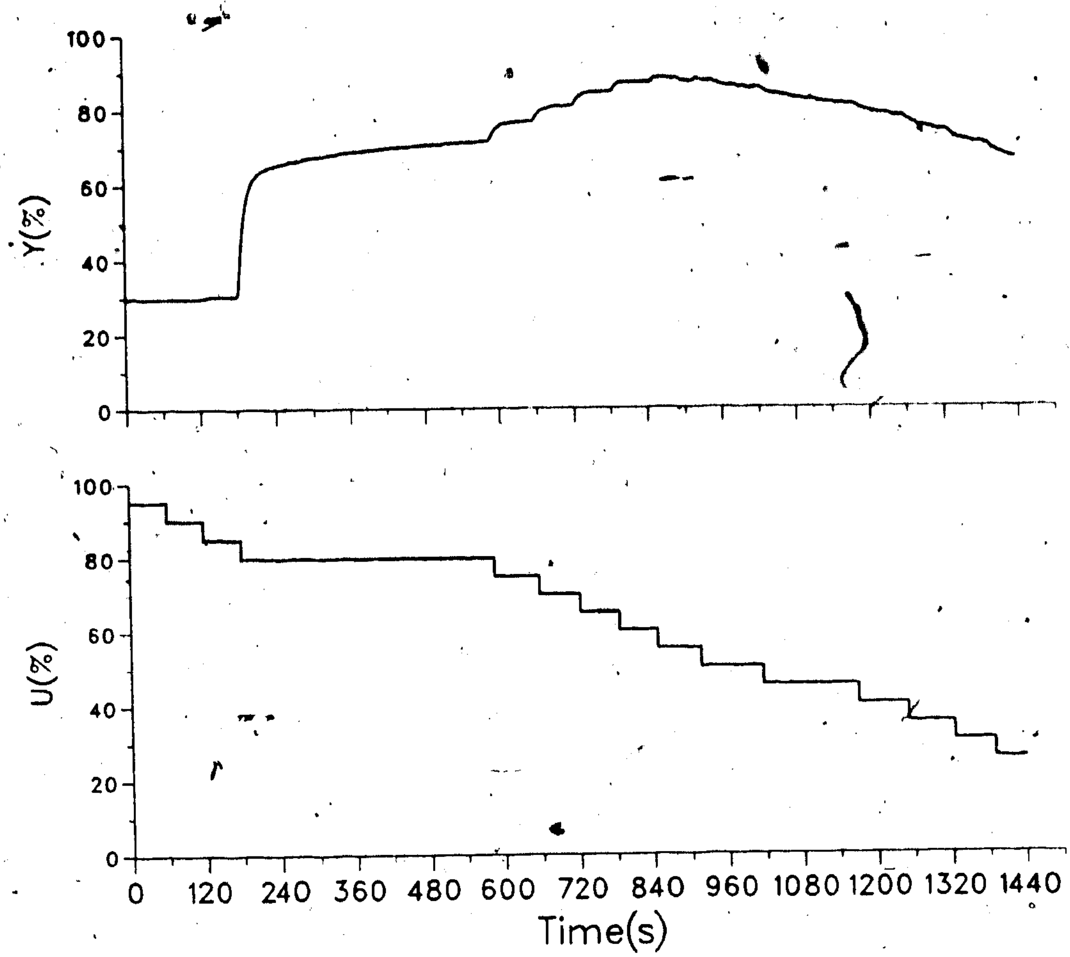


Fig. 5.2 Open Loop Response of Temperature Controlled Pilot Plant (Single Feedback Loop Configuration)

steady state gain.

An alternative control scheme was chosen in which temperature control was cascaded to air flow rate control. Refer to the block diagram in Figure (5.3). Using this cascaded arrangement the sticking valve stem effects could be removed from the temperature control loop. Details of the experimental equipment and operating ranges are presented below.

Experimental Equipment

AT Variable autotransformer 0-120 V., with a maximum current of 10.0 amperes at 50Hz., KVA 1.4.

EMF Foxboro EMF converter, model M/693AT-0A-SU, calibrated for Type J thermocouples within a range of 20-60°C with a 1-5V. output signal and built in linearization.

FT Foxboro electronic transmitter, model E13DM-SAM2, calibrated output signal 10-50mA for a volumetric flow rate range of $0-19.8 \times 10^{-3}$ m^3/s.

FC Inner, air flow rate controller: Turnbull Control System's model 6350, fixed PID controller.

I/P Fisher-Governor Co., Electronic-Pneumatic transformer, Type 546, with a 4-20mA input signal and supply pressure of 138 kPa.

CV Foxboro flow control valve, type V-4, C-3, 0.0127m ID, reverse acting (air to close), 21-103 kPa operating range.

PR Fisher Controls pressure regulator, type 64-27, 0.0035m orifice, calibrated for 0-200 kPa range.

TC Temperature controller. One of either: Foxboro's Exact, TCS's Auto-Tuning or the A^3 controller.

TT Type J copper/constantan thermocouple.

<u>Operating Range for Temperature Process Equipment</u>

Process temperature: 20-60 degrees Celsius.

Air flow rate: $0\text{-}20 \cdot 10^{-3}$.

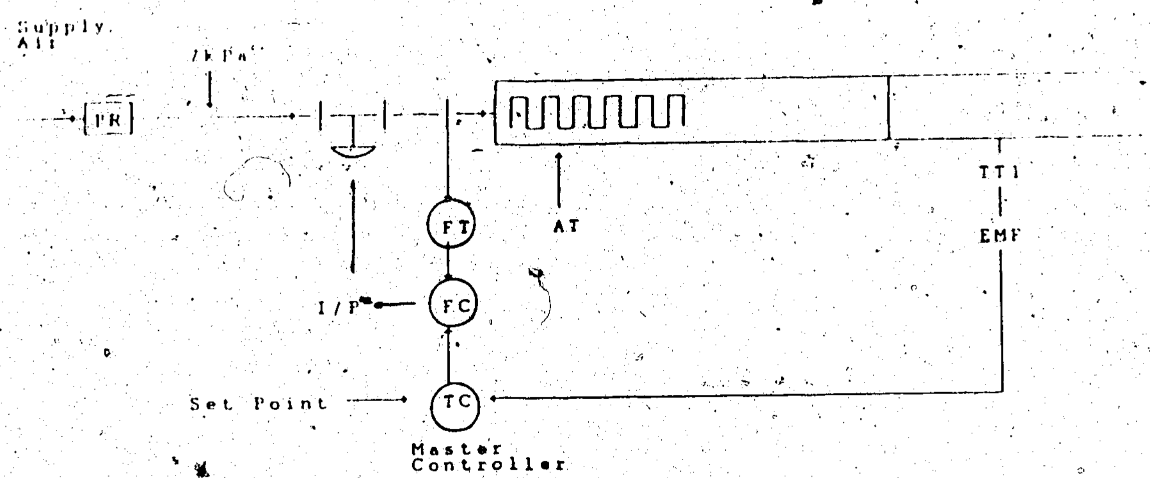
FC Output: 99.99-0% (corresponding to above air flow rates).

TC Output: 0-100% (forward acting control).

Emf Output: 1-4.8 Volts.

FT Output: 1-5 Volts.

I/P Output: 103-21 kPa (corresponding to above air flow rates).



LEGEND

AT - Autotransformer
EMF - Emf Converter
FC - Air flow rate controller (inner loop)
FT - Air flow rate transmitter
PR - Pressure regulator
TC - Temperature controller (master)
TT1 - Type J thermocouple

Fig. 5.3 Schematic of Cascaded Control Scheme

5.1.1.2 Slave Loop Performance

For experimental comparison purposes, a simple feedback control loop was not adequate because the sticking stem affected the repeatability of results. For a given controller output signal, a particular air flow rate could not be guaranteed.

These effects were particularly apparent during the open loop identification tests when a positive step change in control signal was sent to the control valve; the process temperature allowed to approach steady state; and then the controller output returned to its original value. The valve stem did not return to its original position. The result was the final steady state temperature was not equivalent to its original value. The results from initial open loop identification tests were not consistent and could not be compared. For the same reasons it would have been impossible to compare the three controllers' closed loop performances without the cascaded arrangement. This type of operational problem can not be handled by the type of adaptive or self-tuning controller used in this study, because the effect is a random event with respect to both time and magnitude.

This section demonstrates the above problem using a manual run with the original single feedback loop. Figure (5.4) illustrates how the manual changes in air flow rate control signal (to 50%) do not produce consistent steady state process measurements.

<u>Air Flow Rate Control</u>

By having the air flow rate controlled by an inner loop the effects of valve nonlinearities are effectively removed from the outer loop. This allowed the adaptive controllers to tune based on the true process dynamics excluding those due to the valve. Figure (5.5) illustrates the closed loop performance of the air flow control loop for a 10% change in air flow rate to 20% of

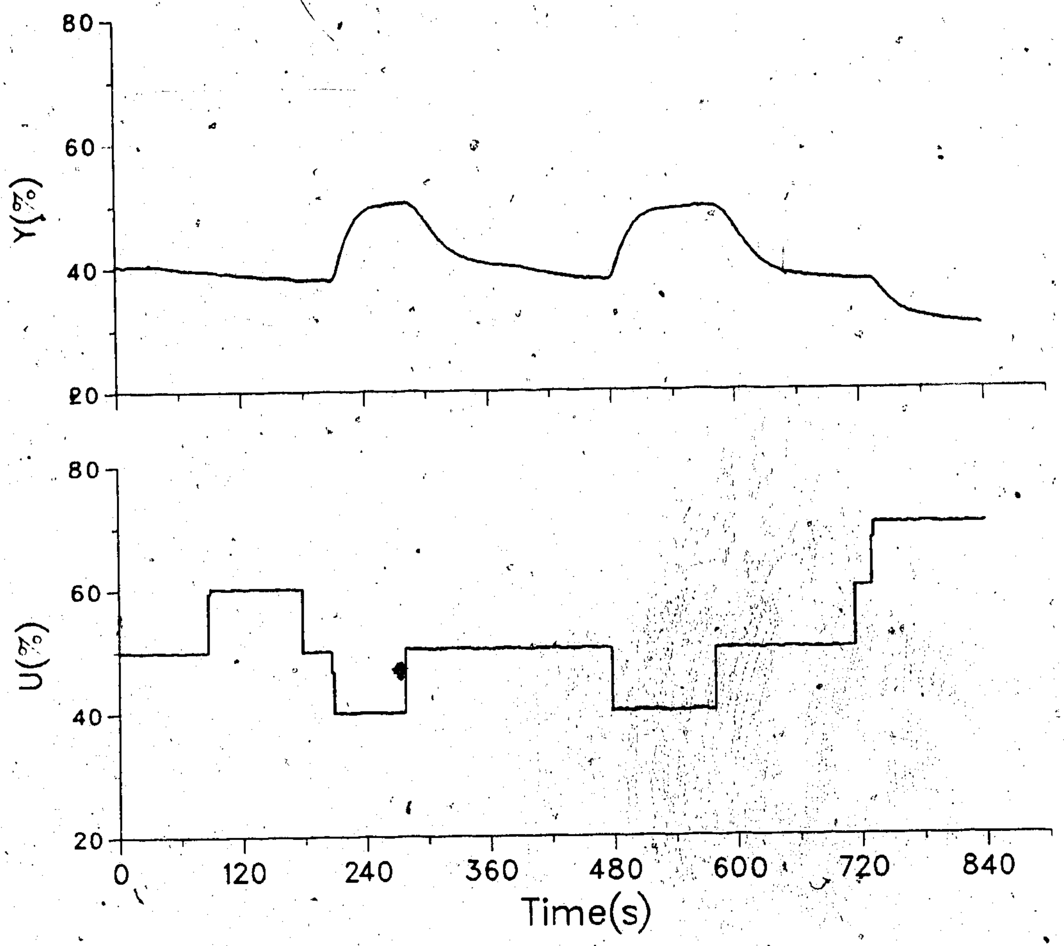


Fig. 5.4 Effect of Sticking Valve Stem on Open Loop Response

its maximum value. Note that the true air flow rate ranged between $0\text{-}20 \cdot 10^{-3}\ \frac{m^3}{s}$ corresponding to a percentage range of 0-100% during all experimental runs.

A Turnbull Control Systems model 6350 PID controller (non-adaptive) was commissioned in the inner loop. PID controller parameters were determined by trial and error. Unlike many industrial cascaded control schemes, integral action was also used to avoid load dependent

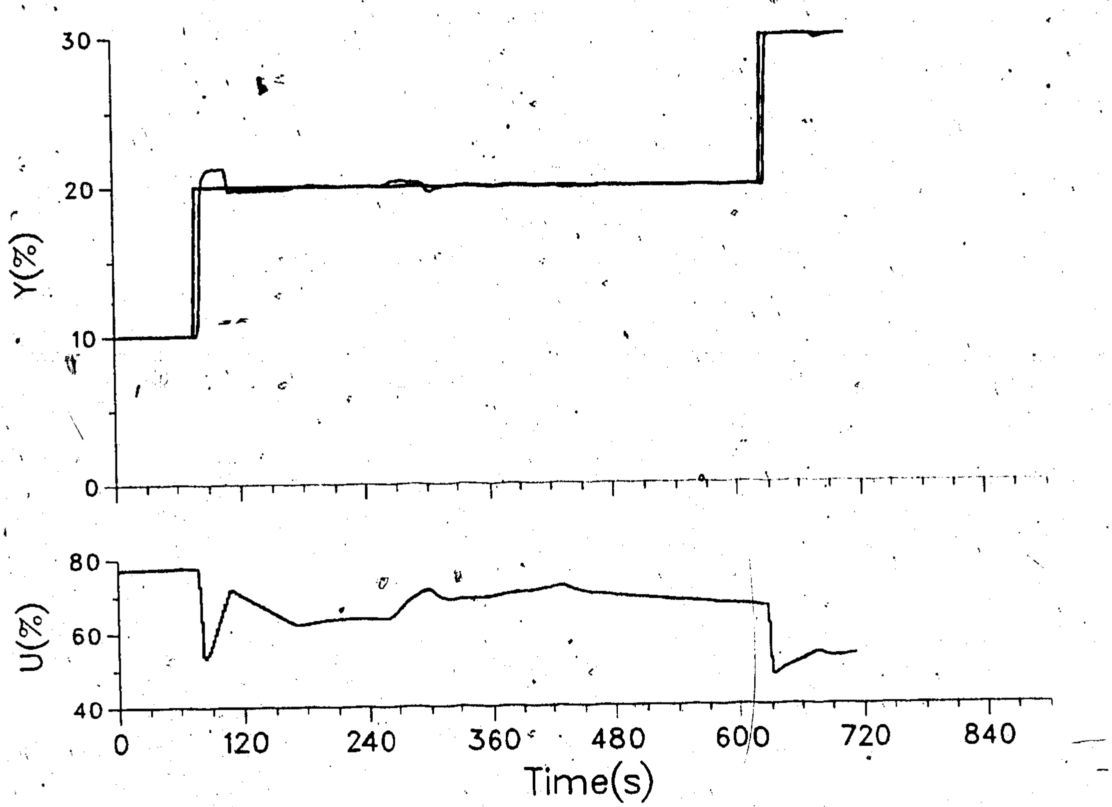


Fig. 5.5 Closed Loop Response for Inner Air Flow Rate Control Loop (SLAVE.1 - Single Set Point Change)

offsets. The supply pressure of air was maintained at 6.5 kPa gauge so that air flow rates corresponding to full span of the flow control valve stem position resulted in the desired temperature range of 20-60 degrees Celsius. The final controller parameters for air flow rate control were:

Proportional band (%)	130%
Integral time	1.15 minutes

Derivative time 0.30 minutes

Figure (5.5) clearly illustrates the effects of the sticking valve stem on the closed loop response. The interaction between the integral action and valve stem results in "drifting" control action action. The run also shows that the chosen PID values give fast response to set point changes.

Figure (5.6) shows the performance of the controller for a series of set point changes through the operating range of air flow rates. The flow control loop itself demonstrates a variable steady state gain (as expected for an equal percentage type valve) and directionally varying dynamics. The chosen PID parameters give satisfactory performance through the complete operating region.

5.1.1.3 Open Loop Characterization of Master Loop

Even with the inner flow control loop this pilot plant demonstrated several important nonideal features. Like many industrial scale processes, the steady state open loop gain varies with operating point. In this work, the temperature control loop was operated using one of two settings on the variable autotransformer: 50 or 70%, known as configurations 1 and 2, respectively. Figure (5.7) contains the open loop characterizations for both configurations. The steady state process gain is approximated as:

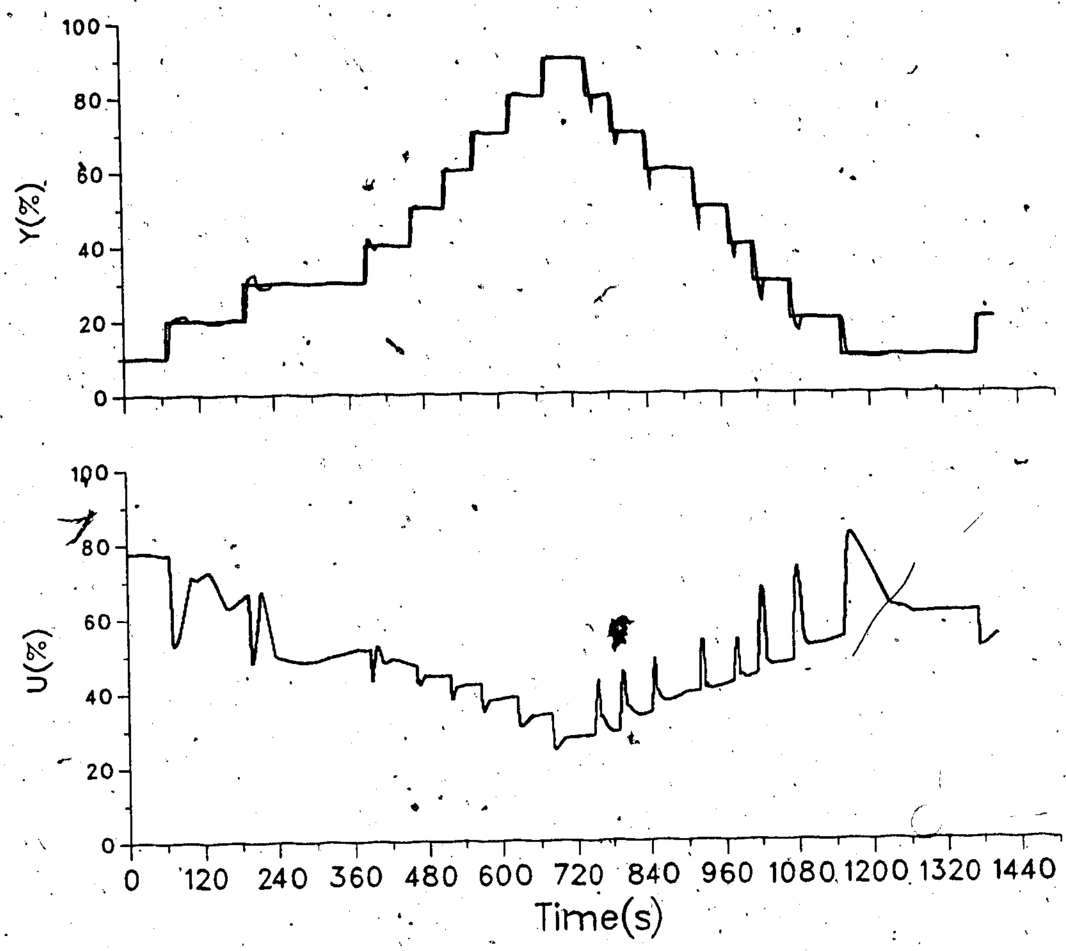


Fig. 5.6 Closed Loop Response for Inner Air Flow Rate Control Loop (SLAVE 2 - Series of Set Point Changes)

$$\operatorname{Lim}_{\Delta U_{ss} \to 0} \frac{\Delta y_{ss}}{\Delta u_{ss}}$$

For configuration #1, the steady state gain was found to vary by an order of magnitude from 3 to 0.3 at operating temperatures corresponding to 0 to 100% span of process measurement. The second twenty point characterization for configuration #2 illustrated that the process gain was also a strong function of operating point. This

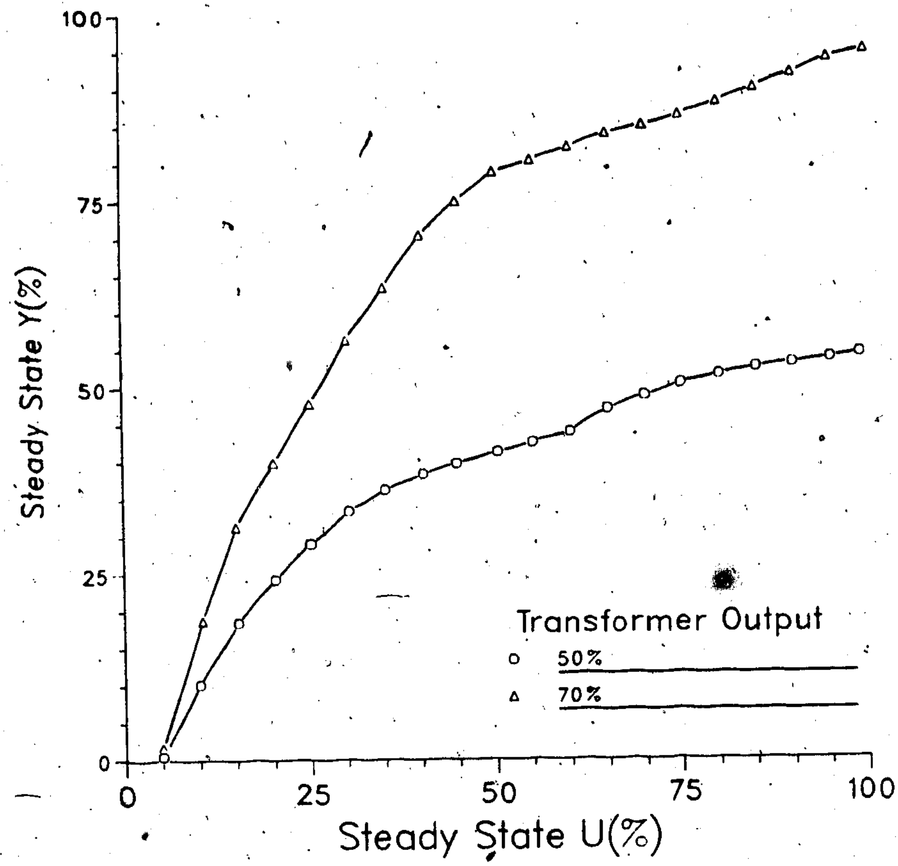


Fig. 5.7 Open Loop Steady State Characterization of Temperature Controlled Process

experimental data was also used in nonlinear process gain compensation techniques by both the Foxboro Exact and GMV controllers (see sections 6.1.3 and 8.1.3).

In both configurations, the effects of changing process gain are clearly demonstrated in Figure (5.8). The open loop dynamic response of temperature to changes in air flow rate shows that the steady state gain decreases with increasing temperature. The dynamics also indicate that the process model contains both low and

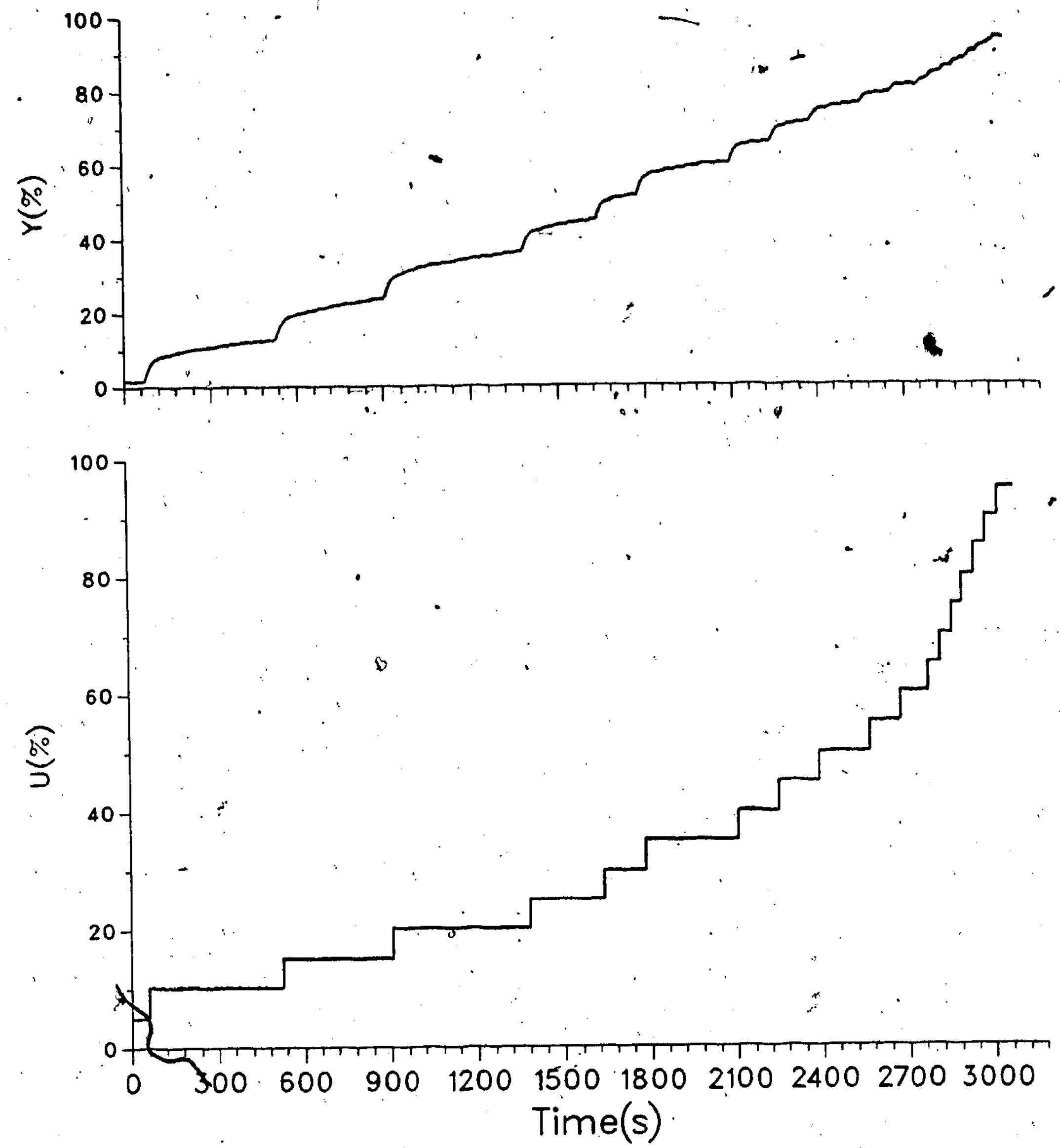


Fig. 5.8 Open Loop Dynamics of Temperature Controlled Process. (Transformer at 70%: 0_1_1.5)

higher order components. At the lower temperatures the slower high order dynamics make up the larger part of a transient response to changes in air flow rate. Note that the measured data was sampled with a two second interval.

In some runs, a random noise generator was used with this process to generate high frequency noise which was then floated about the true measurement signal. The noise was added at the Emf converter outlet.

5.1.2 First Order Process - Level Control of Water

While the use of a strongly nonlinear pilot plant such as the temperature controlled process is a demanding test for any adaptive control scheme, a more ideal plant is also useful for experimental evaluations. Therefore, level control of water in a cylindrical tank was chosen for its relatively fast and simple first order dynamics. It was felt that the adaptive controllers would likely have less difficulty giving good control performance for such a simple process. In addition, if any inadequacies in closed loop performance were observed, it would be easier to identify the cause. The schematic diagram in Figure (5.9) identifies the major components of the process. Liquid level is controlled by manipulating the inlet flow rate of water through a pneumatic control valve.

5.1.2.1 Process Model

The dynamic behavior of the level process described above can be approximated by the following first order model for small deviations in liquid level from some initial value:

$$G(s)=\frac{h(s)}{u(s)}=\frac{K}{\tau_p s+1}$$

For this work experimental evaluations were performed

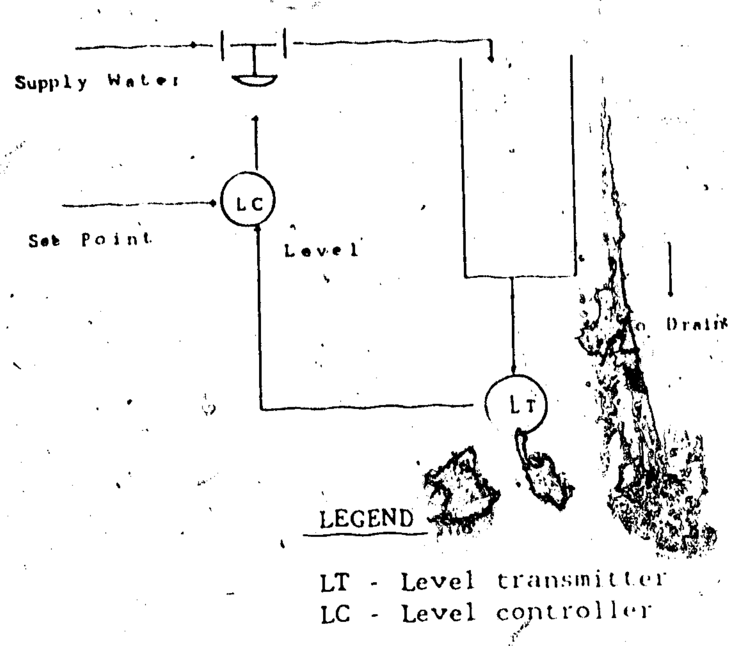


Fig. 5.9 Schematic of Liquid Level Controlled Process

for a liquid level range of 0.10 to 0.90 m (10.00% to 90.00%). A discrete approximation of the above transfer function (assuming a zero order hold sampling device) is given by:

$$G(z^{-1}) = \frac{h(z^{-1})}{u(z^{-1})} = \frac{b_0 z^{-1}}{1-a_1 z^{-1}}$$

where

$b_0 = \frac{1}{A}$, A=cross-sectional area of tank

$$a_1 = e^{-(\frac{R}{A})T_s},\ T_s = \text{discrete sampling time.}$$

For an operating range of 25 to 30% level (T_s=2 seconds), G(z) is :

$$G(z^{-1}) \simeq \frac{.2041z^{-1}}{1-.9592z^{-1}}$$

5.1.2.2 Open Loop Characterization

Several open loop response runs were performed to illustrate the open loop dynamics of the level control system. For the operating range of interest, Figure (5.10) below illustrates that the process gain was not a strong function of level. Based on a linear slope approximation the steady state gain was 5.0.

Note that the negative slope is due to the reverse acting flow control valve. The open loop response to an increase in water flow rate shown in Figure (5.11) indicates a first order process with a process time constant of 0.80 minutes. Figure (5.12) demonstrates that the level control process does not exhibit directionally varying dynamics. This is a useful property since the process model parameters should be constant over a narrow operating region to permit the adaptive controllers to converge. For the steady state characterization shown previously the corresponding volumetric outlet flow rates were measured and shown to vary linearly with tank level which indicates that the

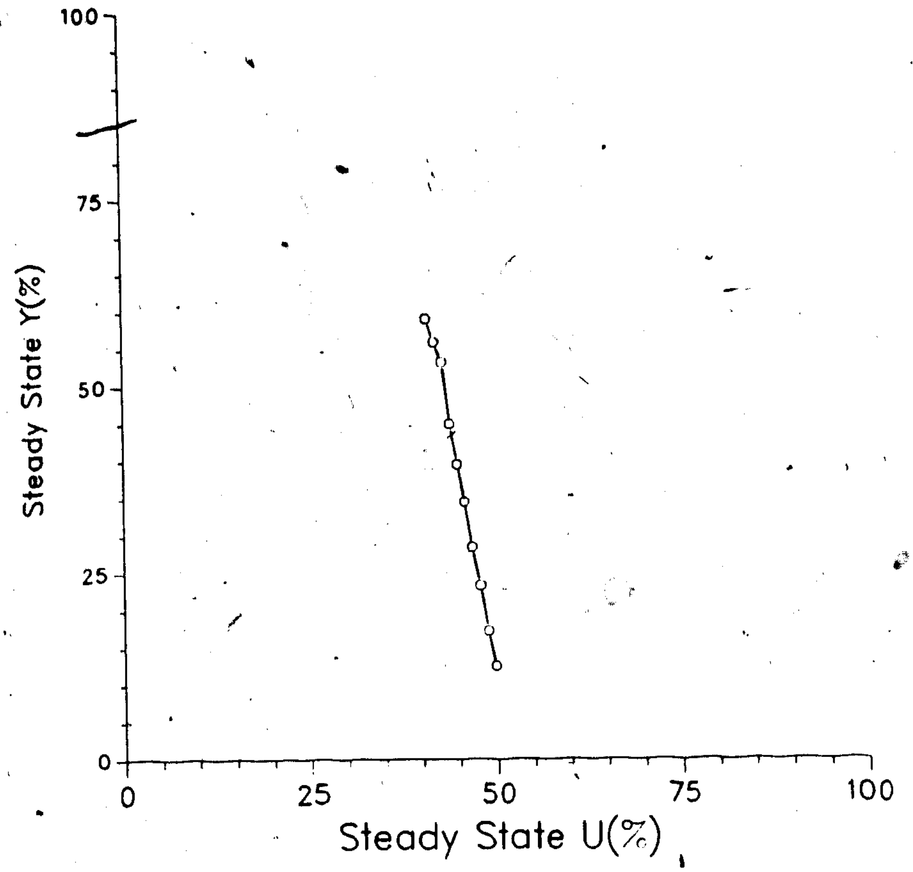


Fig. 5.10 Open Loop Steady State Characterization of Water Level Controlled Process

water tank process is linear over the operating region of interest.

5.2 Description of Computer System

A personal computer (PC) system was used to facilitate experimental evaluation of the two adaptive PID controllers and the implementation of the adaptive predictive controller described in Chapters three and four. In order to document the performance of the adaptive controllers and monitor the

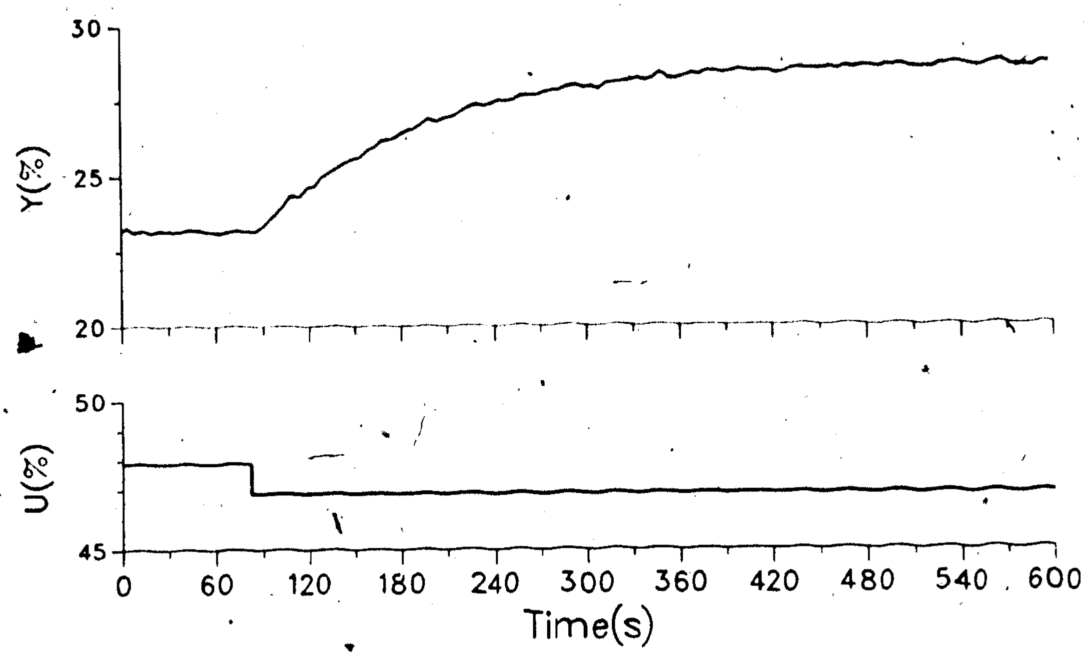


Fig. 5.11 Open Loop Response of First Order Process (0_1_1.2)

state of their parameter estimation schemes, a large amount of information was required each time the controllers were polled. While both the Foxboro and Turnbull controllers have hand held configuration terminals for retrieving and displaying parameters (as well as implementing changes) they are slow and tedious to use. Therefore, software was written by the author to perform discrete sampling of information from the controllers' databases during the experimental evaluations. These supervisory programs also allowed the operator to quickly configure the controllers to perform specific tasks, thereby decreasing the time required to carry out experimental runs.

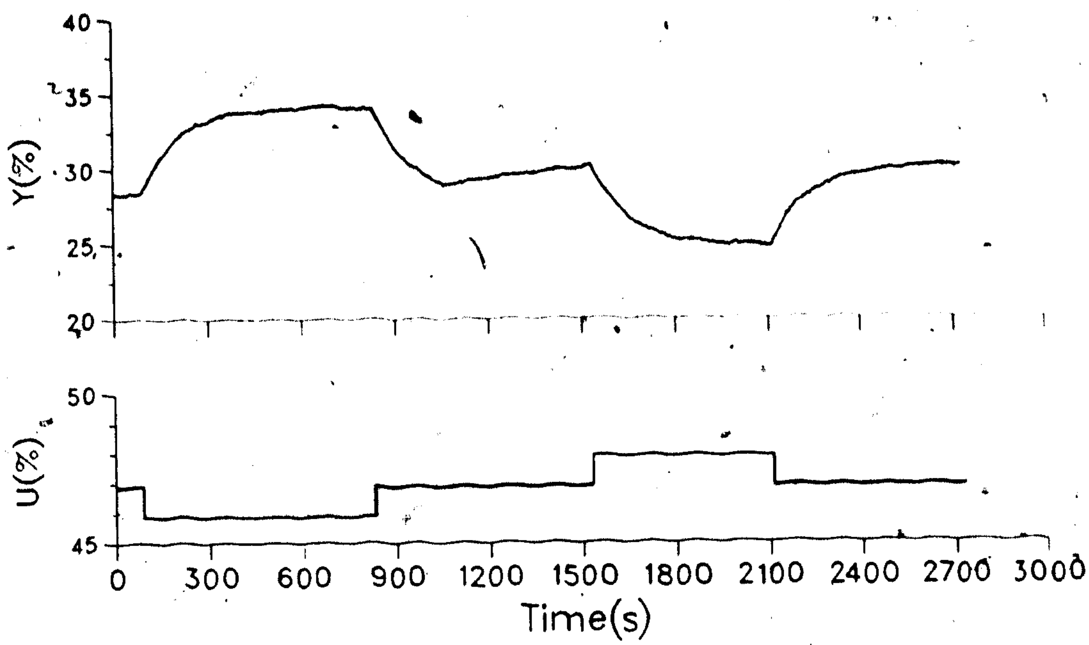


Fig. 5.12 Open Loop Response of First Order Process (0 1 2.2)

The complete system is shown in Figure (5.13). The personal computer system was used for actual supervision and control. Recorded data was stored in hard disk files and later transferred to the University of Alberta's mainframe computer, an MTS based system, for data work-up purposes.

The personal computer system itself consisted of an IBM-XT personal computer with:

- 640kB random access memory (RAM).
- 20MB winchester drive.
- 8087 math coprocessor.
- monochrome monitor (640 x 320 pixels).

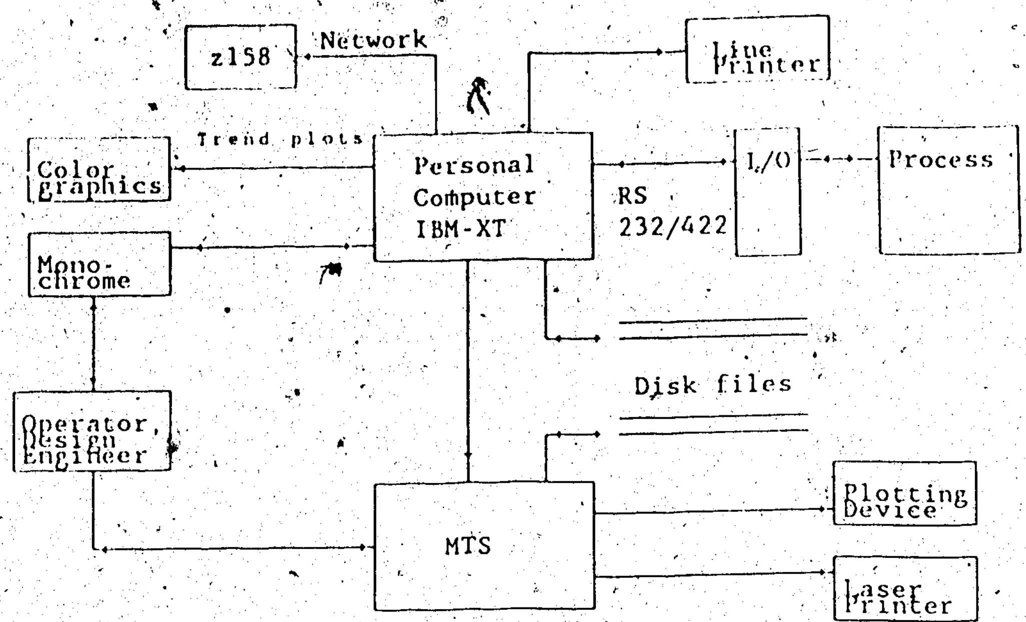


Fig. 5.13 Overview of Computer System

- medium resolution RGB colour monitor.
- single floppy drive.
- EPSON © FX-100 ™ printer.
- asynchronous communications adapter card.
- real-time clock card with battery backup.

An additional Zenith Data Systems Z158 personal computer was available for networking purposes in the implementation of the academic control scheme.

The supervisory program for the Foxboro EXACT controller was implemented with IBM's DOS3.1 version of advanced BASIC and is based on a demonstration program provided by the

Foxboro Company. Interpreted Basic is a relatively slow langauge, but the limitation on the amount of data retrieved per sampling interval was not the execution speed of the supervisory program but the lack of a good communications protocol. The RS-232 connection for the EXACT controller was designed for the hand held configurator and relatively slow data transmission rates (300 baud with no error checking).

Turnbull Control System's line of products support a rigorous communications protocol that operates at transmission rates up to 9600 baud allowing for the transmission of a great deal of information. In order to take advantage of these communication rates, a compiled langauge was necessary. In addition, it was found that the single task IBM-DOS operating system was not practical for real-time applications. Parameter changes must be prespecified or at least implemented within the duration of a sampling period so that the integrity of the sampling interval is preserved. For most operating systems, addressable memory can not exceed 64kB. This is a serious limitation for any practical control scheme which generally requires a considerable amount of terminal I/O for configuration and operator interface purposes. The next section contains a brief discussion of a real-time multi-tasking operating system that does not suffer from these limitations. Both the supervisory system for the TCS "Auto-Tuning Controller" and the Academic Control System were implemented using a full featured compiled BASIC langauge under the QNX operating system.

5.2.1 QNX™ Operating System

QNX, developed and marketed by Quantum Software Systems Ltd. is a multi-user, multi-tasking operating system for personal computers. QNX allows for the concurrent execution of up to 40 tasks through sharing of the CPU and an interrupt driven protocol. The operating system also supports a distributed network. This permits the execution of CPU intensive tasks on dedicated peripheral processors and the transmission of data via high speed communications cables. Inter-task communication is supported using messages, exceptions and ports.

Due to addressing limitations, QNX does not support code larger than 64kB. This limitation is not a major problem since large tasks can be subdivided and then scheduled from within a user written task. Refer to the QNX reference manuals for a more detailed description of the above features.

5.2.2 QNX Programming Langauge - BASIC

At the present time Quantum markets a "C" compiler and a BASIC compiler. For this work, QNX BASIC was the chosen langauge. QNX BASIC is a sophisticated high level compiled langauge offering many important enhancements over the typical BASIC langauges presently found.

The most important, advanced features are:

- separately compiled modules.
- local variables for recursive functions.
- global variables.
- multi-line IF-THEN-ELSE constructs.
- explicit and implicit variable type declarations.

It should be noted that all code written in QNX BASIC can be interfaced with code written in "C", subject to module interface definitions.

5.3 Applications Software

For the design and implementation of a practical control system, a real-time multi-tasking operating system is required. A practical controller must be able to perform the following operations concurrently:

- do closed loop control.
- display important operating parameters both numerically and graphically.
- accept parameter changes via keyboard input.
- data acquisition.
- file I/O.
- handle operator interrupts.

and at fast enough speeds so that "practical" sampling rates can be realized. Such a control system must be easy to startup, allowing an inexperienced user to quickly initialize parameters and configure the system with a minimum amount of knowledge and procedural effort. Default values should be provided along with options designed to provide even better initial startup data. The system must be easy to operate:

- using a hierarchical structure of clearly defined menus and options.
- providing easily understood instructions and guidelines.
- with limits on user adjustable parameters for protection.
- utilizing full featured terminal I/O capabilities.
- and process monitoring information must be functionally organized and easily retrieved.

If a controller is designed with consideration given to these types of requirements it will be much more likely to find acceptance for commercial applications.

The use of a full featured operating system such as QNX allows for the design and implementation of a control system composed of concurrent tasks running at different priority levels and using inter-task communications. Such a system is considerably more complex than a less sophisticated controller based on a single task design, and thus requires a great deal more effort to implement. However, with a good design, it is usually easy to use and well suited for real control applications.

5.3.1 Academic Control System (A^3)

This section provides the reader with some insight into the high level design of the academic control system. The multi-tasked design is discussed in terms of how it allows for the simultaneous performance of control system functions and its advantages over single task controllers. Special design features are reviewed such as the use of task to task communications, use of graphics, networking capabilities, and priority specified task creation.

The A^3 system consists of three main tasks that run in parallel during normal operation. Figure (5.14) illustrates the control system's design at the highest level.

The three main tasks and their purposes are:

A. MI-SO-EXEC - Multi-Input Single Output Executive program.

- creates (starts) other two tasks using library commands.
- verifies that all tasks are running correctly.

During normal operation:

1. Task suspends itself for the duration of the

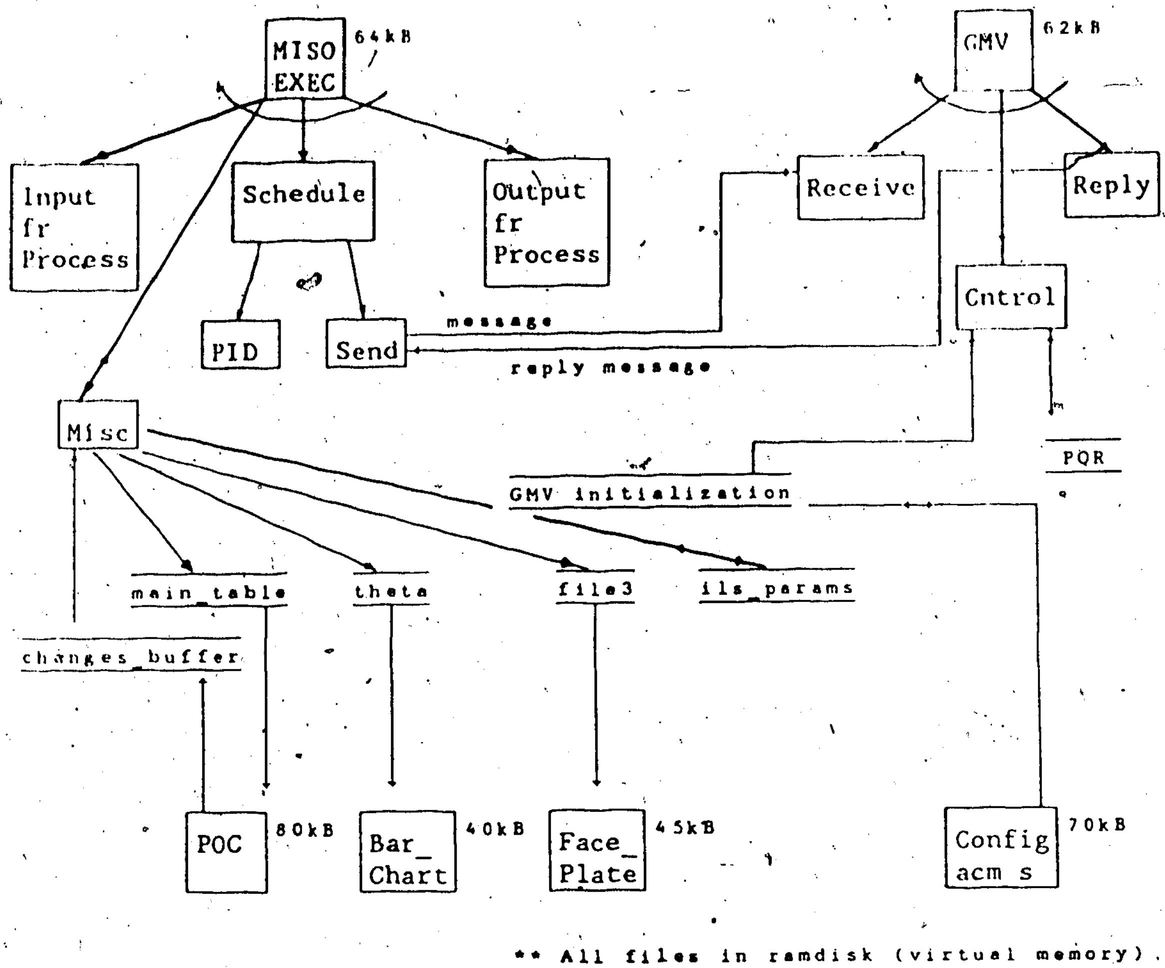


Fig. 5.14 High Level Structure Diagram for A^3 Control System

previous sampling interval by "sleeping". Task does not use the CPU while suspended thus freeing it for use by other tasks.

2. Task checks for the existence of a "changes buffer" file stored in virtual memory that contains operator specified parameter changes sent via the Process Operator's Communications (POC) interface task.

If the file exists, the changes are processed.

3. Miso_exec then performs a series of functions depending on the status of parameters which are specified by the operator via POC. Possible functions include:

- data acquisition.
- digital filtering of measurements.
- determination of controller output.
    - manual mode.
    - automatic mode.
        - fixed gain PID.
        - adaptive GMV.
- writing of information to virtual memory files for retrieval by the POC task.
- logging of data for documentation purposes.
- sending of calculated controller output to process.

B. GMV - <u>S</u>elf-<u>T</u>uning <u>C</u>ontroller task.

This program contains the self-tuning controller described in Chapters three and four. The program's operation is completely determined by parameters given to it within a message buffer sent by Miso_exec. Using task to task communication primitives, Miso_exec sends a transfer buffer containing up to fifty floating point parameters to

GMV every control interval. The first ten elements are "fixed" value and consist of:

yin	-	current filtered process measurement(%).
w	-	current set point (%).
v	-	filtered measured disturbance.
ts	-	sampling interval (s).
uold	-	previous controller output (%).
acm_flag		advanced control module flag that determines what functions are performed by the GMV.
init_flag		initialization flag used to force the GMV to perform varying degrees of reinitialization.
xkappa	-	upper bound on C[P(k)] in ILS.
xk	-	scalar multiplier for P(k).
iota	-	lower bound on $\|P(k-1)x(k)\|$ (used in ILS).

which are required inputs. Timing problems associated with data acquisition (performed by Miso_exec) and control (performed by GMV task) are solved using a "blocking receive" type of message transfer. The GMV task suspends execution until Miso_exec <u>sends</u> the message containing the transfer vector. While waiting, the GMV is left in a <u>RECEIVE-BLOCKED</u> state. When Miso_exec <u>sends</u> the message, it then becomes <u>REPLY-BLOCKED</u>, awaiting a reply message from the GMV task which is now busy calculating a control signal based on the control equation (3.18). Once the GMV task has calculated a controller output it updates the information in the transfer buffer and replies to Miso_exec. Miso_exec is therefore no longer REPLY-BLOCKED and continues execution while the GMV task returns to the top of its program and RECEIVE-BLOCKS until the next sampling interval.

The two tasks with blocking message transfer is almost equivalent to the more conventional superordinate subordinate arrangement for a single task where the GMV would have been a subroutine called by Miso_exec. The single task design is somewhat easier to implement but the two task arrangement offers important advantages.

1. If the GMV code is large, linking it with Miso_exec could exceed code size limitations of 64kB (this was actually the case).
2. If desired, the GMV task could be run on another processor in the network thereby freeing up the CPU on the main PC. Running the GMV on another peripheral would also free RAM for other tasks. Since the GMV task performs no terminal I/O and a minimal amount of file I/O (initialization only), it does not matter where the task runs in the network.
3. Additional advanced control tasks could be developed by others and used with only minor modification of either Miso_exec or POC task. They would only have to use the same transfer procedure and structure for the message

buffer.

C. POC - Process Operator's Communications (POC) Interface task.

The POC task is the operator's interface to the A^3 control system. Operator specified parameter changes are sent to Miso_exec via the changes buffer which is processed every sampling interval. As well, current operating conditions and other data are retrieved from ramdisk files which are updated by Miso_exec. The POC task, created by Miso_exec runs at a lower priority than either Miso_exec or GMV so that these more critical operations are never held up. It is emphasized that unlike a single task controller this system does not require that control be stopped during POC execution and/or that operator specified inputs be quickly implemented so that control sampling intervals are not corrupted or delayed. The data acquisition and control steps are completely uninterrupted by terminal I/O. This allows for shorter control intervals and thus increases the suitable range of applications for the controller as well.

The POC task uses a well organized hierarchical system of menus that allow the operator to easily configure the system for a particular mode of operation. The high memory overhead associated with terminal I/O coupled with the 64kB code size limitation for single tasks has resulted in the subdivision of POC into smaller tasks. When particular menu items are selected from within the main POC task, POC simply starts the "sub-tasks" redirecting their terminal I/O to its screen and suspends itself until the "sub-task" completes execution.

The interface task uses terminal I/O features, such as the following, to improve readability and allow for quick assimilation of information:

- highlighting of displayed parameter values.
- masking of input fields for operator specified parameter changes.
- the use of graphic displays.

Some of the more important "sub-tasks" associated with POC are functionally described below.

1. Configuration of Initialization Files - "CONFIG_ACM"

This task is used to retrieve and modify any of the initialization files used by the GMV task. Information to be specified includes:

a) Model structure.

b) Discrete transfer functions $P(z^{-1})$, $Q(z^{-1})$, $R(z^{-1})$..

c) Initial parameters for least squares estimation.

d) Steady state process I/O data for nonlinear compensation.

e) Miscellaneous data. e.g. digital filter constants.

Options are selected using menus. The task searches for the corresponding data files and retrieves current data values which are then displayed. If desired, the operator may implement parameter changes and when finished, redirect updated data files to a ramdisk. If for a given option, the data file does not exist, the task will display default values. This task can also be executed off-line for database modification if desired.

2. Faceplate Task

This sub-task provides the operator with a graphical display of current operating conditions. Process set point, measurement and current controller output are displayed using vertical bargraphs ranged 0-99.99%. Numerical values are also displayed to four significant figures. The data is retrieved from a ramdisk file that is continually updated by Miso_exec. The bar charts are continually updated from within a non-branching goto loop. The sub-task uses a keyboard interrupt with an exception handler routine to allow for branching out of the loop to a menu. The menu allows the operator to change either the process set point or controller output depending on the controller's mode of operation. If desired, the operator can also change the sampling interval used by Miso_exec.

3. Trend Task

This graphics program produces trend plots of specified data using either medium or high resolution (black and white) modes. The task runs on-line accessing a ramdisk file updated by Miso_exec. Trends are updated at a user specified interval rate and hardcopies obtained using "screen dumps". This task can be useful for the plotting of parameter estimates $\theta(k)$ with time in order to observe variations in $\theta(k)$ when either RLS or ILS are used for parameter identification.

The program allows for the use specified limits on "x-y" data, tick marks, labels and physical location on the RGB monitor.

5.3.2 Applications Software and Documentation

To indicate the amount of programming effort involved in this work, a brief summary of the supervisory systems designed for experimental evaluation of the controllers is presented. The total programming effort resulted in approximately 16000 lines of code written by the author.

<u>Foxboro Supervisory Program</u> "Fox.bas"

IBM DOS3.1 - Basic language implementation - single, menu driven task.

Size - ≃2000 lines of source code.

Supervisory System for TCS 6355 Controller

QNX Operating System - QNX Basic implementation, multi-tasked (two) design.

1. Data acquisition task, DATA_ACQ - 2600 lines code.
2. Process operator interface task, TCS_POC - 1500 lines.

   - the POC task has four subordinate tasks which, in total account for 1200 lines of code. These programs do not run concurrently to TCS_POC.

A[3] Control System

QNX Operating System - QNX Basic implementation, multi-tasked (three) design.

1. Data acquisition task, MISO_EXEC - 2800 lines code.
2. Process operator interface task, POC - 2700 lines code.
3. GMV control task, STC - 2900 lines code.
4. Initialization task, CONFIG_ACM - 1300 lines code.
5. Faceplate task, BAR_CHART - 500 lines code.
6. Trend task(graphics plotting), TREND - 500 line code.

Additional Documentation

1. Functional Description of Foxboro's "Exact" Controller.
2. Functional Description of TCS's "Auto-Tuning" controller.
3. User's Manuals:
   a) Foxboro's supervisory program - for the purpose of demonstrating a self-tuning controller's performance, this interface program may be used within undergraduate control laboratory courses. The manaual provided the user with a description of the features available, the programs capabilities, and system requirements.
   b) TCS supervisory system - as above, this system of programs can be used within a student laboratory environment to demonstrate self-tuning PID control. The manual provides the reader with a description of the system, and the features currently available.
   c) A^3 control system - this report presents a detailed functional description of the controller's capabilities. Details regarding necessary configuration of the QNX environment are given, along with hardware requirments.
4. Programmer's/System Documentation for A^3 System.
   For each of the QNX based systems, current source code listings are provided with: module interface definitions, program structure diagrams and any special details regarding compilation, linking and

execution of the individual tasks.

6. Foxboro's Exact Controller - Experimental Evaluation

This Chapter provides the results of experimental evaluations for the Exact controller. A functional description along with typical performance runs outline the controller's general capabilities. A series of evaluations are then presented that demonstrate the Exact's performance for specific process conditions such as nonlinearities, measurement noise, asymmetric dynamics and others. Features that have been incorporated into the Exact's design to improve performance under these conditions are then evaluated.

6.1 Functional Description

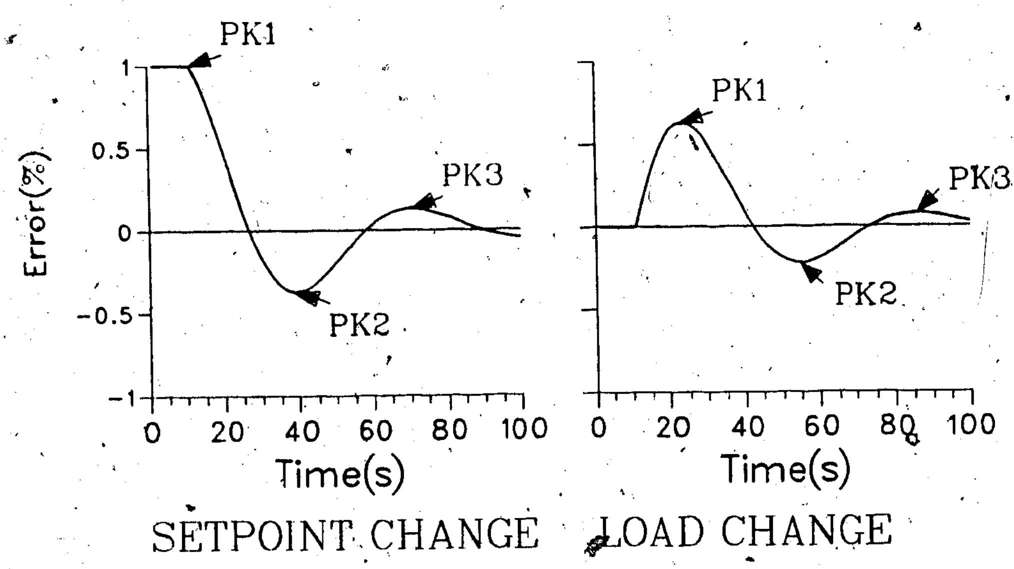
The EXACT controller is a microprocessor based self-tuning PID controller designed for single loop control. The tuning algorithm monitors the closed loop response to a set point or load disturbance and updates the PID parameters so that user specified values of overshoot and damping are achieved.

The EXACT controller uses two separate sets of PID parameters. During non-adaptive operation when the self-tuner is turned OFF, the fixed parameters: PF (band%), IF(minutes) and DF(minutes) are used. When the self-tuner is turned ON these fixed values are transferred to the adapted set: P, I and D which are then adjusted by the tuner. The tuner does not adjust the fixed set which can therefore be used as backup values. If desired the operator can explicitly transfer the P,I,D values to PF,IF,DF.

6.1.1 Expert Adaptive Controller Tuning

The EXACT controller's self-tuning algorithm continually monitors the closed loop system's set point error. Dormant at steady state, the self-tuner becomes active whenever the set point error exceeds twice the noise band, or specified level of measurement noise. For the above disturbance, load or set point, the tuner measures peak heights of error response, time between peaks and steady state error. Peak height information is converted into dimensionless performance variables: overshoot and damping.

Typical error responses for set point and load disturbances are shown in Figure 6.1 with labelled peaks.



$$\text{OVERSHOOT} = \frac{-PK1}{PK2} \qquad \text{DAMPING} = \frac{PK3 - PK2}{PK1 - PK2}$$

Fig. 6.1 Error Response Curves for Set Point and Load Disturbances

Note that the damping ratio term is <u>not</u> the same as the classical damping ratio definition. If a closed loop response is overdamped, pseudo values are assigned for the heights of peaks 2 and 3 given by PK2 and PK3. From the measured information above, the I and D parameters are directly set using the measured period (time between PK1 and PK3). The I and D controller parameters are adjusted to give desired ratios of I/period and D/period which are internally set by the self-tuning algorithm. If the closed loop response is overdamped the I and D parameters are not adjusted using period information.

If the observed overshoot and damping are both less than the user specified maximums, the controller gain is increased. Since overshoot and damping are not independent the gain is increased according to the smallest error in either parameter. Adjusted PID parameters are automatically updated and used in the controller.

The self-tuner requires some estimate of the time scale of the process so that it knows how long to wait for peak heights. Figure 6.2 illustrates the various states of the self-tuner during a transient response to set point change. According to Foxboro, the self-tuner uses knowledge based rules that have been selected to ensure general applicability. For additional insight into the EXACT controller's tuning the reader is referred to papers by Kraus (1984) and Bristol (1983).

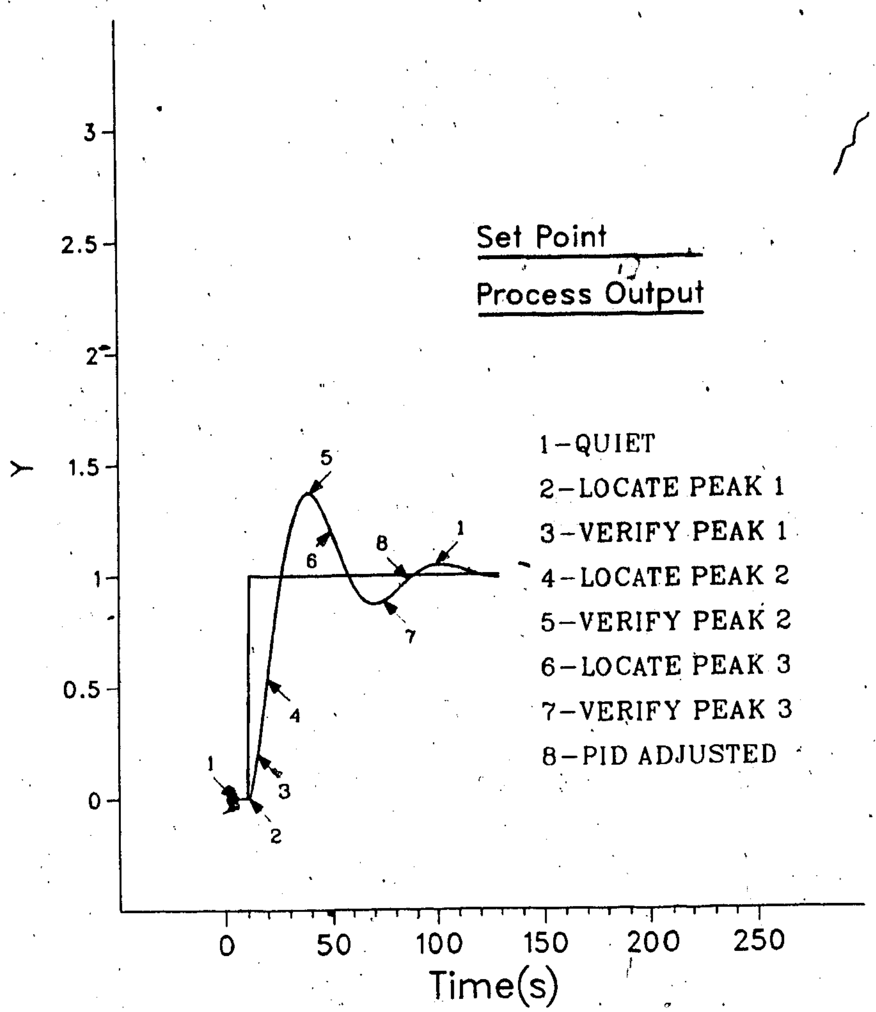


Fig. 6.2 States of Self-Tuner During a Set Point Transient

6.1.2 Required Inputs

One of Foxboro's objectives was to design a self-tuning controller that was easy to use and required a minimum amount of expertise or knowledge to operate. For self-tuning operation the following information must be specified:

PF,IF,DF Initial estimates of desired PID parameters.

NB: Noise band

The user must specify limits (%) for the noise in the measured process output signal which determines when the self-tuner will enter the adaptation cycle. If the specified noise band is too high, the self-tuner will ignore useful dynamic information. If the limits are too low, adaptation may be started due to noise rather than because of disturbances as shown in Figure 6.2.

WMAX: Maximum wait time

The self-tuner requires an estimate of the time scale of the closed loop system. WMAX is the maximum time that the algorithm will wait for the second peak of an error response to a disturbance or set point change. Where T=period of oscillation, WMAX is bounded by:

$$\frac{T}{2} < \text{WMAX} < 8 \cdot T$$

In normal operation, if WMAX is set too small the closed loop system will appear overdamped to the self-tuner. If the process requires 30 minutes to complete a cycle of oscillation and WMAX is set to 1 minute, the self-tuner will decide that the response is overdamped, and tighten the control card settings. This can lead to closed loop instabilities.

If WMAX is set too large the self-tuner will decide that the closed loop response is much faster than the response expected based on WMAX. The self-tuner will not adjust PID parameters and WMAX will have to be decreased to enable the self-tuner to operate.

If the above required inputs are correctly given, the controller will be able to operate in the self-tuning mode. When the above information is not available, a special open loop procedure can be used to obtain initial estimates for PF,IF,DF,WMAX and NB. This PRETUNE feature is described in section 6.1.4.

6.1.3 Additional Features and Inputs

Foxboro has provided additional features and optional inputs that improve overall performance and widen the controller's range of applications. These options are not critical for the operation of the self-tuner and default values are provided that generally give good results.

<u>Optional Inputs</u>

Maximum allowed overshoot and damping limits: (*OVR and DMP*).

These two measures of performance are not independent so the self-tuner adjusts PID parameters based on the limit that is closest to being exceeded. Since large overshoot usually requires higher controller gains, an operator should exercise caution when specifying overshoot for overdamped processes. Default values of 0.5 and 0.3 for overshoot and damping respectively are provided by Foxboro.

Derivative factor (*DFCT*).

The controller's derivative action can be modified by using the derivative factor. The EXACT controller multiplies the calculated derivative action by the derivative factor to either increase or decrease its contribution to u(k). Normal derivative action is specified by setting DFCT=1 and disabled using DFCT=0. The initial identification procedure PRETUNE provides an initial value for the derivative factor. If the self-tuner is OFF, the operator must explicitly set the derivative constant, DF, to zero to disable derivative action.

Clamping of PID parameters using Change Limit (*CLM*).

The operator can limit the values of controller parameters within a range expressed as a fraction and multiple of PF and IF (the fixed PI constants) as:

$$\frac{PF}{CLM} < P < CLM \cdot PF$$

The CLM parameter has a range of 1.25 to 100.0 and a default value of 4.0. The use of the change limit can prevent the self-tuner from producing unreasonable controller parameters due to unforeseen factors.

Limits on cycling of controller output (*LIM*).

The self-tuning algorithm monitors the controller output when it changes at a frequency higher than that which the plant can respond to. If the peak to peak magnitude of those oscillations exceeds the specified value of LIM for 3 minutes or more, the controller is automatically detuned by increasing the proportional band P and decreasing D. LIM has a range of 2-80% and a default value of 80%.

Digital filtering of measurement (*LAG*).

Measurement can be filtered with a second order Butterworth filter given by:

$$G(s)=[\frac{(\gamma s)^2}{2}+\gamma s+1]^{-1}$$

where

$$G(s)=\frac{Y(s)}{X(s)}$$

and

Y=filter output

X=filter input (raw measurement).

γ=time constant.

s= Laplace Transform variable but for linear O.D.E.'s with zero initial conditions s→d/dt.

The actual filter used is a finite difference approximation of the transfer function above with user specified LAG replacing γ. The reader should note that this is an underdamped filter rather than the widely used exponential filter.

Nonlinear process gain compensation (*NLN*).

The EXACT controller has an option that allows its user to make use of known steady state I/O data. An "n" (up to twenty) point characterization entered into a nonlinear table by the user allows the self-tuner to adjust its PID parameters in response to the difference between an expected nonlinear error response curve and the true curve. This effectively results in less variation in adjustable controller parameters over the nonlinear operating range.

6.1.4 Pretune - Open Loop Identification Option

For startup procedures when initial estimates of required parameters are not available, Foxboro has provided the PRETUNE option. Operator initiated when the process is at steady state, and in manual mode, the process is perturbed by a user specified (±BUMP%) step change in controller output. From the resulting open loop response, the self-tuner estimates the PID parameters, noise band, maximum wait time and derivative factor.

The user specified BUMP must be large enough so that the process measurement will change enough to give a reasonable signal to noise ratio, eg. 2.5%. The four main stages of the PRETUNE are:

1. Process is perturbed by a step change in controller output of ±BUMP%. If the BUMP is too small, the PRETUNE status parameter, PTUN, will display "PTUN=SMALL 1". The

operator will have to restart the PRETUNE with a larger BUMP.

2. Process reacts to the input and moves towards a new steady state.
3. Once steady state is reached the PID parameters are calculated and the controller output is returned to its original value. If the process has a high gain or is an integrating type, the controller output is returned when the measurement changes by 10% of its span or the BUMP size, whichever is larger.
4. During this stage the controller waits for the process to again reach steady state at which time the noise band and derivative factor are estimated. If the process measurement is "noisy", the derivative factor is reduced to minimize the effect of noise on the derivative action.

These four stages are illustrated below for the temperature controlled pilot plant in a single loop scheme. Controller output (0-100%) corresponds to air flow rate ($20\text{-}0 \cdot 10^{-3}\ \frac{m^3}{s}$) and process measurement (0-100%) corresponds to 20-60°C.

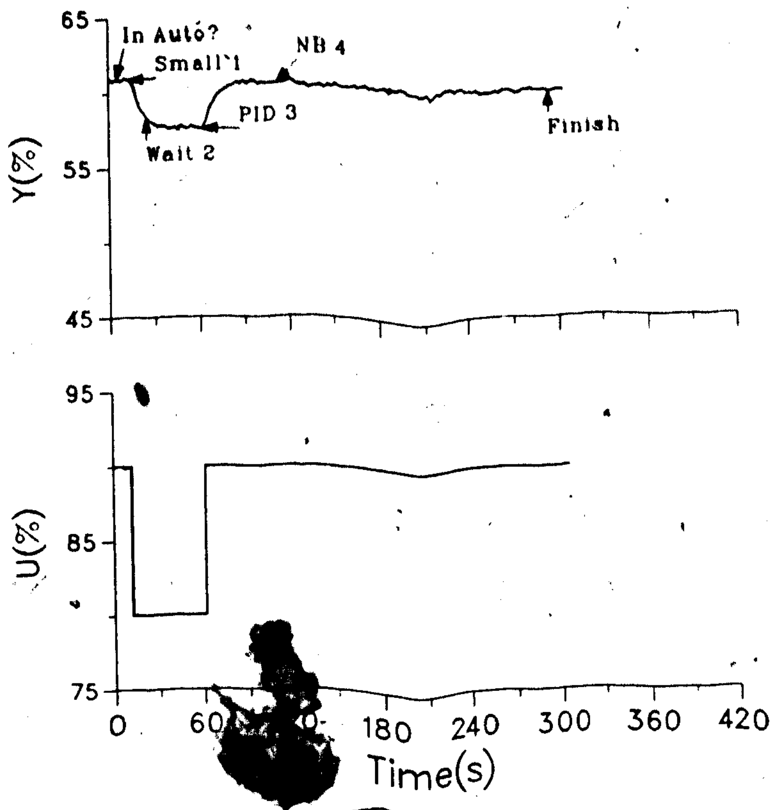


Fig. 6.3 Pretune States - Using Temperature Controlled Pilot Plant

6.2 Open Loop Identification - Pretune Performance

When the Pretune option is used to obtain initial estimates of required input parameters, the operator has only to specify the direction and magnitude of the step change in controller output. Using the resulting open loop response the EXACT controller gives good initial PID constants that result in conservative, damped, closed loop performance. The procedure used to calculate initial parameters is not specified but is independent of user specified limits on overshoot and damping.

Figure 6.4 illustrates two Pretune sequences for the temperature controlled pilot plant described in Chapter five. In these and all subsequent experimental runs, the cascaded control scheme was used so that the EXACT controller's output was the set point to the inner air flow rate controlled loop. For these two runs, the autotransformer was operated at 50% output corresponding to configuration #1 (see section 5.1). Both runs used BUMPs of 10% magnitude as shown but opposite sign.

Comparison of the two open loop responses clearly indicates the nonlinearity of the plant. At the lower operating region the plant is much more sensitive to changes in air flow rate with a steady state gain that is twice that at the higher temperature. The Pretune results indicate the degree of nonlinearity as well. The recommended controller gain in the 30-40% operating region was 2.9 while in the 55-60% region (lower process gain) it was 7.1. The process time scale remains relatively constant as indicated by the maximum wait times specified. The process measurement is relatively noise free as shown by the controller's estimate of the noise band and derivative factor.

For the control of nonlinear processes such as this the widely varying open loop responses should indicate that:

- operator should be wary of large set point changes particularly from low to high process gain regions.
- the use of the nonlinear compensation option should be considered particularly for processes subject to

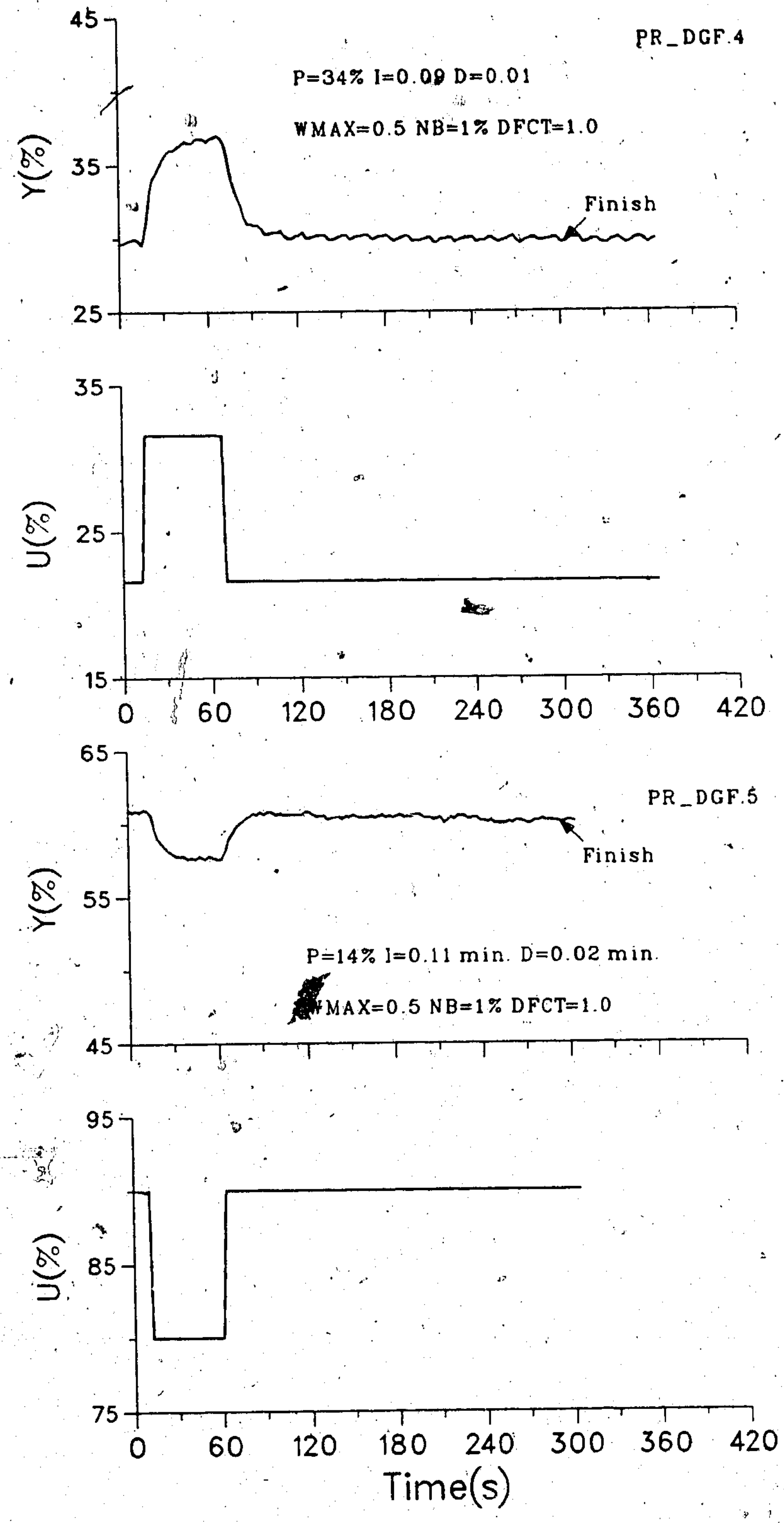


Fig. 6.4 Use of Pretune to Obtain Initial Estimates for Required Inputs

large load disturbances or changes in desired set point.

For a given operating point the EXACT controller's Pretune gave consistent results. Figure 6.5 demonstrates a typical response to up/down set point changes immediately following a Pretune.

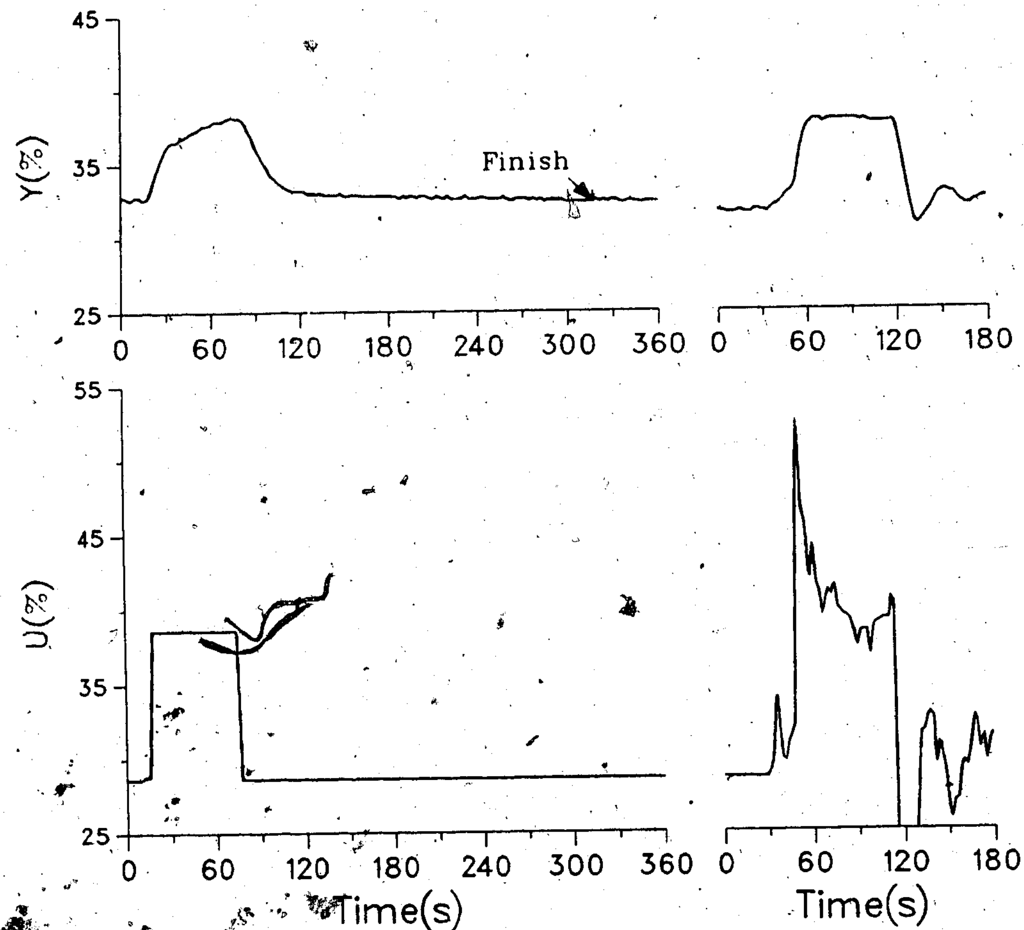


Fig. 6.5 Closed Loop Performance Following a Pretune

Section 6.3 discussing convergence of adapted PID parameters shows that the estimated values from the Pretunes are very close to the final values obtained for a specified overshoot and damping.

6.3 Closed Loop Performance

Results presented in this section demonstrate typical performance for the Exact controller. The self-tuner's ability to adjust PID parameters so that user specified overshoot and damping are achieved is shown. Other results verify that the Exact controller can adjust PID values in response to changing process dynamics and changes in performance specifications.

6.3.1 Typical Controller Performance

Figure 6.6 illustrates the EXACT controller's closed loop performance with the temperature controlled pilot plant (autotransformer at 50% output). Following a series of set point perturbations and automatic PID parameter adjustments, the EXACT controller was able to deliver the desired closed loop response.

The specified values for overshoot and damping were 0.25 and 0.20, respectively. For this particular transient, the controller delivered an overshoot of 0.23 and a damping ratio of 0.17. The adapted controller parameters had remained unchanged over five previous set point changes.

Figure 6.6 demonstrates an important point about the EXACT controller's self-tuner. The self-tuner only considers the transient response of the closed loop system to the set point change. Following the completion of the adaptation cycle, at steady state the process measurement oscillates at high frequency within the noise band. These oscillations are

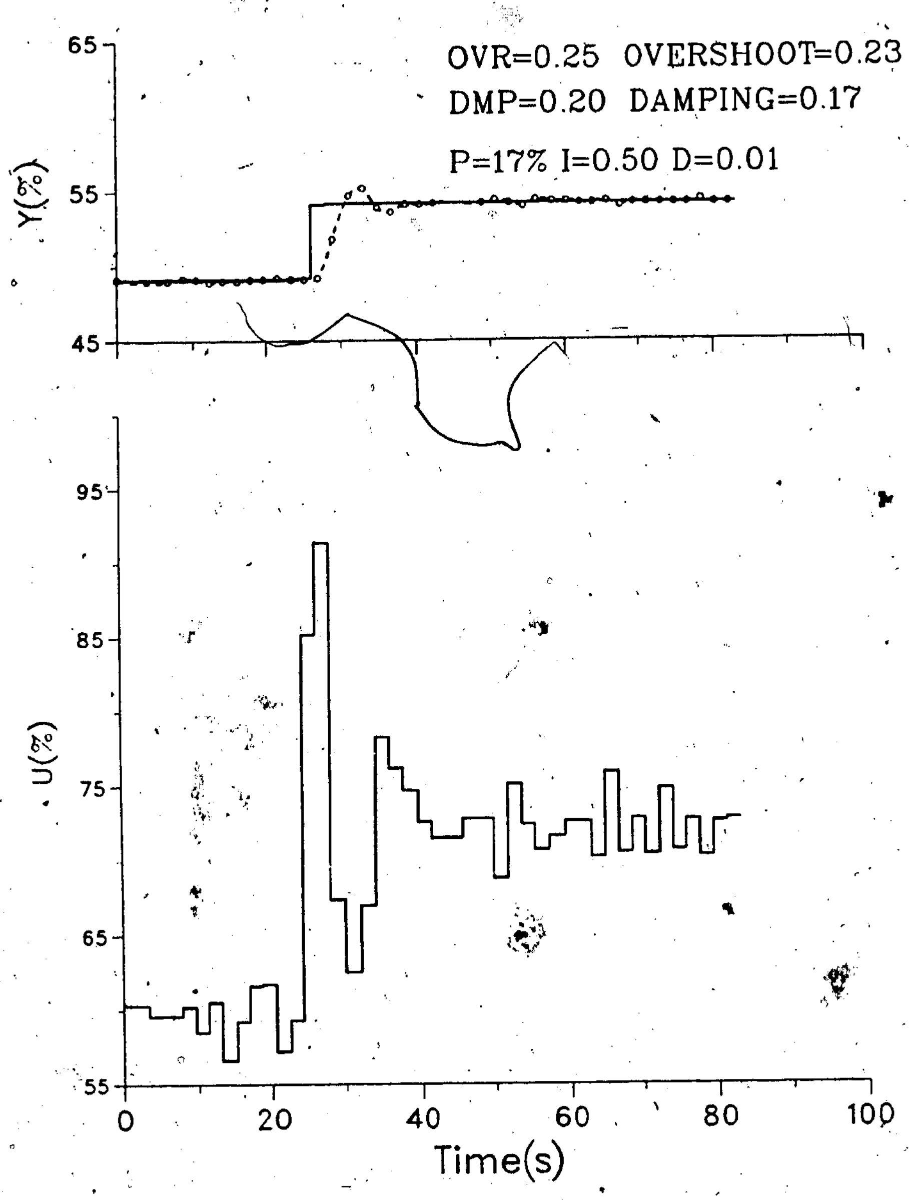


Fig. 6.6 Typical Closed Loop Performance of Exact Controller for a Step Change in Set Point

due to interaction between this loop and the inner air flow control loop. As the tuner increases controller gain in an effort to give desired overshoot and damping, the relative speed of the inner loop in relation to the outer loop, decreases. The EXACT controller's self-tuner will not recognize these oscillations since they fall within the noise band level of one (1) percent.

In order to achieve the desired closed loop response it was required to use smaller set point changes so that the effect of process nonlinearities and asymmetric dynamics could be reduced.

6.3.2 Convergence of PID Parameters

Two runs were performed to demonstrate the EXACT controller's ability to deliver a specified response and converged PID parameters given poor initial estimates (PF=100, IF=0.5 and DF=0.0). For each run, a series of set point perturbations were used to provide transients for the self-tuner. Both runs were performed about the same operating point (52.5%) using different specifications for overshoot and damping. Table 6.1 contains the final PID parameters and the resulting patterns of closed loop response.

TABLE 6.1 Parameter Convergence Using the Exact Controller

Run	P(%)	I	D	OVR	DMP	OVR act	DMP act	Setp.
4	21	0.07	0.02	0.10	0.10	0.11	0.10	52.5
5	17	0.09	0.01	0.25	0.20	0.24	0.25	52.5

Figures 6.7 and 6.8 contain plots of process measurement, set point and controller output for runs CODGF.4 and

CODGF.5. In both cases the closed loop response to set point changes is initially sluggish due to low controller gains. As PID values are adjusted following each set point transient, closed loop performance improves. For CODGF.5 with the larger overshoot and damping specifications, approximately fourteen transients were required before the EXACT controller was able to give the desired closed loop pattern. With more conservative closed loop specifications for run CODGF.4, approximately twelve set point changes were required. Figure 6.9 contains parameter trajectories as a function of tuning transient number for proportional band and integral time. These plots indicate that the EXACT's self-tuner delivers a better than linear rate of convergence (near quadratic) of PID parameters. Note the corresponding values obtained from a Pretune run indicated by the dashed line. These initial values are reasonably close to the final values and would have required only five or six tuning transients before giving converged results. Derivative action was not shown because it remained relatively constant throughout the series of set point changes. The above results indicate that if initial PID parameters are not available the Pretune option should be exercised to obtain good initial values and minimize the number of subsequent tuning transients required to give converged controller constants.

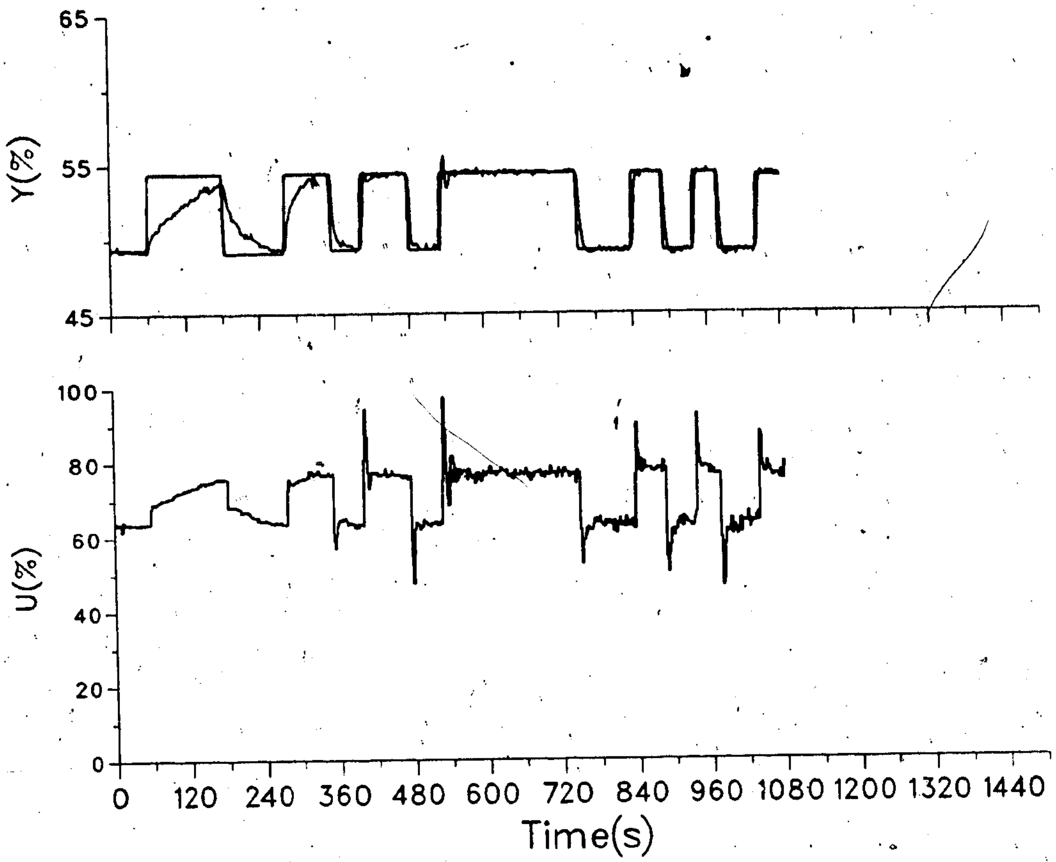


Fig. 6.7 Convergence of PID Parameters About 52.5% Operating Point (OVR=0.1 Run CODGF.4 DMP=0.1)

6.3.3 Tracking Ability of Self-Tuner

One of the prime motivations for adaptive control is the need for a controller that can automatically adjust its controller parameters in response to changing process dynamics. As shown in Chapter five the temperature controlled process is highly nonlinear. To demonstrate tracking performance the following sequence of steps were performed:

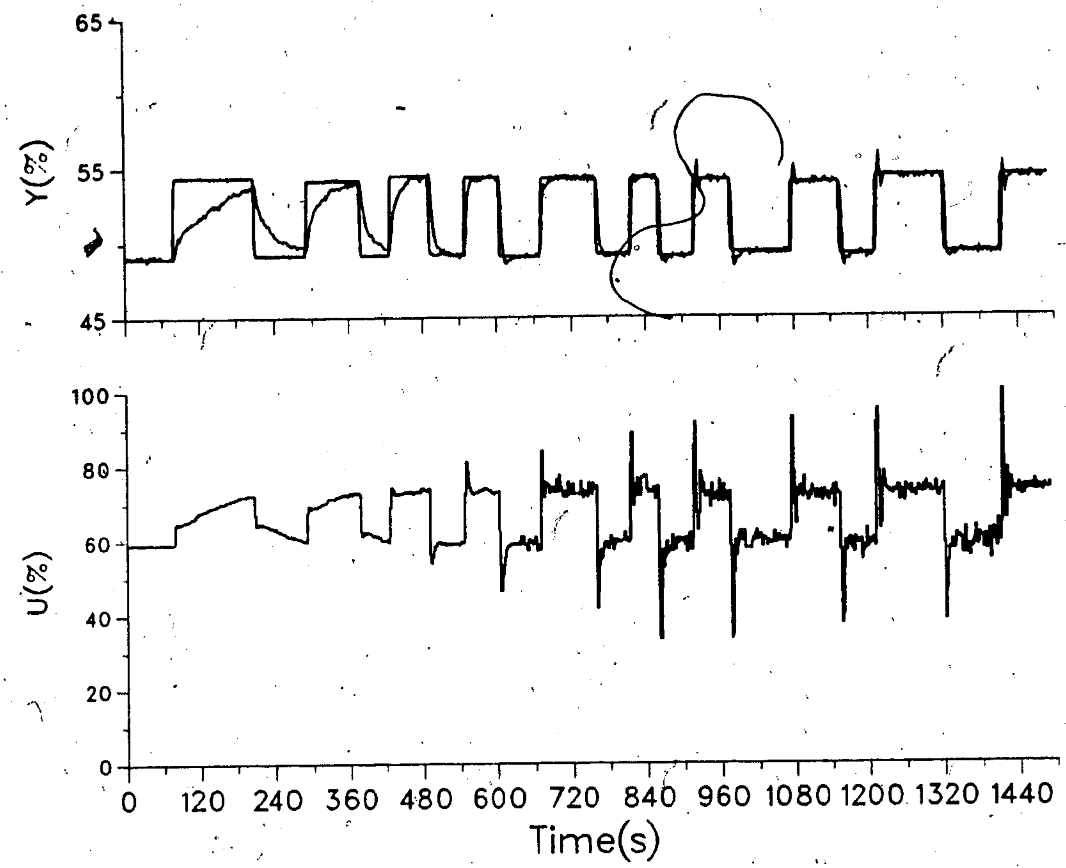


Fig. 6.8 Convergence of PID Parameters About 52.5% Operating Point (OVR=0.2 Run CODGF.5 DMP=0.2)

1. A Pretune followed by a sequence of set point perturbations about the 60% operating point was used to obtain converged PID parameters and stable closed loop control.
2. A series of negative step changes in set point were then used to move the process temperature to a lower operating point corresponding to 30% measurement span.
3. Several additional perturbations were then used to allow the self-tuner to give converged PID parameters.

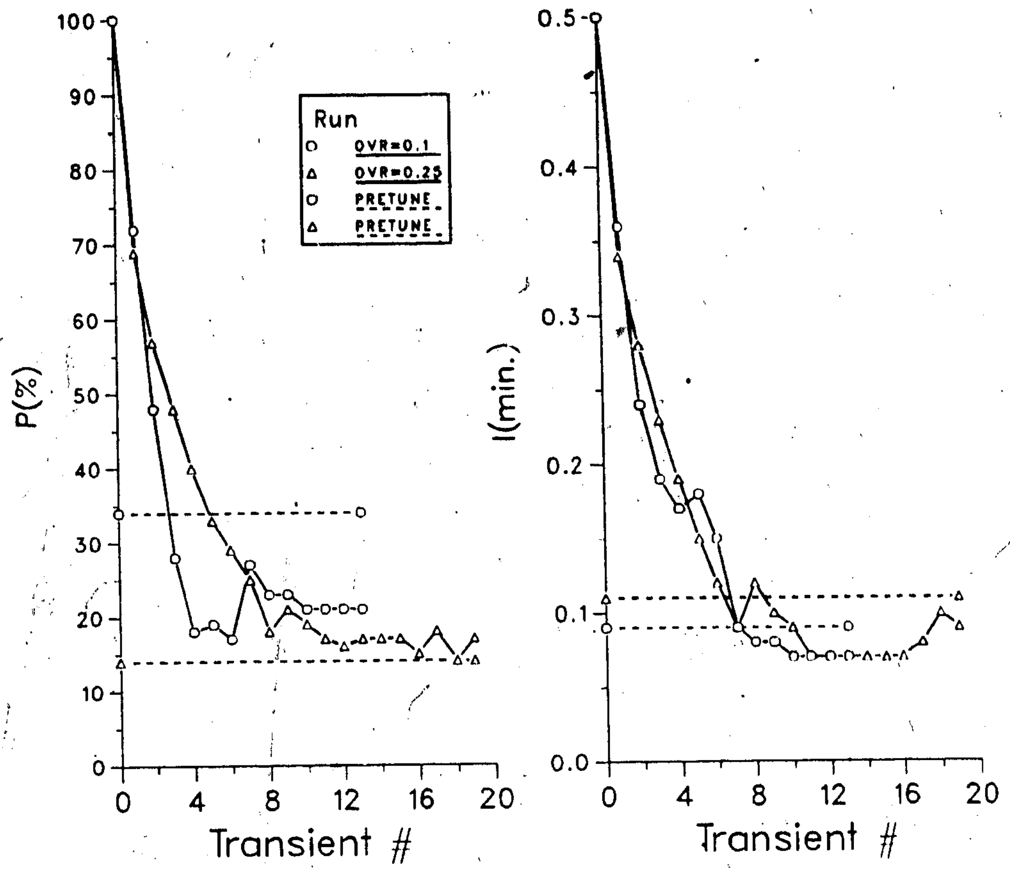


Fig. 6.9 Parameter Trajectories Demonstrating Convergence of P and I

Figure 6.10 illustrates closed loop performance for steps two and three. Figure 6.11 shows the corresponding trajectories for proportional band and integral time through the series of set point changes.

As the set point was decreased and the process measurement moved towards the high (open loop) process gain region, the controller gradually "detuned" in an effort to maintain specified closed loop performance. The third and fourth transients show that the EXACT controller's a-posterior, once-per-transient tuning philosophy does not allow it to detune fast enough. As a result, the fourth set point transient was oscillatory with a damping of 0.33. As the set point was moved towards the 30% operating region, the error response became more oscillatory due to high controller gains.

By using a square-wave sequence of set point changes an operator can give the Exact controller time to "catch up" by giving it additional transient responses as was done in the 40-45% range.

These results demonstrate that the EXACT controller maintains its tracking ability and can handle nonlinear processes. They also demonstrate a limitation of its tuning philosophy: parameters are only adjusted after an appropriate error transient, but not continually during arbitrary transients, as in continuous forms of identification. Estimated parameters are not adjusted until the tuning cycle has completed.

6.3.4 Tailoring of Closed Loop Response Characteristics.

Using the tank level controlled process as described in Chapter five, an experiment was performed to demonstrate the EXACT controller's performance following a change in performance specifications. For a tank level corresponding to 25% of measurement span and the following values for required inputs:

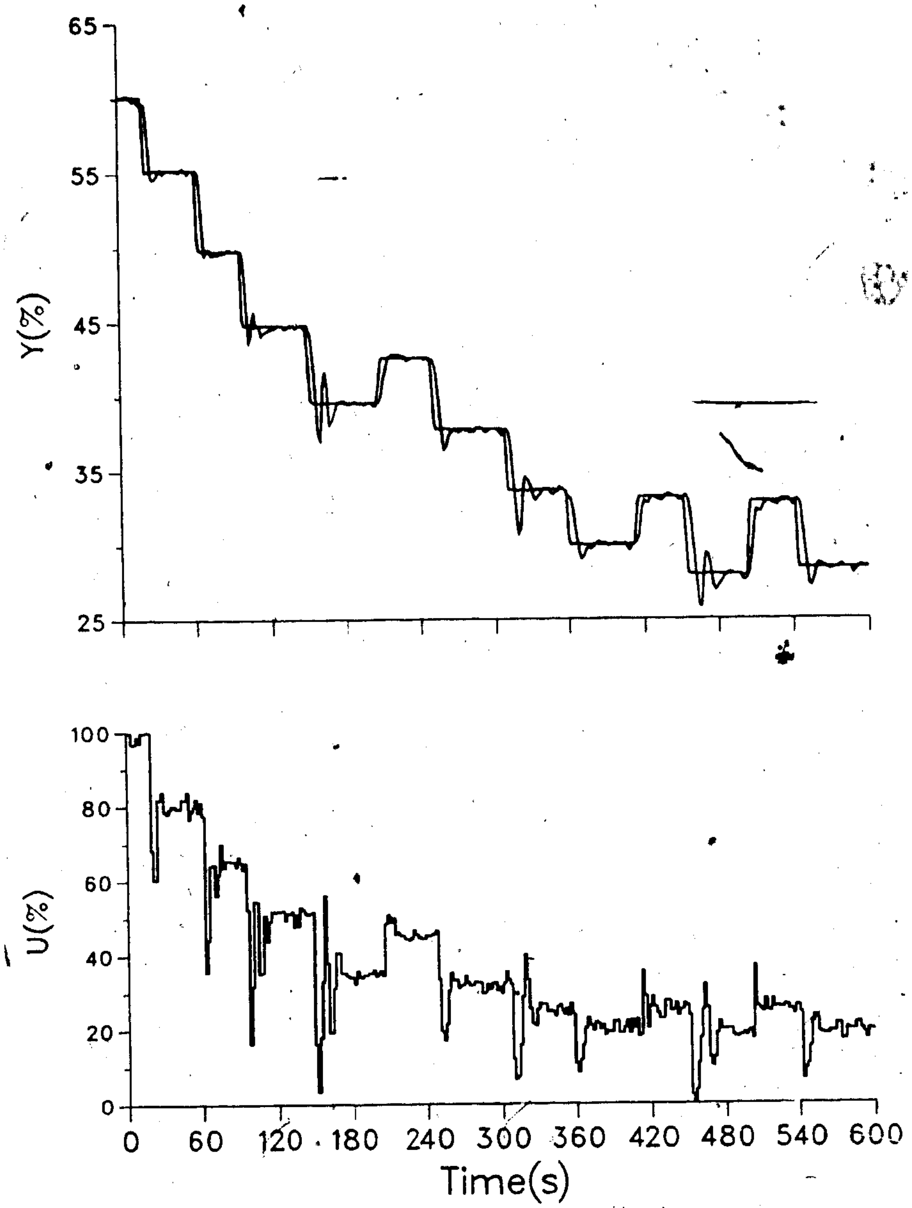


Fig. 6.10 Tracking Ability of Exact Controller (series of -5% set point changes to 30%) Run CODGF.7

WMAX=3.0 min. NB=1.3% OVR=0.5

LAG=0.1 min. DFCT=0.68 DMP=0.3

a series of set point perturbations were used to obtain

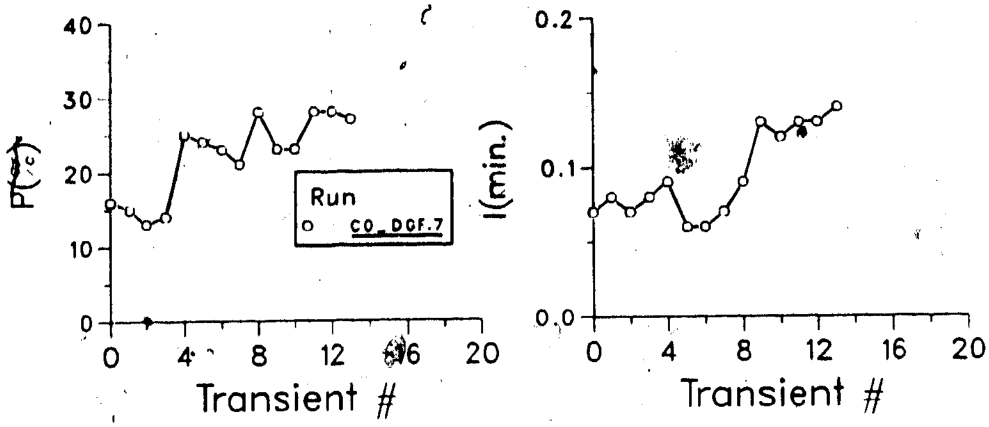


Fig. 6.11 Parameter Trajectories of P and I (for a series of -5% set point changes to 30%)

converged PID values of:

P=17% I=0.41 min. D=0.07 min.

Following parameter convergence the specified overshoot and damping limits were changed to:

OVR=0.0 and DMP=0.1

in order to give a critically damped response. Figure 6.12 illustrates subsequent closed loop performance for a series of set point perturbations between 25% and 30%. The first transient response had an overshoot of ≈30% and a damping ratio of ≈25%, after which the self-tuner increased the proportional band in an effort to detune. After approximately five set point changes, subsequent closed loop responses were critically damped. Figure 6.13 contains parameter trajectories for the corresponding series of set point changes. The final set of PID parameters were:

P=46% I=0.65 min. D=0.09 min.

The more conservative specifications resulted in smoother control action since controller gain was reduced by a factor of 2.7.

The reader should note that the digital filtering option was used for the above runs with a six second time constant. The use of filtering effectively increased the order of the closed loop system making it easier for the EXACT controller to force the system to oscillate. Refer to the following section for a discussion of highly damped processes and the problems associated with the specification of overshoot. In practice a control engineer would not use digital filtering in order to allow a closed loop system to become oscillatory and thus increase closed loop response times.

6.4 Evaluation Under Selected Process Conditions

Results in the previous section demonstrated the self-tuner's ability to meet the basic requirements for an adaptive controller under typical conditions. In this section, the effects of specific process conditions on closed loop performance are presented. Results show that these practical problems must be considered for a generic controller design that will be useful for a wide range of

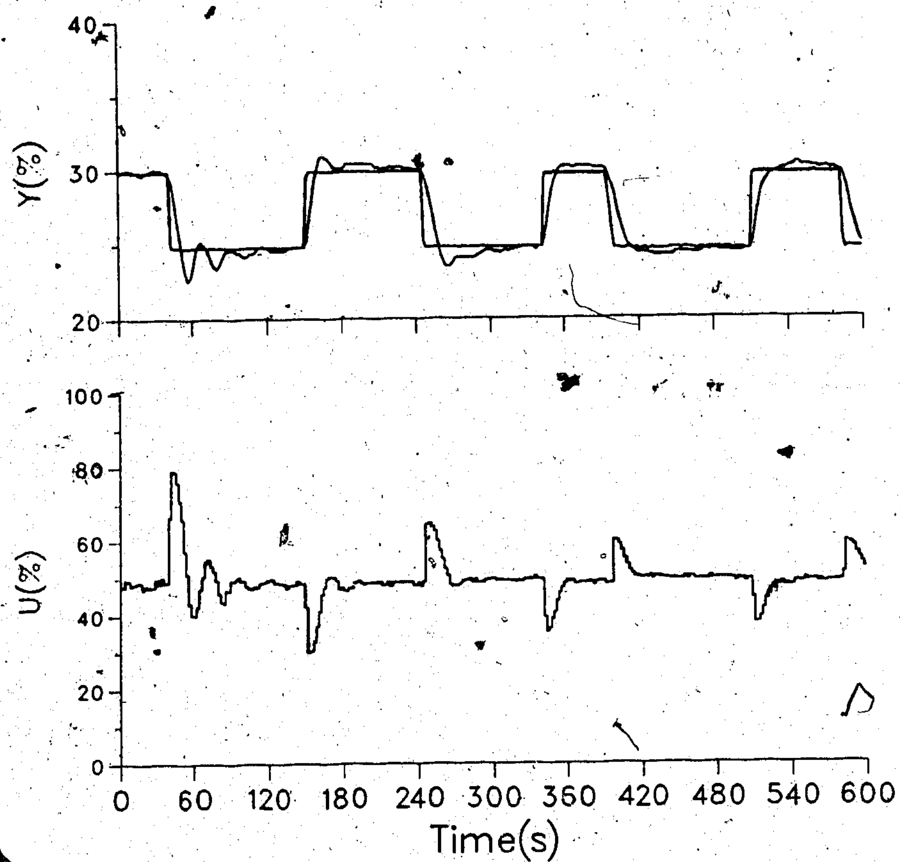


Fig. 6.12 Tailoring of Closed Loop Response Characteristics Run C_2_1.2

process conditions.

6.4.1 Overdamped Processes

The liquid level process is a dominantly damped first order process with a time constant of approximately five minutes and a steady state gain of 5.0(Δ%measurement span/Δ%controller output). This pilot plant was chosen for its approximately linear, first order dynamics because it was thought that the adaptive controllers would have no difficulty in controlling its tank level.

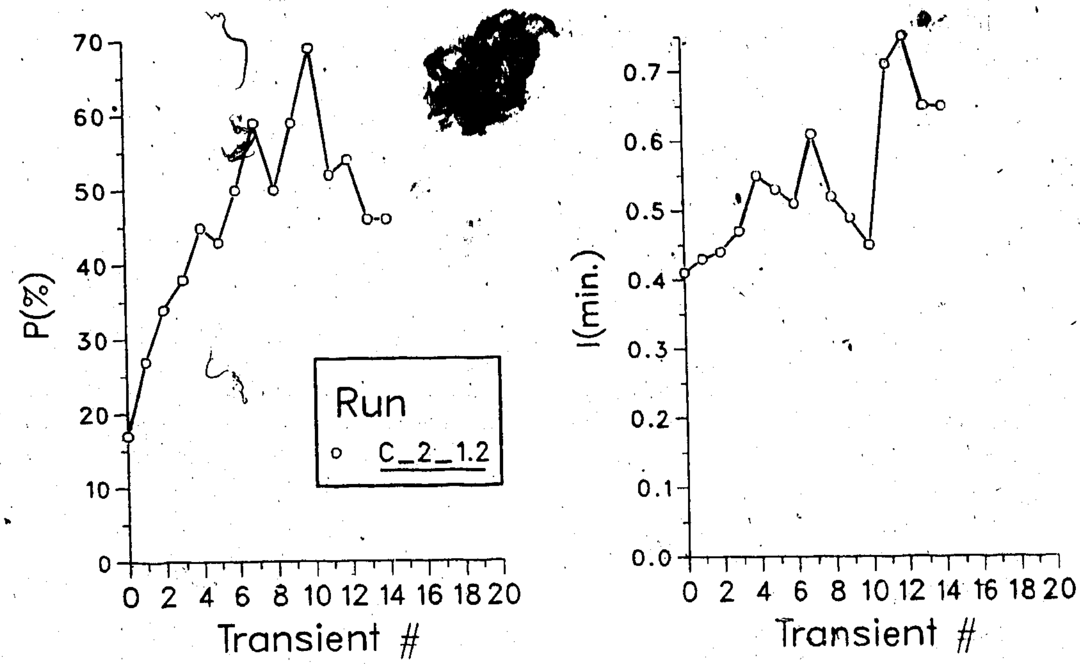


Fig. 6.13 Parameter Trajectories of P and I for Pattern Tailoring

This section discusses a problem associated with the specification of overshoot for overdamped processes. Following the operating guidelines for the EXACT controller, a Pretune was used to obtain initial estimates for required inputs. Figure 6.14 illustrates the open loop test and shows the final results. The process measurement was initially allowed to reach a steady state level corresponding to 24.4% of measurement span for a controller output of 48.8%. The Pretune sequence used a BUMP of -5.0%. Note that the process measurement was not allowed to reach steady state after the BUMP had been implemented. The controller output was returned to its original value when the change in process measurement reached 10% of span (see section 6.1.4). As can be seen from the results, the process measurement is relatively noise free(NB=1%) and the process time constant was relatively small (WMAX = 0.5 min.).

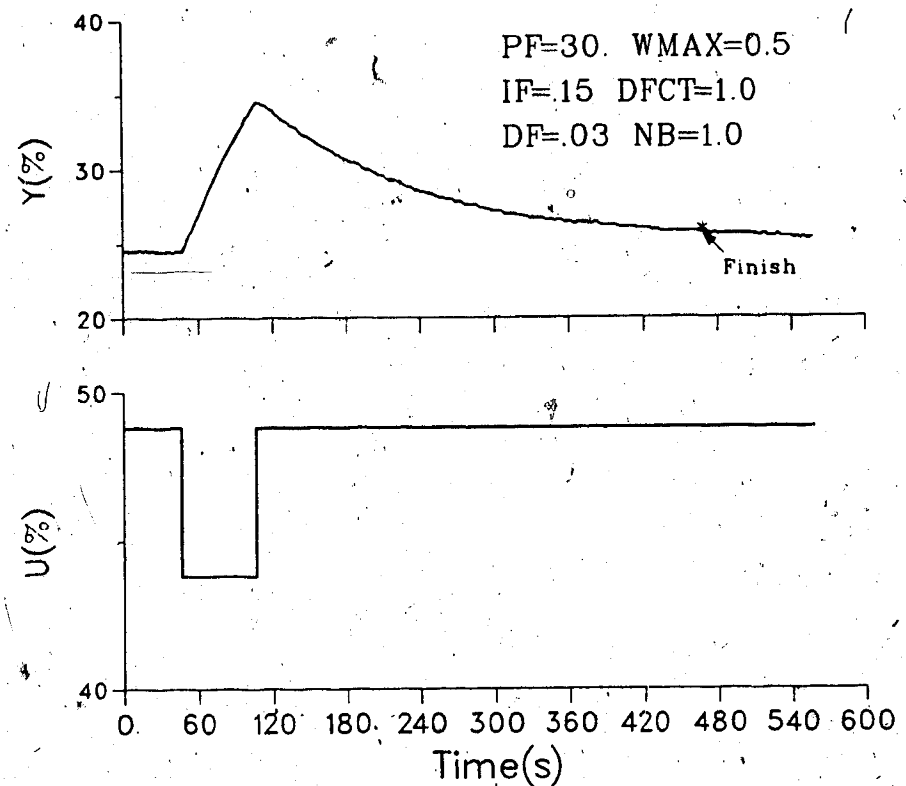


Fig. 6.14 Pretune (P_1_1.2) for Level Controlled Process. OUT=48.8% MES=24.4% BUMP=-5% LAG=0.0 min.

Using the above Pretune results and the default overshoot and damping specifications (OVR=0.5 DMP=0.3) a series of set point perturbations were then used to enable the self-tuner to adjust the PID parameters. The self-tuner continually decreased the proportional band after each transient. After ten set point changes the EXACT controller's tuner had decreased the controller band from thirty to one (1). This phenomena which will be referred to as gain wind-up, results in unacceptable, high frequency oscillations in controller output. Figure 6.15 illustrates the closed loop performance for the series of set point changes and the eventual gain wind-up problem. When closed loop control was disabled the PID parameters had been adjusted to values of:

P=1%, I=0.15 min. and D=0.03 min.

It is emphasized that the gain wind-up problem described above is due to unreasonable specifications for limits on overshoot and damping and not due to poor controller design. For dominantly damped processes such as this the operator of an EXACT controller must use more conservative overshoot and damping specifications.

This is demonstrated below for a run performed using the same initial conditions as above but overshoot and damping limits of OVR=0.0 and DMP=0.1. The resulting controller output was much less oscillatory due to lower resulting controller gains as shown in Figure 6.16.

The trajectories for P and I in Figure 6.17 indicate that the proportional band is continually decreased after each set point transient even though the transient responses exhibit approximately zero overshoot.

It has been observed that the EXACT's self-tuner appears to drive the controller gain as high as possible for the given performance specifications. Subsequent set point changes resulted in even higher controller gains corresponding to a proportional band of 10%, and hence even more oscillatory control action.

For any adaptive control scheme in which the performance

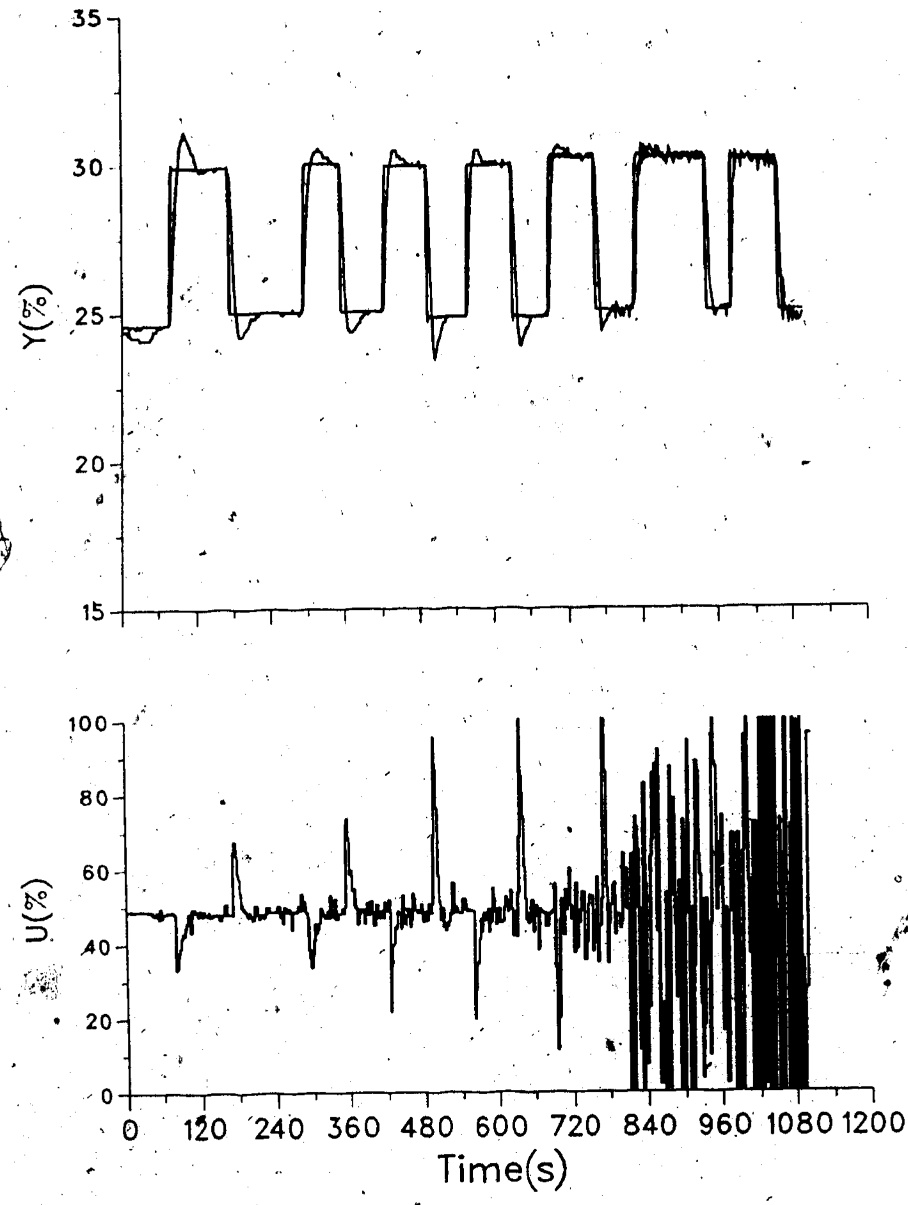


Fig. 6.15 Gain Wind-Up Due to Unreasonable Specifications for OVR=0.5 and DMP=0.3 (C_1_1.2)

index does not take the resulting controller output into consideration, care must be taken. The EXACT controller adjusts the PID parameters based only on the error response of the closed loop system following a disturbance. The

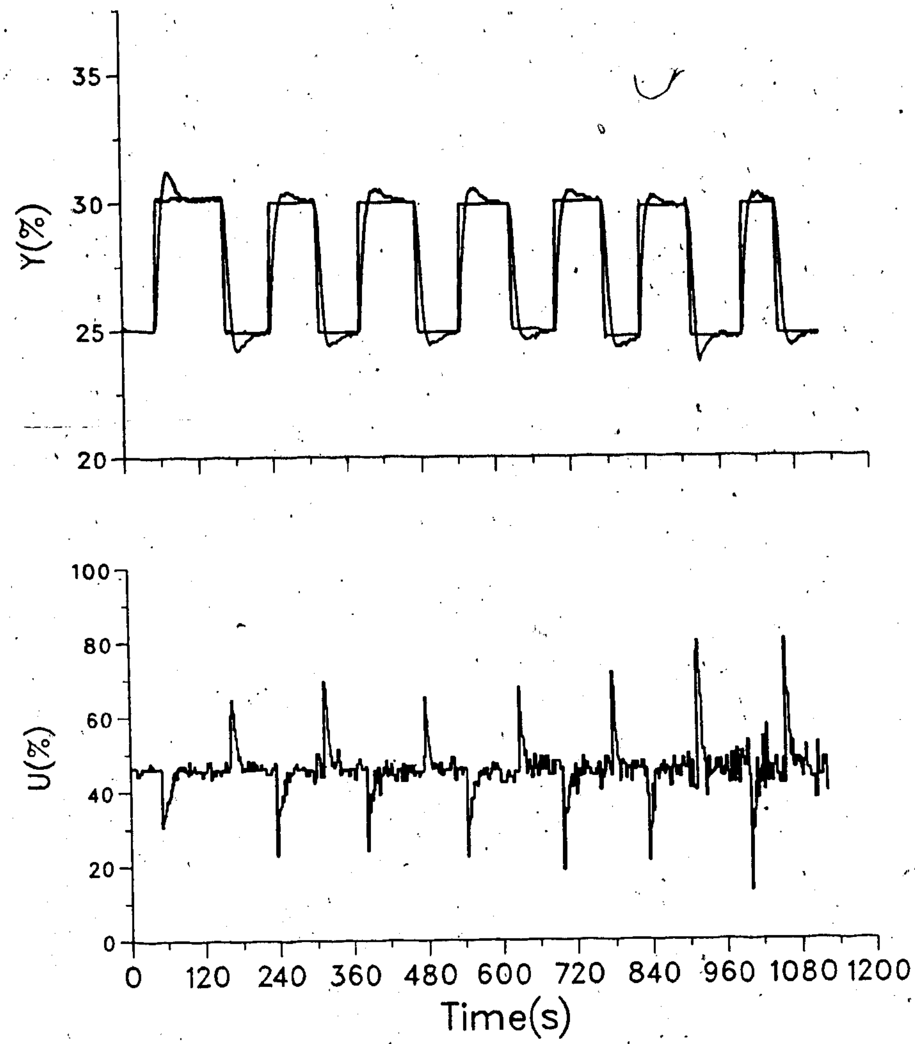


Fig. 6.16 Use of Conservative Performance Specifications to Prevent Gain Wind-UP
(OVR=0.0 DMP=0.1) (C_i_6.2)

self-tuner does not consider the resulting controller output when adjusting controller parameters. That is one reason why Foxboro has provided the two features for clamping of PID values and the automatic detuning of band [illegible] derivative time when control output is too oscillatory. The detuning feature is particularly important for processes subjected to large load disturbances [illegible] controller gain wind-up may

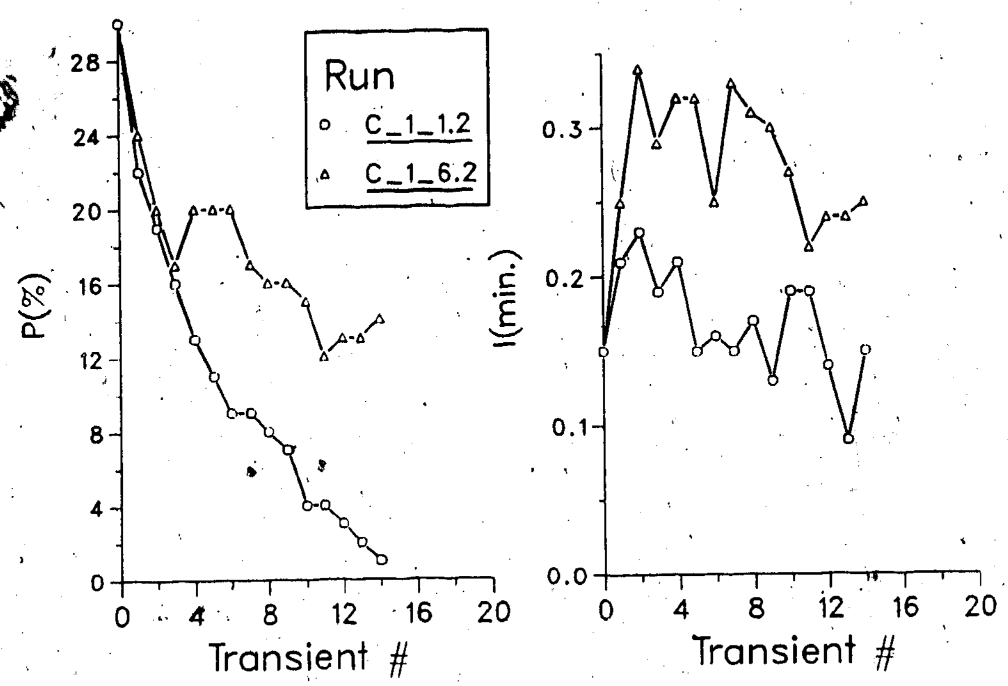


Fig. 6.17 Parameter Trajectories Using Conservative OVR=0.0 DMP=0.1

occur when an operator is not supervising the EXACT's operation.

The above problem is analogous to minimum variance type control where variations in process measurement only are considered in the objective function. The GMV self-tuning controller includes a penalty on control action in its performance index which can be used to control the amount of variation in u(k), at the expense of greater variation in the predicted set point error.

An ad hoc solution to the gain wind-up problem involves the use of digital filtering to effectively add dynamics to the controlled plant. This then allows the sel[illegible] give PID parameters that will result in oscil[illegible]rol and hence allow for overshoot specification. This is not a practical solution since the addition of dynamics through measurement filtering results in more sluggish control and longer response times. Section 6.5.1 demonstrates this technique.

6.4.2 Processes With Asymmetric Dynamics

Many plants exhibit not only nonlinearities but behave differently depending on the direction of process measurement changes, ie. (heating vs. cooling dynamics). The temperature controlled pilot plant demonstrates such behavior. Figure 6.18 illustrates the effect of asymmetric dynamics on the EXACT controller's closed loop performance for a series of ±5% set point changes.

The negative changes in set point resulted in oscillatory transient responses with average values of 0.38 for overshoot and 0.27 for damping. Conversely the positive set point changes resulted in near damped responses with an average overshoot of 0.10. The parameter trajectories in Figure 6.19 show that the proportional band and integral constants were oscillating through the series of set point changes about particular values.

The above results indicate that a once-per-transient adaptation mechanism such as the EXACT's can not give converged constants for the chosen sequence of set point changes.

If it were desired to obtain converged PID parameters, a

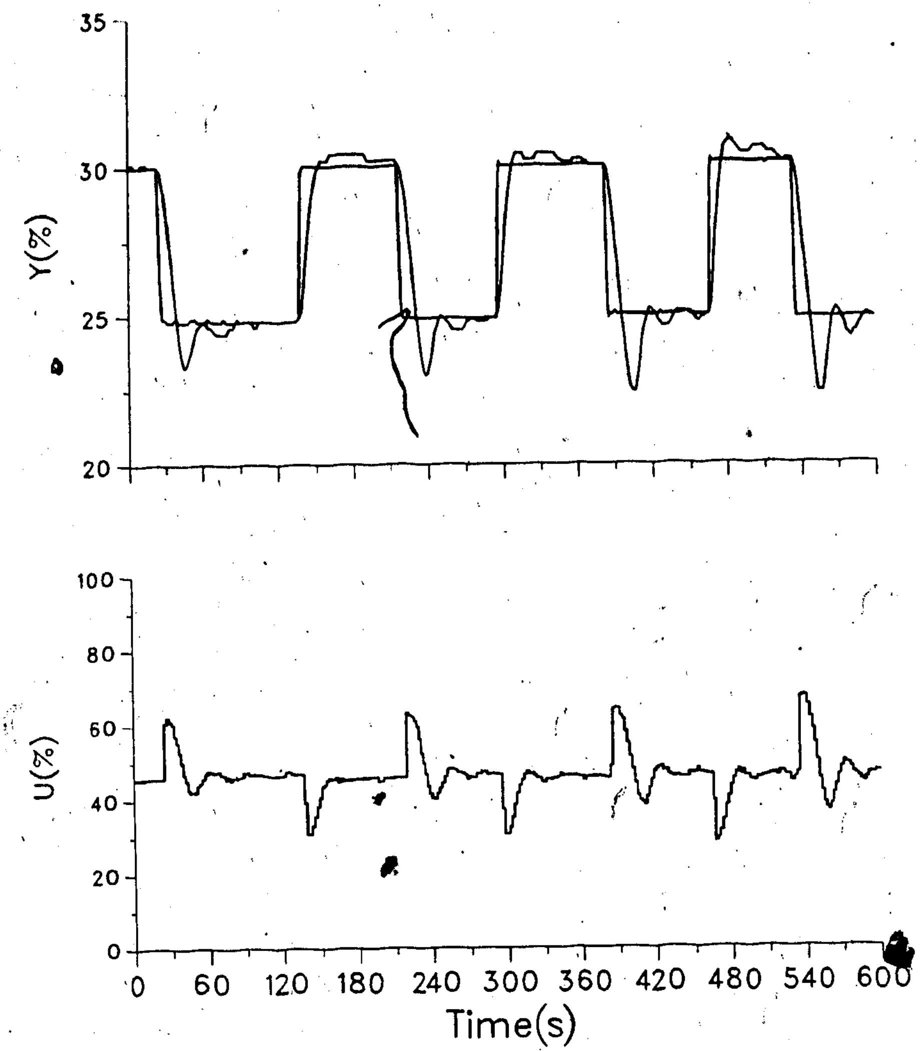


Fig. 6.18 Effect of Asymmetric Dynamics on Performance (C_3_0.2 LAG=0.1 min. WMAX=3.0 min. DFCT=0.68 OVR=0.3 DMP=0.2)

series of unidirectional set point perturbations would be more appropriate. Four unidirectional set point changes from 30 to 25% were implemented. After each transient response and completed adaptation cycle, the process set point was slowly ramped back to 30% so that the self-tuner was not activated. The final transient response shown in Figure 6.20 had an overshoot of 0.28 and a damping of 0.19 which is

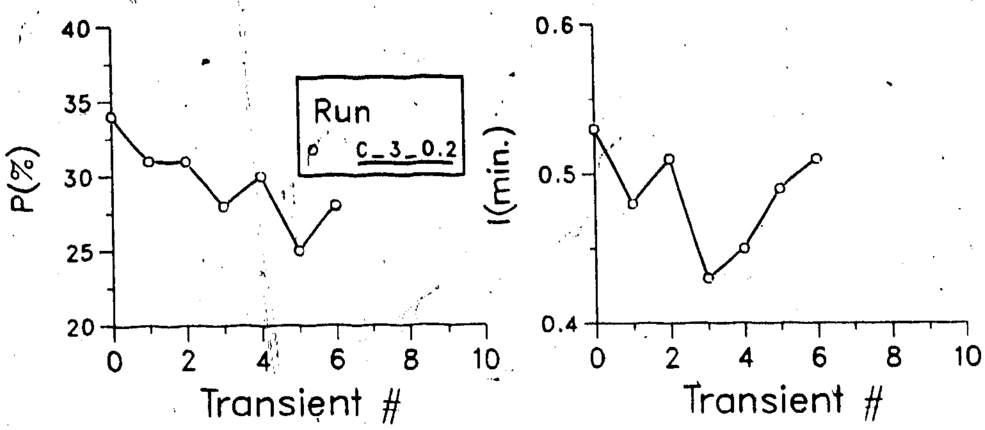


Fig. 6.19 Effects of Asymmetric Dynamics on PID Parameters (C_3_0.2)

equivalent to the specified values of OVR=0.3 and DMP=0.2.

For both the positive and negative series of set point changes, Figure 6.21 illustrates the convergence of measured overshoot and damping to the specified limits. Similarly the parameter trajectories are shown in Figure 6.22 for the same series of transients.

For the series of positive set point changes from 25 to 30%, the final set point transient in Figure 6.23 had an overshoot of 0.20 and a damping of 0.15. PID parameters had converged to values of 21%, 0.46 min. and 0.08 min., respectively.

Adaptive control systems that adjust controller parameters based on previous transient responses can not be expected to give user specified performance when operating points are moved or process dynamics change. Adaptive controllers that use recursive parameter estimation schemes should be able to track such changes faster and hence

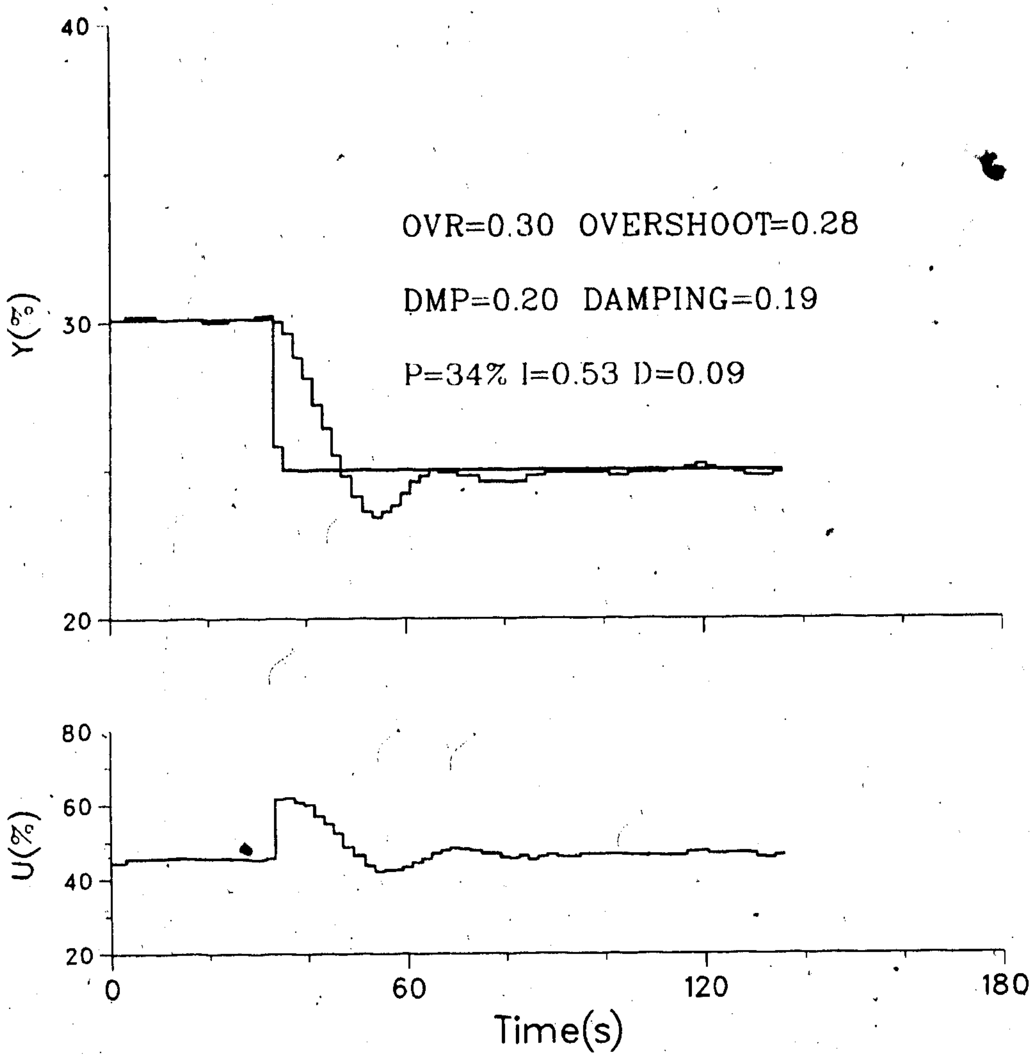


Fig. 6.20 Final Transient from a Series of Negative Set Point Changes (C_1_0.3)

improve control performance.

6.4.3 Measurement Noise

In real applications measured process output signals are often corrupted by nearby external signals. It is generally assumed that these superimposed signals can be characterized by some function of random noise that has zero mean and

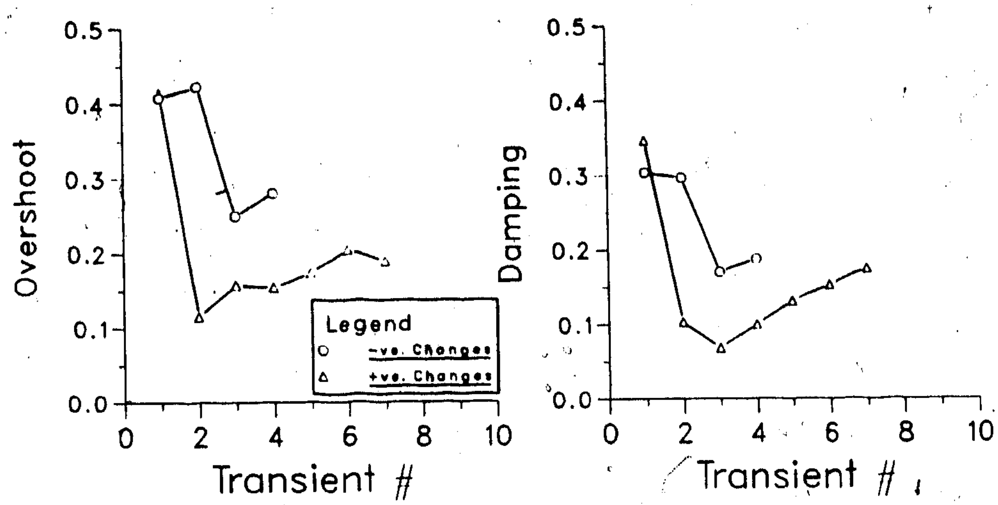


Fig. 6.21 Convergence of Closed Loop Performance to Desired Pattern

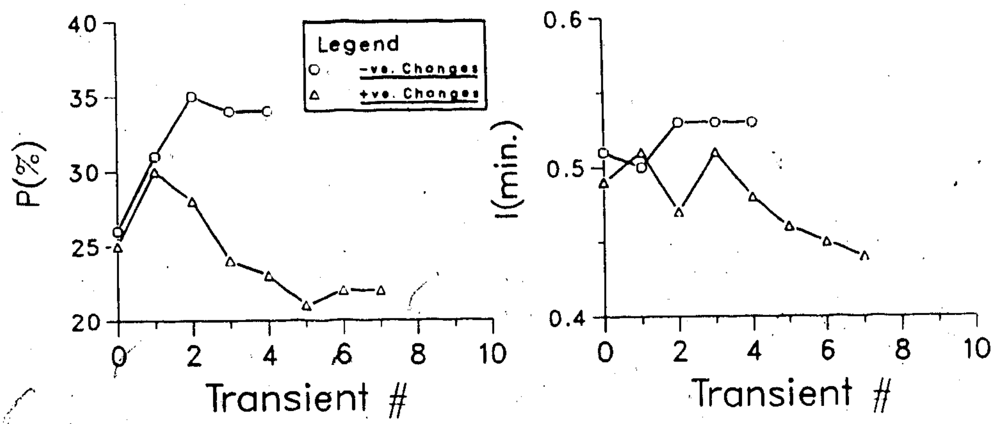


Fig. 6.22 Convergence of PID Parameters for Two Series of Unidirectional Set Point Changes.

known variance. This measurement noise contains no useful dynamic information. Adaptive mechanisms if not "protected" from noise may adjust controller parameters inappropriately, resulting in degraded closed loop performance.

If the noise signal is not random and its dynamic model

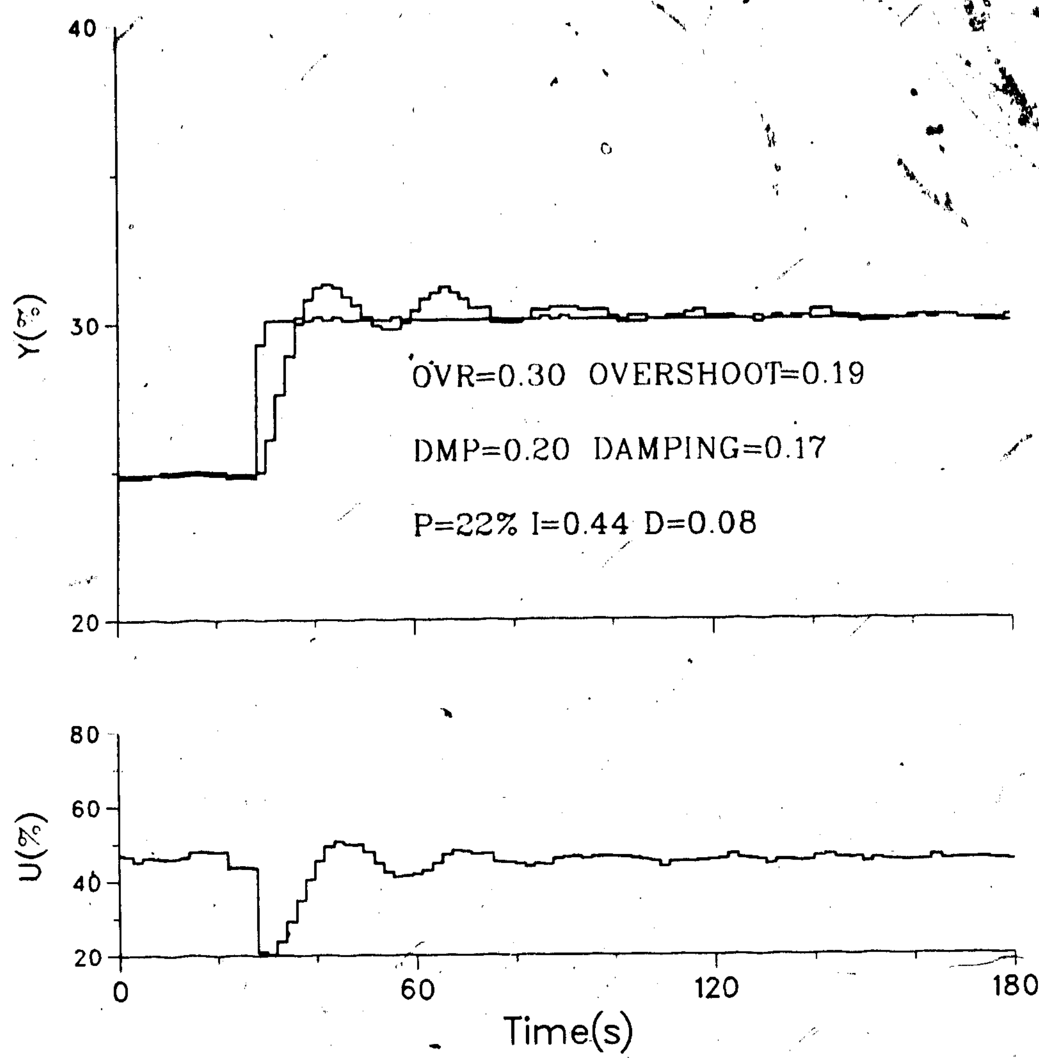


Fig. 6.23 Final Transient from a Series of Positive Set Point Changes (C_3_6_2)

structure known, it should either be identified or removed from the measured signal. Noise signals generally have high frequency content and can therefore be removed using low pass filters.

The EXACT's tuner handles noisy signals by ignoring set point errors until they exceed some specified dead band. Tuning is based only on transient information that exceeds the limits of this specified noise band, NB.

If noise levels exceed the specified band the tuner will be activated unnecessarily. If these noise signals contain high frequency oscillations with period T_n such that WMAX is greater than $8T_n$ the tuner will not adjust PID parameters unnecessarily. The EXACT controller's tuner is not susceptible to high frequency noise and the resulting problems associated with parameter drift.

Several open loop runs were performed to first determine if high levels of superimposed noise actually entered the EXACT controller and to what extent that noise was filtered. Using the random, high frequency noise generator described in section A.3, the following levels of noise were superimposed on the measurement input signal to the EXACT controller, for the temperature controlled process.

TABLE 6.2 Levels of Superimposed Noise on Controller Input.

Level#	Noise Variance (V)	Noise Variance (%)
1	0.5	12.5
2	1.0	25.0
3	1.5	37.5
4	2.1	52.5

For a constant controller level of 53.6%, Figure 6.24 indicates that superimposed noise is filtered out. The digital filter constant, LAG, was set to zero for these runs. The true process measurement was 46.6% of measurement span. As noise levels were increased, the measured signal did oscillate somewhat, but not a significant amount.

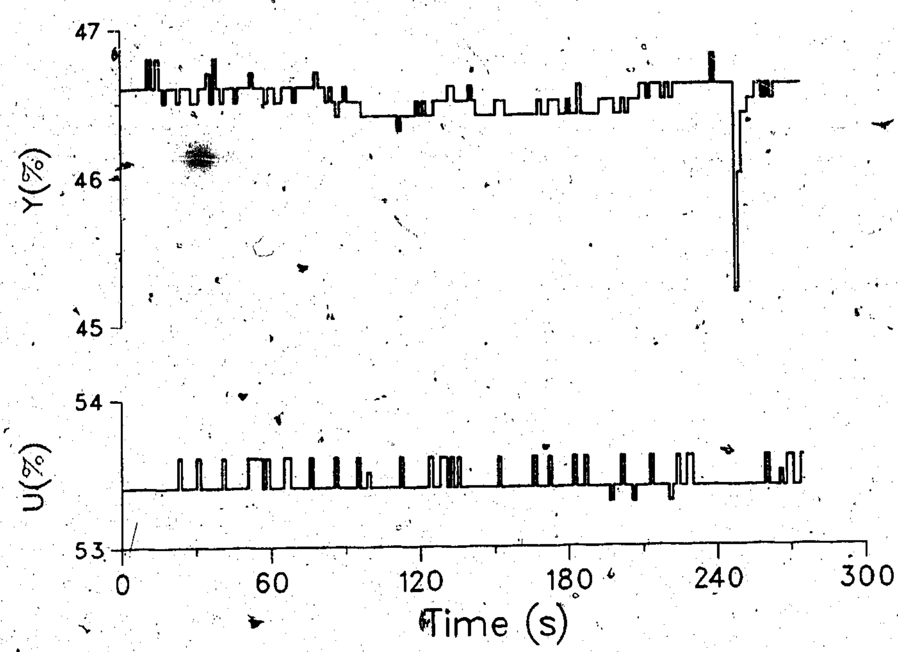


Fig. 6.24 Effect of Superimposed Noise on Process Measurement Levels for the Exact Controller (0_7_1.5)

A second run was performed using the fourth level (52.5%) of noise and resulting oscillations were still not significant.

6.4.4 Effects of Slowly Drifting Disturbances

The EXACT controller's self-tuner is only activated when disturbances force the set point error to exceed twice the noise band. Slowly drifting disturbances or changes in process dynamics will not perturb the closed loop system enough to force the self-tuner to adjust PID parameters. If the resulting changes in process dynamics are significant, the closed loop system may become unstable or very sluggish. In the former case the EXACT controller may adjust PID parameters appropriately if the resulting oscillations in process measurement exceed the noise band limit.

The following series of experiments demonstrate how slowly drifting disturbances can affect control performance. Figure 6.25 contains the two open loop, steady state characterizations for the temperature controlled process. Configuration #1 corresponds to the case when the autotransformer is operated at 50% and configuration #2, at 70%. For a temperature corresponding to 35% of measurement span the static gains for the two configurations are approximately 0.6 and 2.0, respectively. Using this known information the following series of runs were performed.

1. With the autotransformer at 50% output, a Pretune followed by a square wave series of set point changes was performed to obtain converged PID parameters for specified overshoot and damping values of OVR=0.0 and DMP=0.25 at an operating point of 35%. Pretune used a BUMP of +10.0%.
2. While running at a steady state temperature corresponding to 35% of measurement span, the autotransformer's output was reset to 70%. This effectively increased the open loop static gain by a factor of three to 2.0. The PID parameters determined in

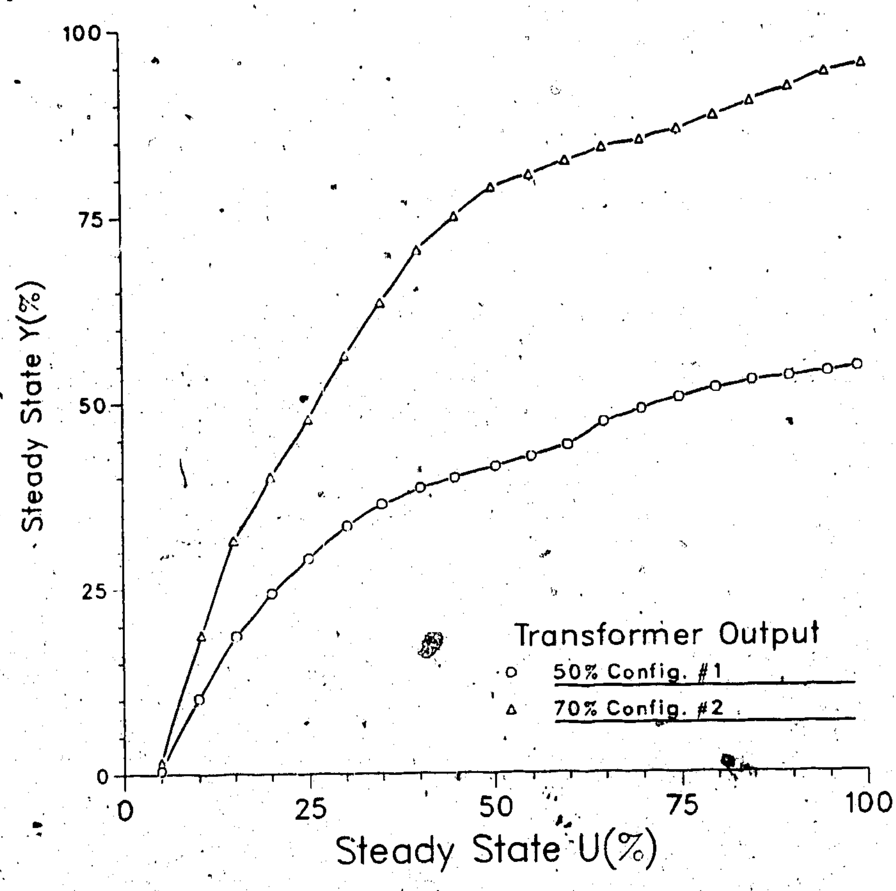


Fig. 6.25 Open Loop Characterizations for Temperature Controlled Process

Step 1 then resulted in oscillatory control.

3 Step 1 was repeated with the autotransformer now at 70% output for the same operating point.

Table 6.3 summarizes the Pretune results from Steps 1 and 3 and clearly indicates that the static process gain has increased based on the difference in recommended proportional band. The converged PID parameters for each series of set point changes are also presented. Note that the recommended maximum wait time is almost twice

TABLE 6.3 Tuning Results at 35% Level for Configurations 1 and 2

Step #	PF(%)	IF	DF	NB	WMAX	DFCT
1	25	0.13	0.02	1.0	0.50	1.0
2	67	0.29	0.05	1.0	0.97	1.0
	P(%)	I	D			
1	20	0.12	0.03			
2	71	0.27	0.05			

as large for configuration #2.

Figure 6.26 shows the Pretune sequence for configuration #1 and Figure 6.27 the subsequent series of set point changes used to obtain converged overshoot and damping. The effect of asymmetric dynamics on transient responses is clear. Negative set point changes towards the higher process gain regions consistently overshoot while positive changes are damped, as is desired. Figure 6.28 shows the closed loop system's response to the increase in transformer output. With the converged PID constants from Step 1, the controller maintains the process temperature at the desired level, thus preventing the self-tuner from turning on. As the process dynamics change and the effective static process gain increases, the closed loop system begins to oscillate within the specified noise band. The result is unsatisfactory closed loop performance. Figures 6.29 and 6.30 show the transient responses for the Pretune and series of set point changes for configuration #2. Controller performance is better following the reestimation of required inputs but is slightly oscillatory at "steady state" due to interaction between the master loop and the inner air flow control loop.

The above results indicate as expected that the EXACT's tuner will not adjust PID parameters in response to slowly drifting disturbances or slowly changing process dynamics until their negative effects on closed loop performance have been realized. If the resulting change in process dynamics is significant, it may be necessary to perform a Pretune to obtain a more appropriate value for WMAX.

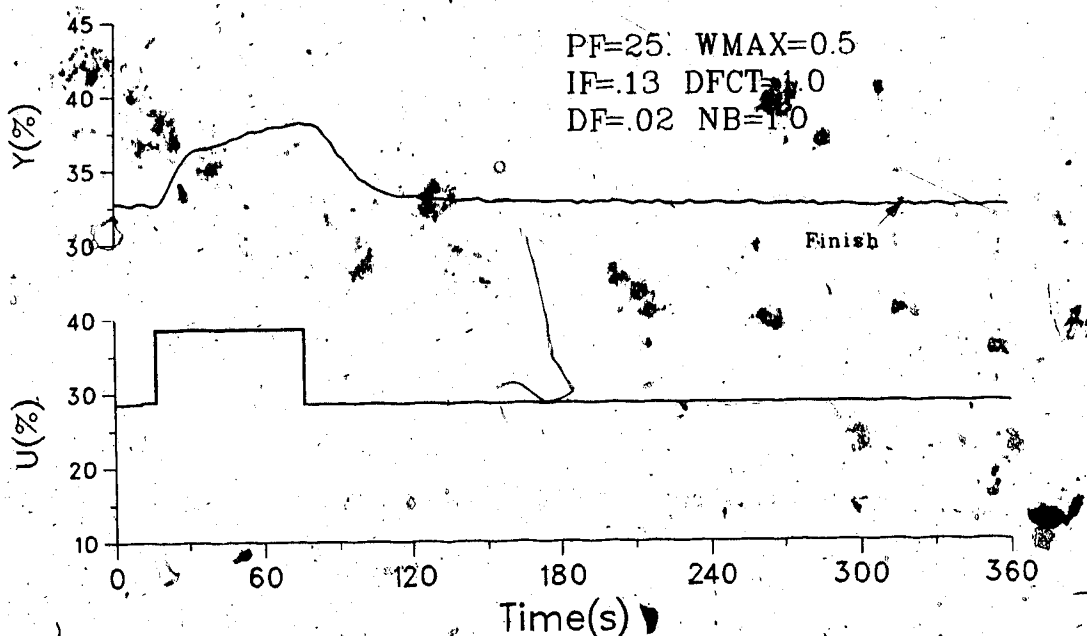


Fig. 6.26 Pretune (config #1) at 35% Operating Point: OVR=0.0 DMP=0.25 BUMP=+10.0%
Run P_6_1.5

6.4.5 Effects of Oscillatory Disturbances

The EXACT controller's self-tuner adjusts PID parameters in response to step changes in set point or load disturbances provided those changes are large enough to

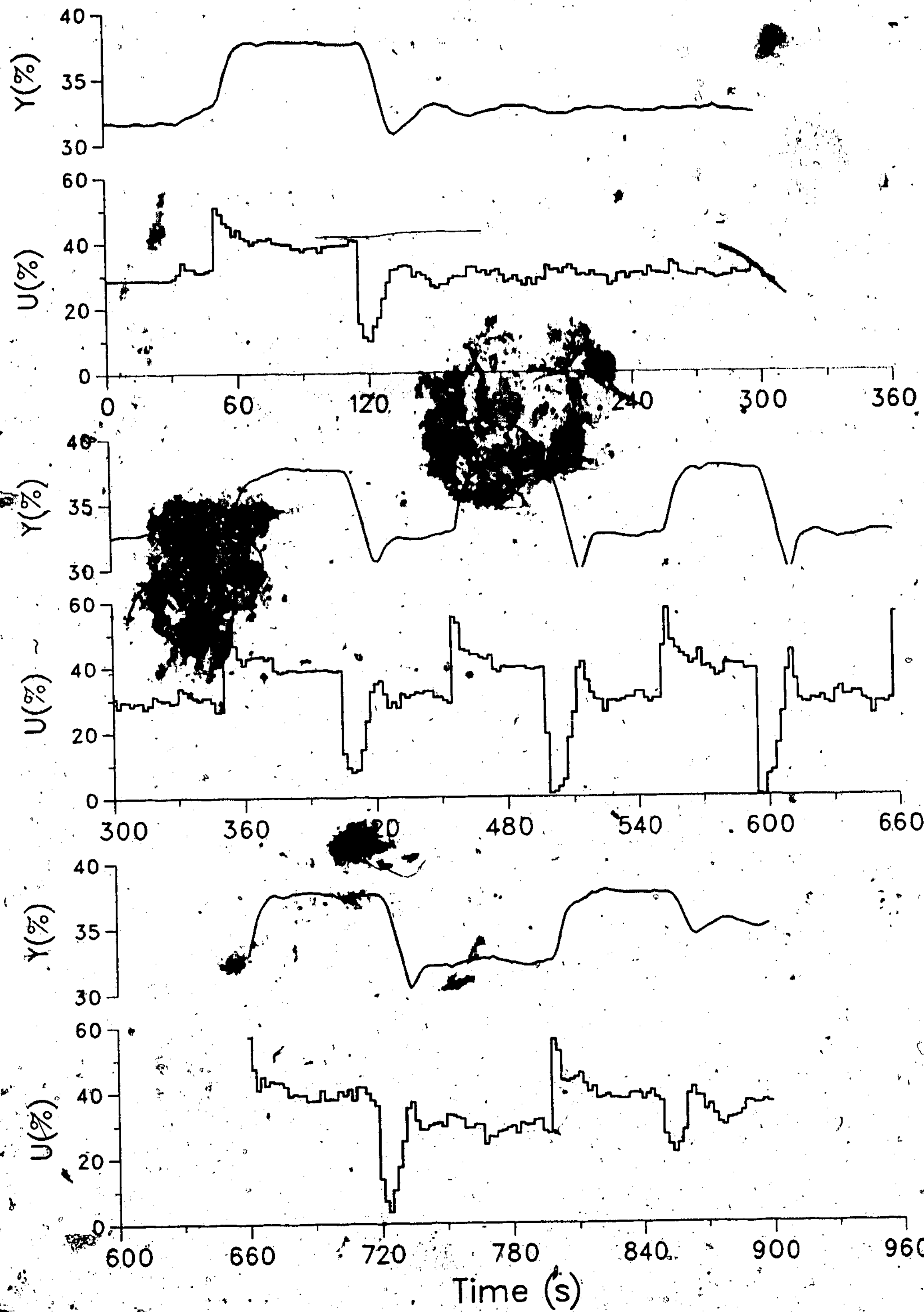


Fig. 6.27 Series of ±5% Set Point Changes About 35% Operating Point: Config #1
OVR=0.0 DMP=0.25 Run C_6_1.5

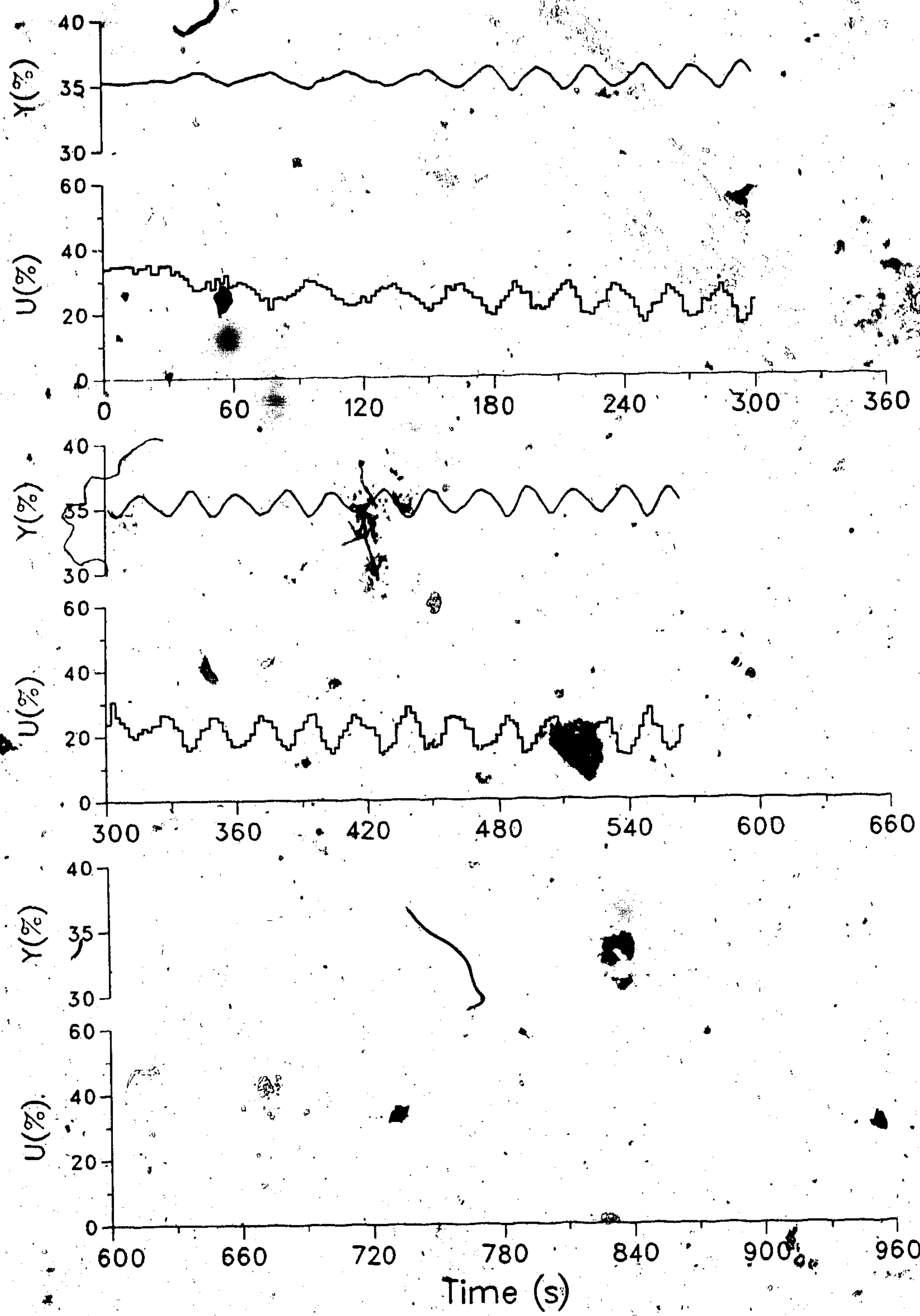


Fig. 6.28 Closed Loop System Response to Change in Autotransformer Output (sudden disturbance) Run C_6_[illegible]5

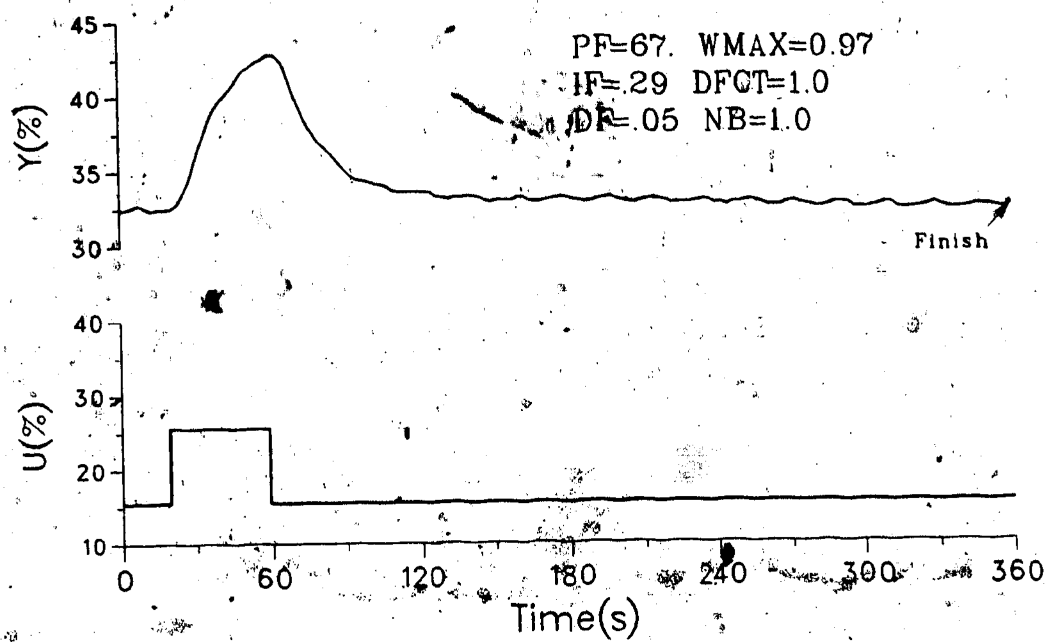


Fig. 6.29 Pretune (config #2) at 35% Operating Point: OVR=0.0 DMP=0.25 BUMP=+10.0%
Run P_6_2.5

generate an error response as illustrated in Figure 6.1. If either the set point or load again change during a transient response, the self-tuner will "restart" the adaptation cycle. If an unmeasured load disturbance or MIMO process interaction was oscillatory and large enough to force the set point error to exceed twice the noise band, the self-tuner although active, would continually restart itself and therefore never adjust its PID parameters which is the appropriate action to take under these conditions.

6.4.6 Processes with Dominant Dead-Time

Neither the liquid level controlled process nor the temperature controlled pilot plant contained significant transport delays and so could not be used to evaluate

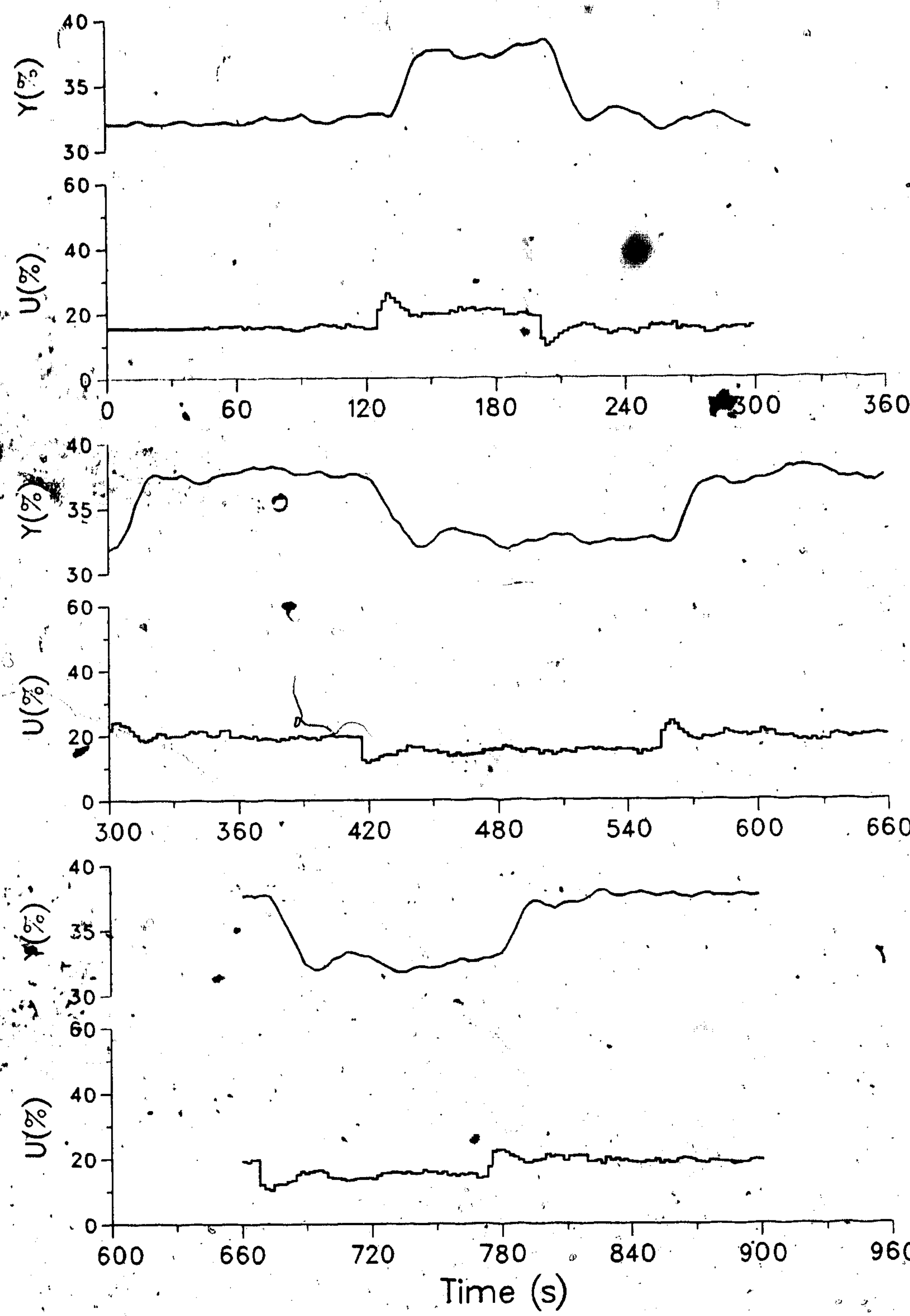


Fig. 6.30 Series of ±5% Set Point Changes About 35% Operating Point: Config #2
OVR=0.0 DMP=0.25 Run C_6_3.5

dead-time effects on controller performance. Some simulated results by Nachtigal(1986a,b) indicate that large time delays result in closed loop responses with longer periods of oscillation. Since the EXACT controller does not use explicit time delay estimation or compensation, this would be expected. The value of WMAX must be appropriately chosen. If there are large variations in process delay time, WMAX will have to be respecified by the operator.

6.4.7 Nonlinear Processes

In section 6.3.3, the tracking ability of the EXACT's self-tuner was demonstrated for a change in operating conditions using the temperature controlled process. Because this process is so nonlinear the change was implemented using a series of -5% set point changes in order to enable the tuner to adjust its PID parameters in response to changing process dynamics.

Figure 6.31 contains a run with the same change in operating point implemented using a single -30% set point change. The oscillatory response illustrates a drawback of the once-per-transient adaptation philosophy. Because the process gain is much higher at the 30% operating point, the controller gain obtained in the 60% region is not appropriate and the closed loop system becomes unstable. The self-tuner adjusted the PID parameters twice before the system stabilized.

The parameter trajectories shown in Figure 6.32 indicate

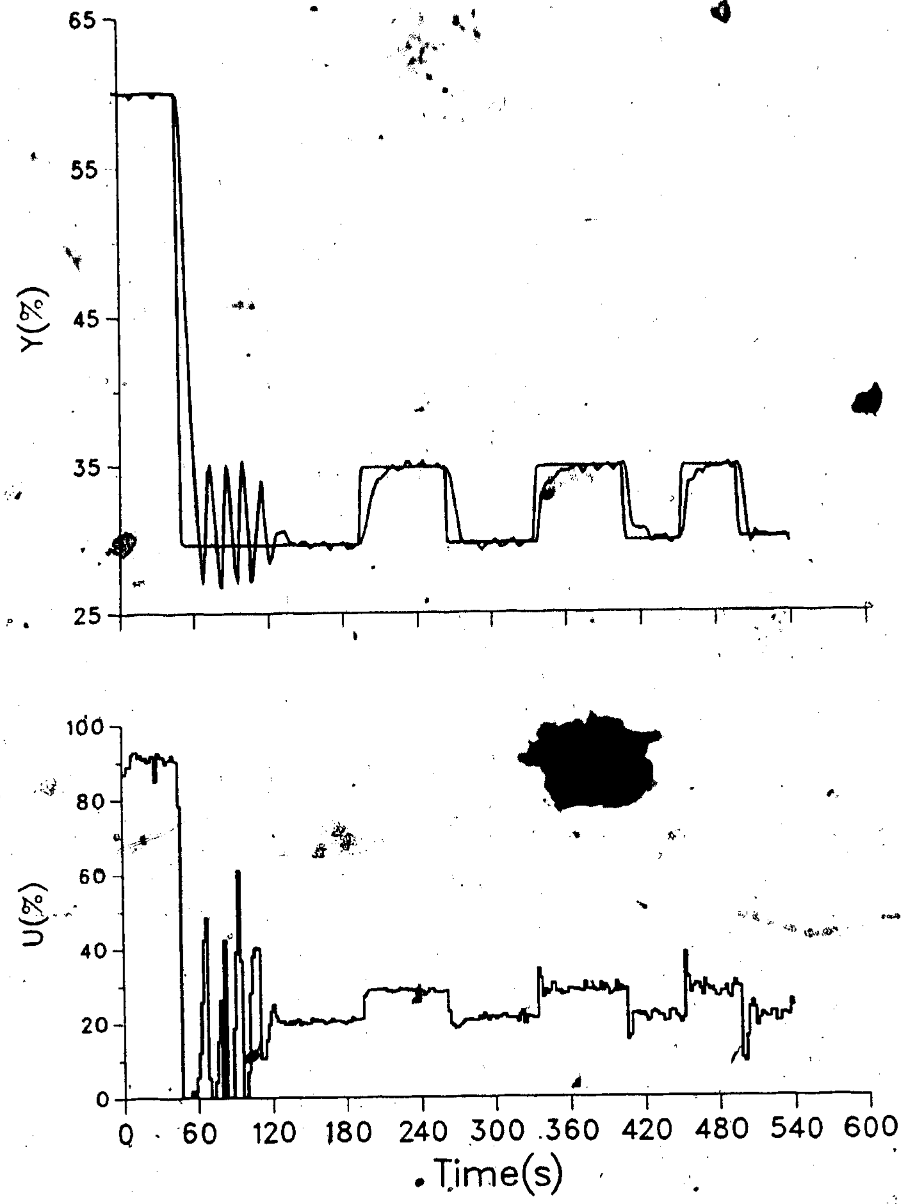


Fig. 6.31 Effect of Nonlinear Process Gain on Closed Loop Stability (CODGF.6 OVR=0.25 DMP=0.20 LAG=0.0 min.)

that the self-tuner "overreacts" in response to the oscillations in temperature. The proportional band initially at 16% is increased to 88%. Subsequent set point changes result in the proportional band being decreased to a value of 28%. Note the PI parameter trajectories for the run using the series of smaller set point changes. The smaller set point changes resulted in more conservative adjustments of PID parameters. The use of the nonlinear compensation option to overcome the problem of oscillatory control due to large set point changes and slow tracking is discussed in 6.5.4.

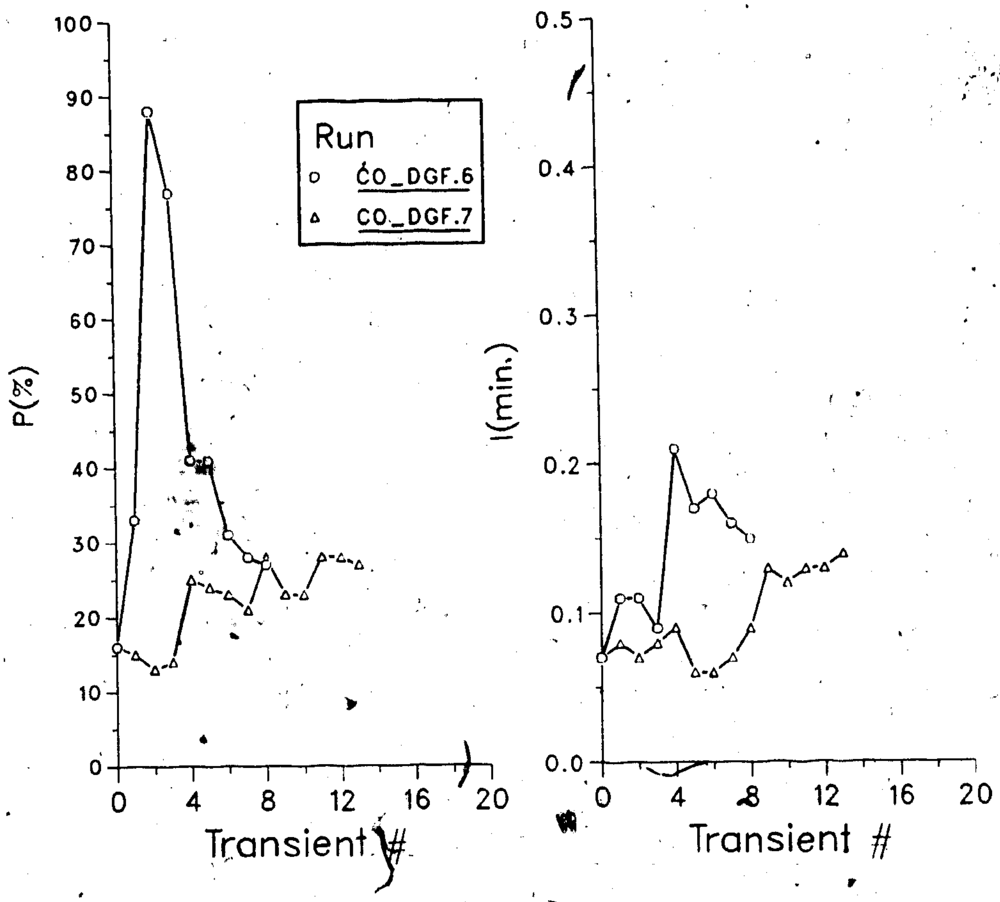


Fig. 6.32 Effects of Nonlinear Process on Parameter Estimates

6.5 Evaluation of Controller Features

Several features have been incorporated into the EXACT controller's design to improve its overall performance and range of applicability. Many of these features, [illegible] the functional description, were used to overcome some the problems discussed in section 6.4. It is these results that are presented below.

6.5.1 Digital Filtering of Process Measurements

Besides removal of measurement noise, the digital filtering option was used to prevent the gain wind-up problem discussed in section 6.4.1. The inclusion of the digital filter effectively increases the overall order of the closed loop system and thus allows for oscillatory closed loop responses with lower controller gains.

Figure 6.33 illustrates the effect of using a relatively large time constant (six seconds) on closed loop performance. A Pretune run was first performed using the following initial conditions:

LAG=0.03 min.	y(k)=24.6%
BUMP=-5.0%	u(k)=48.5%

with the following results:

P=120%	I=0.44 min.	D=0.05 min.
NB=1.3%	WMAX=1.47 min.	DFCT=0.68

For specified overshoot and damping limits of 0.5 and 0.3 a series of set point perturbations were then implemented in an attempt to get converged PID parameters and the desired closed loop response pattern. The particular series of set point changes was suspended when the controller parameters had attained values of:

P=4% I=0.15 min. D=0.04 min.

due to undesirable oscillations in controller output and tank level (within the 1.3% noise band).

Figure 6.33 illustrates performance when the digital filter constant was increased from 1.8 to 6.0 seconds. The closed loop response initially became unstable due to the added dynamics and the self-tuner quickly detuned PID values in order to stabilize the system. Several set point changes were then implemented to obtain the desired closed loop response.

Figure 6.33 also demonstrates one drawback of the EXACT controller's tuner. The tuner delivers the specified three peak response for a set point transient but it does not consider subsequent oscillations in tank level. Even though these oscillations are within the specified noise band, they are not acceptable.

6.5.2 Output Cycling Limits

As discussed in section 6.1, the specification of limits on magnitude of sustained oscillations in controller output can be used to automatically detune the EXACT controller. Such a feature is particularly necessary for adaptive

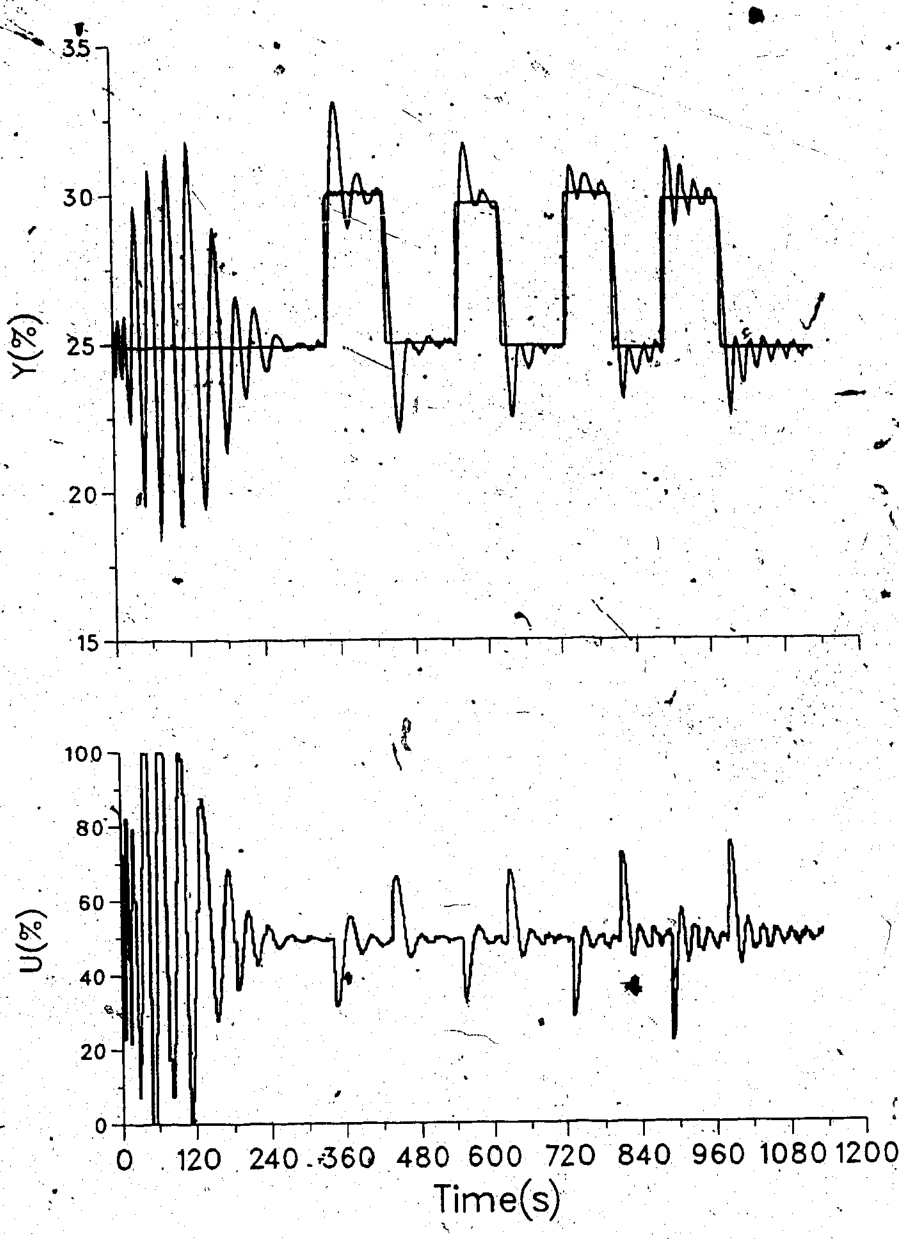


Fig. 6.33 Use of Digial Filtering to Prevent Gain Wind-Up (C_1_3.2; LAG=0.3→0.10 min. OVR=0.5 DMP=0.3)

controllers whose performance indices do not include variations in control signal when monitoring "performance". The gain wind-up and resulting large, high frequency oscillations in controller output due to large overshoot specifications illustrates the problem with such adaptive schemes.

Using the final PID parameters obtained from the series of set point transients in Figure 6.15 of section 6.4.1, the use of the output cycling limit feature is shown in Figure 6.34. Initial conditions are summarized as:

P=10% I=0.15 min. D=0.03 min. OVR=0.5 DMP=0.3

LIM=80.0%

The desired set point was held at 25% of measurement span for approximately eight minutes. The limit on output cycling was changed from its default value of 80% to 3%. Figure 6.34 illustrates the effect of automatic detuning on oscillations in control output (at times 90, 260 and 440s.). It also demonstrates how subsequent set point changes will again result in high controller gains and undesirable oscillations in controller output. The self-tuner and the cycling limit feature effectively "fight" against each other. In most practical applications, high frequency oscillations of any magnitude are undesirable due to wear on actuator devices. It would be appropriate to maintain a small value of LIM for general operation rather than the default value of 80%.

6.5.3 Clamping of Controller Parameters

When the Exact controller's self-tuner is turned on, the non-adapted PF, IF and DF constants are transferred to the P, I and D parameters. By specifying PF, IF and DF via the

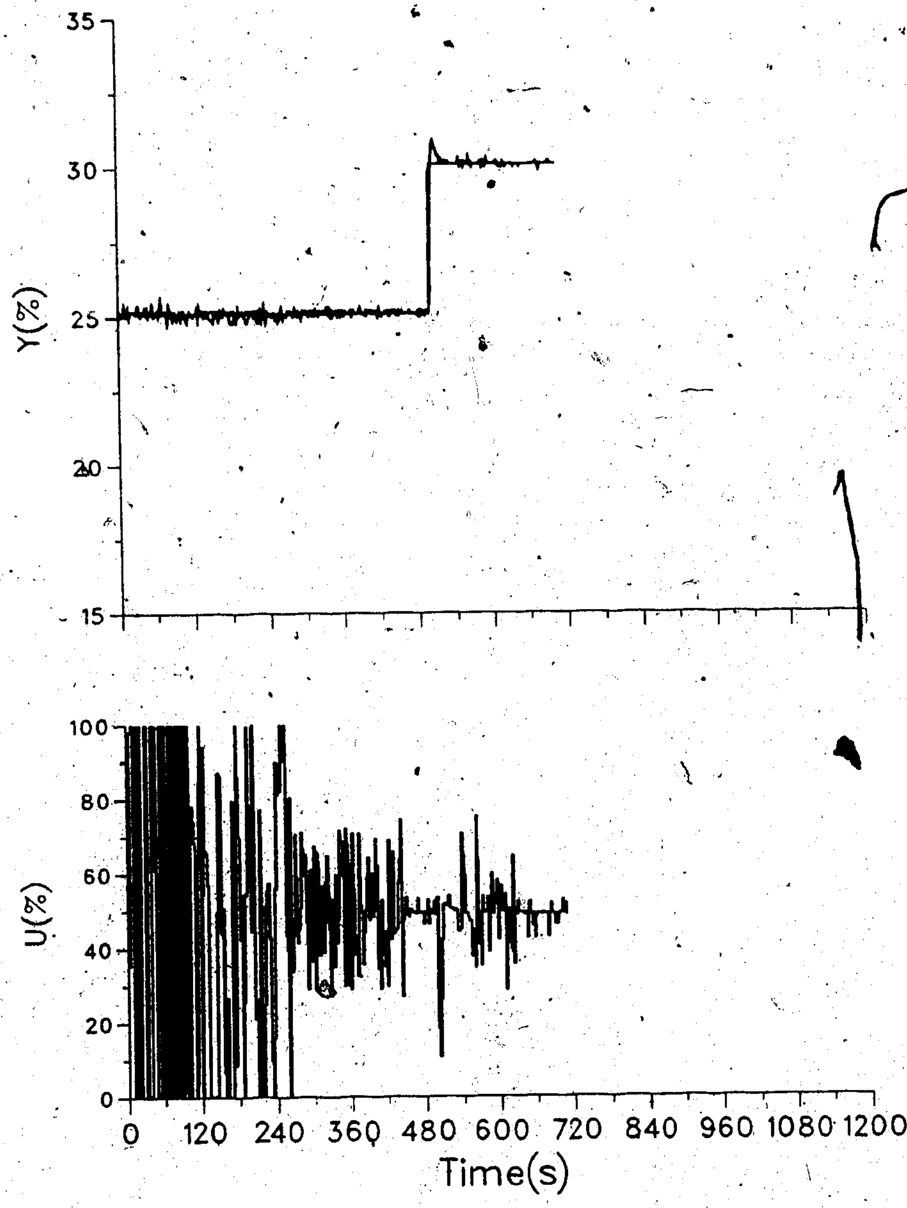


Fig. 6.34 Use of Output Cycling Limits to Prevent Large Oscillations in Controller Output (C_1_5.2).

hand held configurator or a supervisory computer, an operator can initialize the P,I and D values simply by switching the self-tuner OFF and then ON.

Figure 6.35 demonstrates how clamping can be used to prevent the gain wind-up problem discussed in section 6.4.1. The same initial conditions were used as in Figure 6.15 (section 6.4.1) but CLM was set to 4.0, thus limiting the possible range of adapted proportional band (with PF=30%) to:

$$7.5\% < P < 120\%$$

With the specified overshoot and damping ratios of 0.5 and 0.3, the controller's proportional band was quickly decreased by the self-tuner following several set point changes. After three transients, P was 8% and subsequent transients resulted in no further decreases due to clamping. This effectively prevents the gain wind-up problem. The operator must specify the constraints on proportional band to use this feature.

6.5.4 Use of Nonlinear Compensation

The performance of the nonlinear compensation feature is illustrated in Figure 6.37 for a large set point change to an operating point with a higher static process gain. The temperature controlled process was used with the autotransformer at 50% output. Steady state I/O data shown in Figure 6.36 was entered into the Exact's nonlinear table.

With compensation the initial transient was less oscillatory but subsequent tuning tranients did not dampen out, likely due to an inappropriate value of WMAX. The use

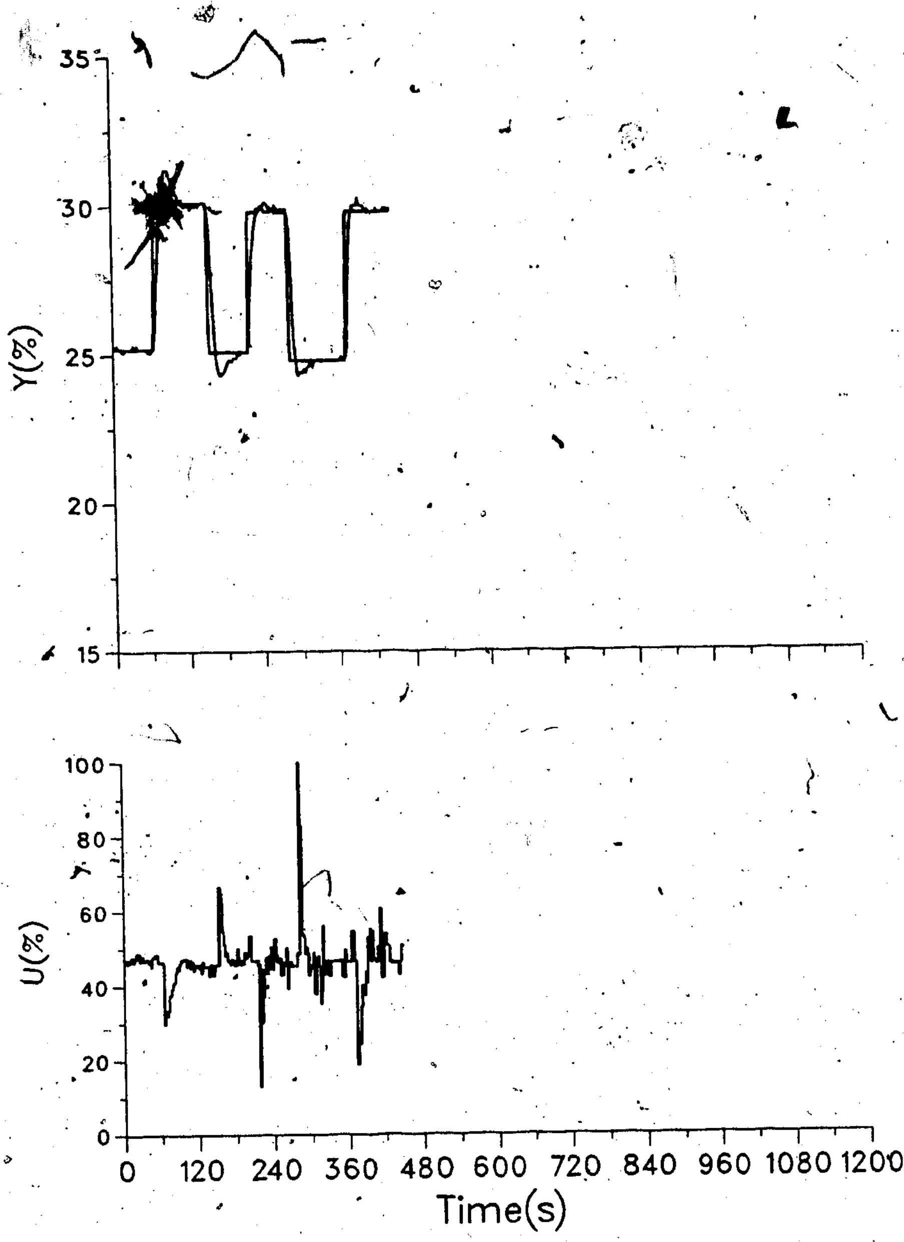


Fig. 6.35 Clamping of PID Parameters to Prevent the Gain Wind-Up Problem

of steady state I/O data for nonlinear gain compensation does not compensate for changes in process dynamics.

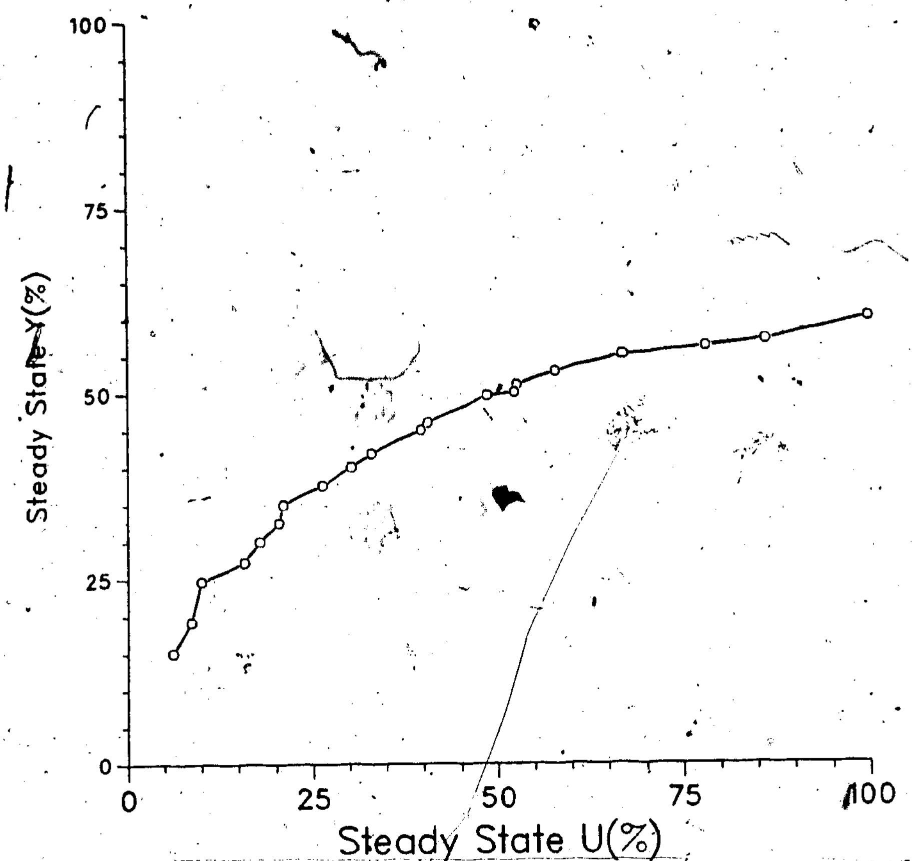


Fig. 6.36 Twenty Point Characterization for Temperature Controlled Process (50% transformer output).

6.5.5 Directionally Varying Derivative Action

One feature not discussed in section 6.1 was the implementation of directionally varying derivative action. The EXACT controller allows the operator to force the PID control law to use derivative action when it: decreases the process measurement (DER=-1), increases the process measurement (DER=1) or increases/decreases the process measurement (normal derivative action) (DER=0).

The feature is useful for processes exhibiting asymmetric

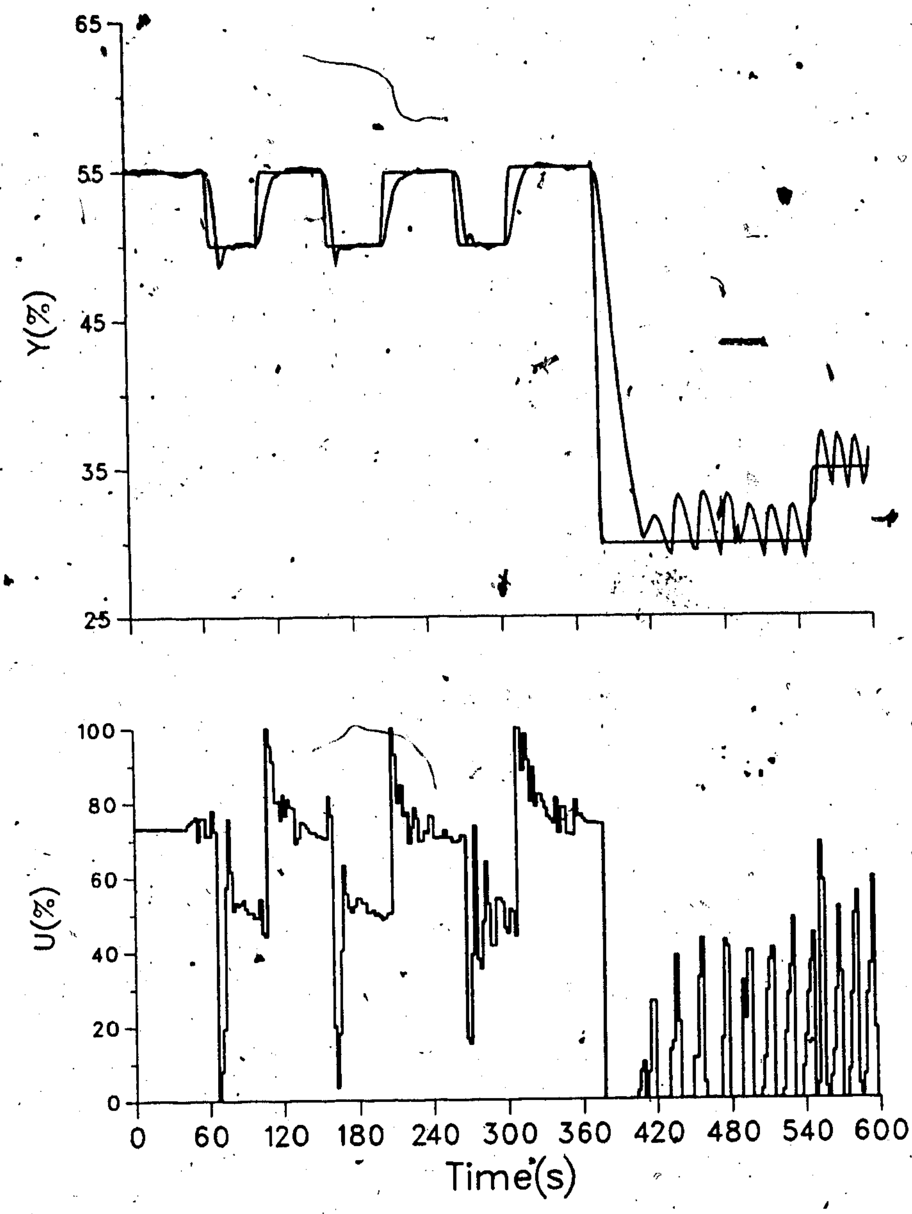


Fig. 6.37 Nonlinear Compensation Effect on Closed Loop Performance

dynamics. The first order liquid level controlled process was used for demonstration of the option. Specified overshoot and damping were 0.30 and 0.20, respectively. A digital filter constant of 0.10 minutes was used. PID parameters were initialized to P=34%, I=0.53 minutes and D=0.07 minutes. Figure 6.38 shows a series of set point changes while DER=+1 and Figure 6.39 for DER=-1. For the first series of set point perturbations the closed loop responses are much more symmetric with the derivative action being appropriately used. Figure 6.40 compares the adapted P and I parameters for both cases as a function [illegible]ing transient number. In the first case, DER=-1, the user specified limits are closely matched and controller parameters were approximately converged. When DER=+1 the effect of asymmetric process dynamics is even more pronounced with damped responses for positive set point changes. The adapted PI parameters varied by large magnitudes resulting in poor performance. It was not intuitively easy to choose the appropriate value for DER.

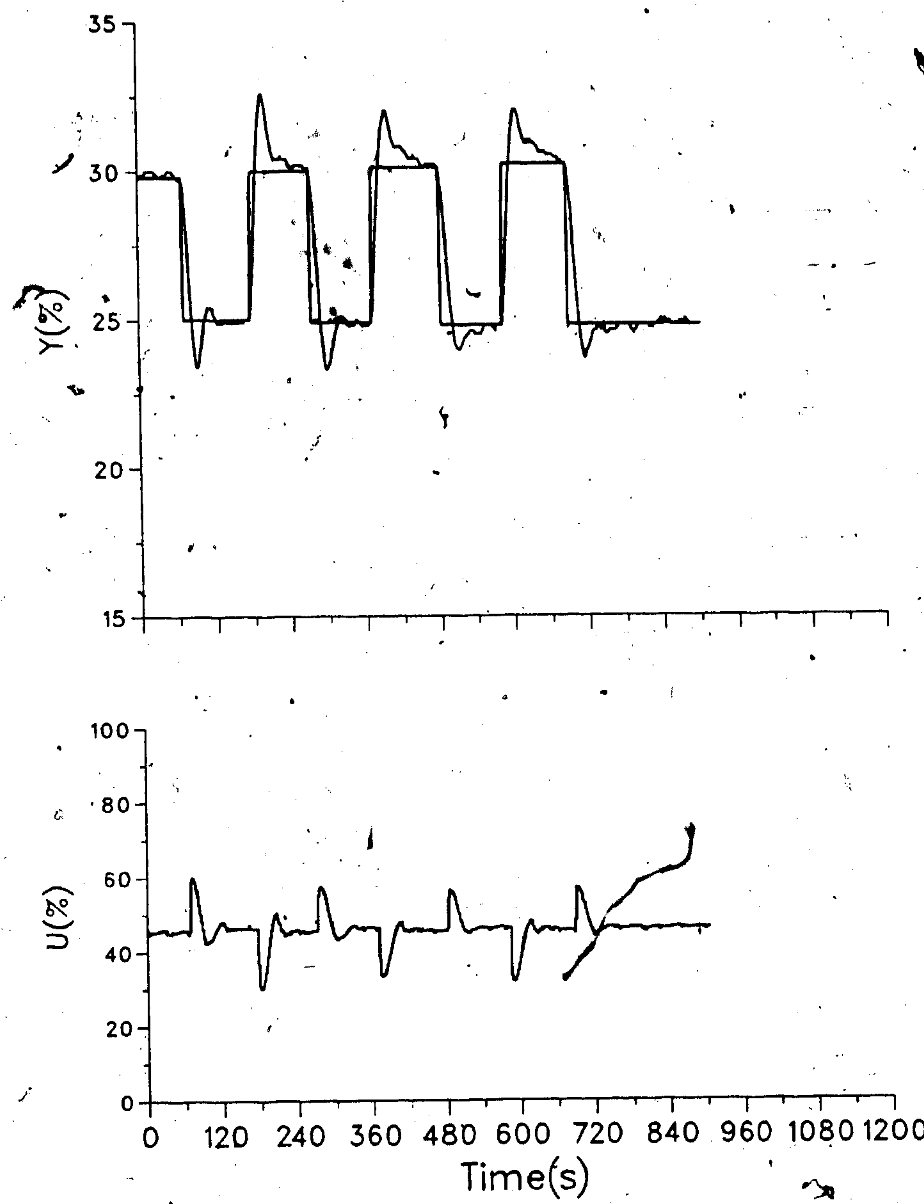


Fig. 6.38 Series of Set Point Perturbations Using DER=-1

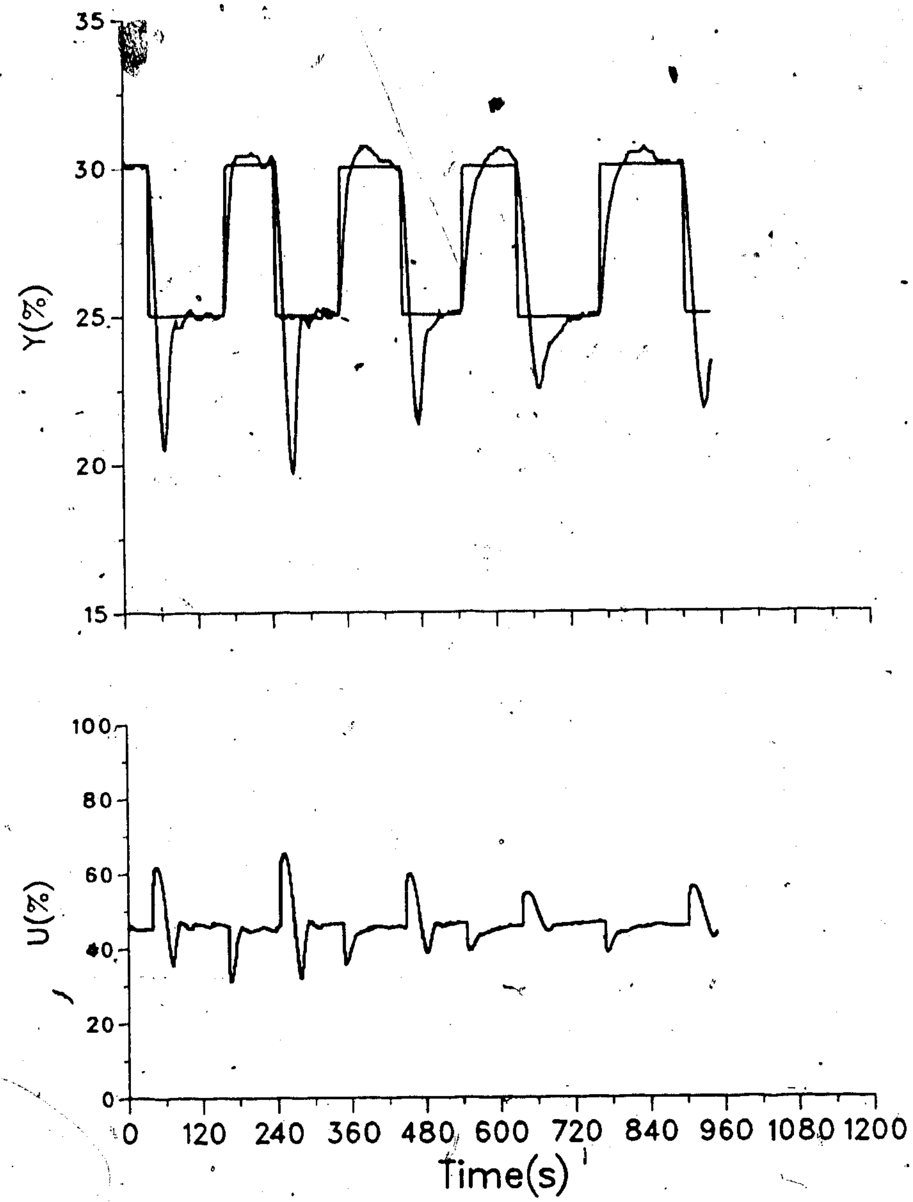


Fig. 6.39 Series of Set Point Perturbations Using DER=+1

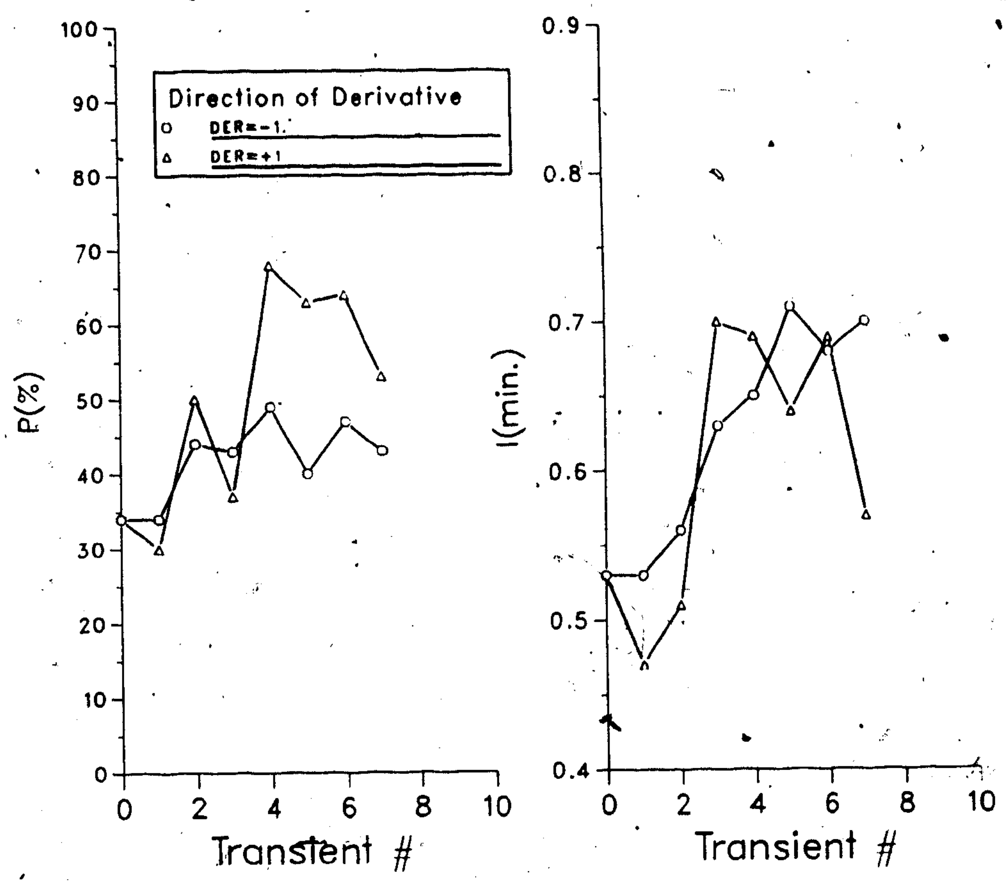


Fig. 6.40 Effect of Directionally Varying Derivative Action on Closed Loop Performance

6.5.6 Summary of Features and Performance - EXACT Controller

1. The Exact controller uses an error driven PID control law. Derivative action is filtered to prevent derivative kick and sensitivity to measurement noise.
2. The Exact's self-tuner automatically adjusts PID parameters following closed loop responses to set point or load disturbances. The pattern of the response is compared with a user specified pattern characterized by desired overshoot and damping.
3. The Exact controller requires five to ten set point or load disturbances to give converged overshoot and damping. The controller demonstrated its ability to adjust PID in response to changes in process dynamics or changes in specified overshoot and damping (see section 6.3).
4. For self-tuning operation there are five required inputs (see sections 6.1.4 and 6.2). If initial estimates are not available, an open loop PRETUNE test gives excellent initial values that result in damped responses for set point changes. With these initial values, user specified overshoot and damping is attained in approximately five tuning transients.
5. The performance criteria, overshoot and damping are not

suitable for overdamped processes. Overshoot specifications result in large controller gain and unacceptable variations in controller output (see section 6.4.1).

6. The tuner's once-per-transient adaptation approach can not track large changes in process dynamics given one tuning transient (see section 6.4.7). For highly nonlinear processes, a large change in operating point should be implemented using a series of smaller set point changes so that the self-tuner has more tuning transients to adjust PID parameters.
7. The Exact's tuner will not adjust PID parameters in response to slowly drifting disturbances or slowly changing process dynamics. If the change in process dynamics is significant, it may be necessary to perform a PRETUNE (section 6.4.4).
8. With no dead time compensation, the Exact controller is not suited for processes with large transport delay. Closed loop control will be too oscillatory with long periods of oscillation. If transport delays are time varying, WMAX may have to be respecified by the operator (section 6.4.6).
9. The Exact's tuner is not sensitive to the effects of high frequency measurement noise (section 6.4.3). PID parameters will not be adjusted even if noise levels exceed the user specified noise band, NB%.
10. The Exact controller is not suited for processes that

can not be forced to follow a second order underdamped type of closed loop response.

11. For processes with asymmetric dynamics the Exact's once-per-transient tuning philosophy can not give converged performance using up/down set point perturbations (section 6.4.2).
12. The use of a nonlinear process gain compensation option reduced the number of transients required for convergence of overshoot and damping. It also prevented large oscillations in process outputs following large set point changes into high process gain regions (section 6.5.4).
13. A digital filter option is available for use as a low pass filter to remove measurement noise. This option was used to allow overshoot specifications for overdamped processes (section 6.5.1) by introducing additional dynamics into the closed loop.
14. If controller outputs oscillate at high frequency for over three minutes, a user specified limit on the magnitude of those oscillations will force the tuner to adjust controller gain so that oscillations will be within that limit.
15. As a safeguard, high and low limits on proportional band can be specified. This clamping feature guarantees that adjusted band will lie within some specified range.
16. If the self-tuner gives PID parameters that result in poor performance, a back-up set of constants can be used

simply by turning the tuner off using a hardware switch on the control card.

17. The Exact controller is very easy to use. The status of self-tuning operation is easily monitored using a hand-held configurator. Information stored in the controller's memory is retrieved using meaningful, up to eleven (11) character mnemonics organized into four "tables". Most features required the specification of one or two paramters and a minimal amount of process control knowledge.

7. Turnbull Control System's Auto-Tuning Controller

The model 6355 Auto-Tuning controller is presented in this Chapter. The important features of the controller are presented in a functional description. This is followed by a presentation and discussion of closed loop performance evaluations in controlling the two pilot plant processes described in Chapter five. The experimental design follows that work performed using the EXACT controller in the previous Chapter.

7.1 Functional Description

Turnbull Control Systems 6355 Auto-Tuning Controller is a microprocessor based PID controller with a self-tuning algorithm that recommends adapted PID parameters in response to changing process dynamics. The 6355 features continual on-line identification of process dynamics and computation of "optimum" PI or PID controller parameters. However, new PID constants are not implemented until authorized by the operator.

7.1.1 Adaptation Mechanism and Tuning Philosophy

The TCS 6355 Auto-Tuning Controller uses an adaptive predictive estimation scheme in which a statistical model is identified based on measured I/O data. Given a user specified estimate of the process delay time, the 6355's tuner "forms" a model of the controlled plant's dynamic characteristics, based on measured I/O data, which is updated as more information is acquired.

Based on the estimated plant model, an unknown controller design procedure is then used to calculate <u>recommended</u> PID parameters as well as a recommended tuner sample time. While the 6355 itself uses a sampling rate of approximately 30 times a second, the self-tuner automatically selects an internal sampling rate for process model identification. Recommended PID parameters are specified as: proportional band (RP%), integral time (RI seconds or minutes) and derivative time (RD s./min.) and are not automatically used by the PID control law.

The 6355 algorithm calculates the value of a confidence factor (CF%) derived from how accurately the calculated model follows the true process variable. High values of the confidence factor indicate that the recommended PID parameters are closer to the "optimum" values. Using the confidence factor as a guideline, the operator can then explicitly force the controller to transfer the recommended constants: RP, RI, RD and RT into the set that are actually used by the controller: XP, TI, TD and TT. If the process time delay specified by the user is incorrect, the recommended parameters will not converge and the confidence factor will remain low due to poor modelling.

The 6355 controller has not been designed to operate as a self-tuning controller with continual automatic adjustment of the PID controller parameters. Instead the user is expected to use the adaptive mechanism for periodic retuning when operating conditions, desired set point or process dynamics change.

When the 6355's self-tuner has been performing model identification for long periods of time, the estimated parameters of this plant model change more slowly than at first. If rapid identification is required, the operator should reinitialize the plant model. By starting "fresh", old I/O data is discounted and only current data is used, resulting in faster adaptation. This is accomplished by using the "Retune" option. By setting the parameter "T?" equal to one, the current statistical model is discarded along with old I/O data. Only current data is then used for modelling, resulting in faster adaptation.

Turnbull Control Systems recommends the use of a user specified, superimposed set point perturbation signal during retuning that will provide adequate excitation and result in faster adaptation. The magnitude of these automatic changes in set point are selected by the operator (AD% of measurement span). These perturbations are not implemented until the operator selects either the "continue tuning" (T?=0) or "retune" (T?=1) options. Perturbations are suspended by either setting AD=0% or aborting tuning (T?=A). The reader should note that for plotting purposes, a numerical value of T?=-1 corresponds to T?=A. For example, in Figure 7.4, tuning was aborted at t=265s (T?=-1) and then the "continue tuning" option was started at t=415s. Without this perturbation signal, the 6355 will not be able to determine an accurate model or optimum PID value.

7.1.2 Required Inputs

The Auto-Tuning Controller contains a considerable number of adjustable parameters that are used for specifying alarm limits, signal conditioning, specification of engineering units on measured signal, and so on. These parameters are typical of many non-adaptive industrial PID controllers and provide some flexibility when configuring the hardware for a specific application.

For "auto-tuning" operation, there are four required parameters that must be specified so that the self-tuner can be used:

XP,TI,TD Initial values for PID parameters.

DT Process Delay Time.
The operator must specify the magnitude of the process delay time (assumed to be constant). This parameter must be reasonably accurate or the auto-tuner will not provide converged PID parameters.

When the auto-tuner is turned on, it will determine an appropriate tuner sample time for subsequent adaptation. This parameter is not user adjustable, but is required for self-tuning operation.

Like the EXACT controller, if the four required inputs described above are unknown, an "initial test" procedure can be used to determine starting values. The initial test is described in section 7.1.4.

7.1.3 Additional Features and Inputs

Because the Auto-Tuning Controller does not automatically implement updated PID parameters during self-tuning operation, no features or safeguards have been provided for limiting PID parameter esimates calculated by the self-tuner. The design implicitly assumes that the operator will make the correct decision about whether to update PID parameters guided by the given confidence factor.

Error Limit - *EL*

Turnbull has provided a feature that enables the operator to indirectly increase the controller's closed loop speed of response to step changes in set point. If, for a given set point change and current PID parameters, the controller output saturates at its upper or lower limit (HO or LO%) the error limit can be used to maintain saturation until the set point error falls within the specified value of EL%. The "error limit" parameter is used as a means of regulating the time that the controller output is held at saturation and hence can be (loosely speaking) used to control the amount of overshoot exhibited by a particular process. It is emphasized that the error limit parameter only becomes effective if the controller output saturates following a set point change.

Digital Filtering of Process Measurement - *IF*

The Auto-Tuning Controller's control law structure has the following continuous time representation:

$$OP=\frac{100}{XP}\cdot(ER + \frac{1}{TI}\int ERdt + TD\cdot\frac{\delta PV}{\delta t})$$

where. . . OP= output of controller

XP= proportional band.

ER= error or difference between setpoint and process variable.

TI= integral time constant.

TD= derivative time constant.

PV= process variable.

Note that derivative action acts on the process output and not on the error, thus preventing "derivative kick" on set point changes. To prevent undesirable feedback of noise, Turnbull has provided a first order exponential digital filter. The use of this option significantly reduces the variation in controller output for process measurements corrupted by noise. It should also improve the performance of the self-tuner since measured I/O data is better conditioned.

Identification Mode - *IM*

The operator can use the hexadecimal parameter, IM, to disable/enable the self-tuning of integral and derivative modes during auto-tuning operation.

7.1.4 Initial Test - Open Loop Identification Option

Turnbull Control Systems has provided the "initial test" option to be used when reliable starting values for required inputs are not available. With the process initially at steady state and in manual mode, a user specified square wave perturbation of u(k) is sent to the process. Based on the resulting dynamic response, estimates for PID parameters, process delay time and tuner sample time are calculated.

The magnitude of the ±OD% change in u(k) must be large enough to force the process measurement to deviate from steady state by at least 2% of span. A user specified maximum allowable deviation in process measurement, ID%, determines the period of the square wave signal.

Following completion of the initial test, the derived confidence factor can be displayed as a measure of reliability for the recommended parameters (See section 7.2 for an initial test run performed on the nonlinear process.).

7.2 Initial Test - Open Loop Identification

Using the nonlinear plant, a typical initial test and subsequent closed loop performance is illustrated in Figure 7.1. For this run, the deviation in controller output was OD=10% and the maximum allowable deviation in process output was ID=5%.

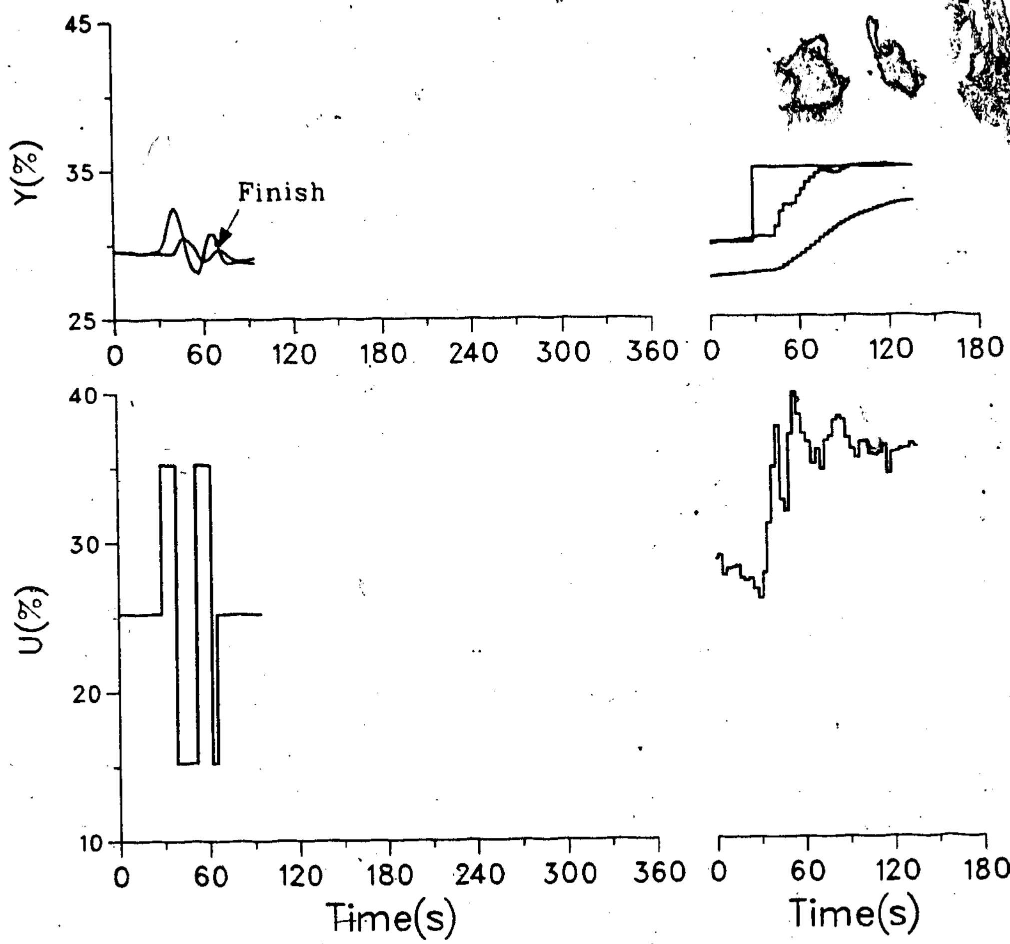


Fig. 7.1 Initial Test Using Nonlinear Process (Ti1_4.5)

The model output, plotted along with the process output did not track true process dynamics very well. Note that the process output curve has the larger oscillations. Upon completion of the test, the recommended parameters were

RP=23.5%, RI=0.52 minutes, RD=0.02 minutes and RT=0.02 minutes. A confidence factor of CF=40% was displayed upon completion of the test. Figure 7.1 shows that Turnbull's statistical model approach is slow to converge with a fairly low level of adaptability. This is illustrated by the transient response of the modelled output for the set point change and indicates why Turnbull recommends the "retune" option after long periods of modelling.

7.3 Closed Loop Performance

This section demonstrates the performance of the Auto-Tuning controller and the use of its "continue tuning" and "retune" options to obtain good PID parameters.

7.3.1 Typical Controller Performance

A closed loop transient response to a +5% step change in set point shown in Figure 7.2 demonstrates the 6355's typical control performance. To obtain this level of performance, an initial test was first performed (OD=10.0% ID=5.0%) about an operating point of 30% of measurement span to give RP=23.5%, RI=0.52 min., RD=0.07 and RT=0.02 with a confidence factor of 40%.

The controller was then switched to automatic mode and automatic set point perturbations were used to provide excitation for the "continue tuning" option. Set point perturbations were used for ≃5 minutes during which time the recommended parameters were transferred to the controller two times corresponding to confidence factors of ≃50 and ≃70%, respectively. Following the two PID updates, automatic tuning was disabled (T?=-1) and several step changes in set point were implemented to demonstrate resulting closed loop performance. The damped closed loop response of Figure 7.2 was typical of all runs and had a rise time of ≃ one minute.

7.3.2 Convergence of PID Parameters

Unlike the EXACT controller, user specified initial PID values are not modified by the "Auto-Tuning" controller when the tuner is "turned ON". Subsequent recommended PID parameters are not a function of the initial values. This is due to the controller design which calculates recommended values based on an estimate of the process model. Because the operator can not specify desired closed loop performance, convergence of the controller parameters was demonstrated for two operating points, using the nonlinear temperature process.

The first run (Tc1_5.5) used the "continue tuning" option with automatic set point perturbations (AD=2.0%) about an operating point of 45.0%. The second run (Tc1_6.5), used the "retune" option with AD=4.0% about an operating point of 30.0%. Both runs used initial PID parameters of XP=100.0%, TI=00.50, TD=00.00 and a tuner sample time, TT, of 00.05.

Figures 7.3 and 7.4 contain the graphical results for the "continue tuning" run. The transient responses of both the process and model output to the automatic set point

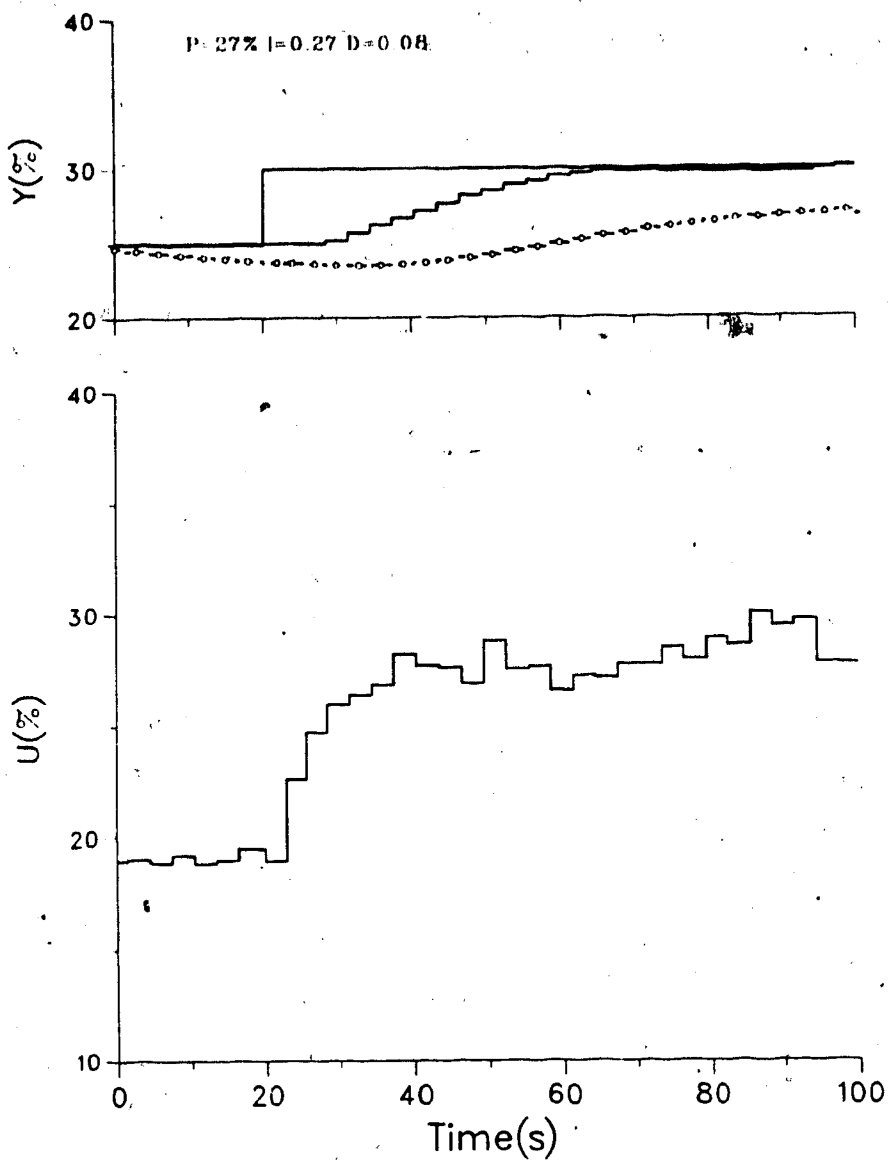


Fig. 7.2 Typical Performance for the Auto-Tuning Controller (Tc1''4.5)

perturbations are presented in Figure 7.3 along with the resulting control action. Figure 7.4 shows the transient responses of the confidence factor(CF), recommended band(RP), actual proportional band(XP) and the status of the self-tuner(T?). The actual band, XP, initially at 100% was updated five times as can be seen from Figure 7.4, when confidence factors reached values of 40,57,58,66 and 67%. During the initial part of the tuning period when T? was set to zero (t=80s), confidence factors remained low. With low controller gains and set point changes of fixed magnitude come low levels of excitation which, in turn, result in conservative recommended PID and low CF. If excitation levels remained low for too long, the 6355's tuner could lose its adaptability before giving good PID parameters. As the controller gain was increased due to updates by the operator, CF rose due to higher levels of excitation. From Figure 7.3, however, the combination of set point perturbations and lower proportional band resulted in unexpectedly vigorous control action. When the continue tuning option was suspended (T?=-1 at t=515s) closed loop control was slightly oscillatory due to interaction with the inner flow control loop. Process measurements oscillated about the set point with a deviation of ±0.1%. The final controller constants were: XP=4.5%, TI=0.14, TD=0.04 and TT=0.01.

Run Tc1_6.5 demonstrates convergence of PID parameters using the "retune" option at the 30.0% operating point (higher static process gain in this region) using 4.0% automatic perturbations in set point. Figures 7.5 and 7.6 illustrate the transient results for the "retune option" used. When the "retune option" was started (T?=1) at t=142s, the tuner reset the confidence factor to a value of zero.

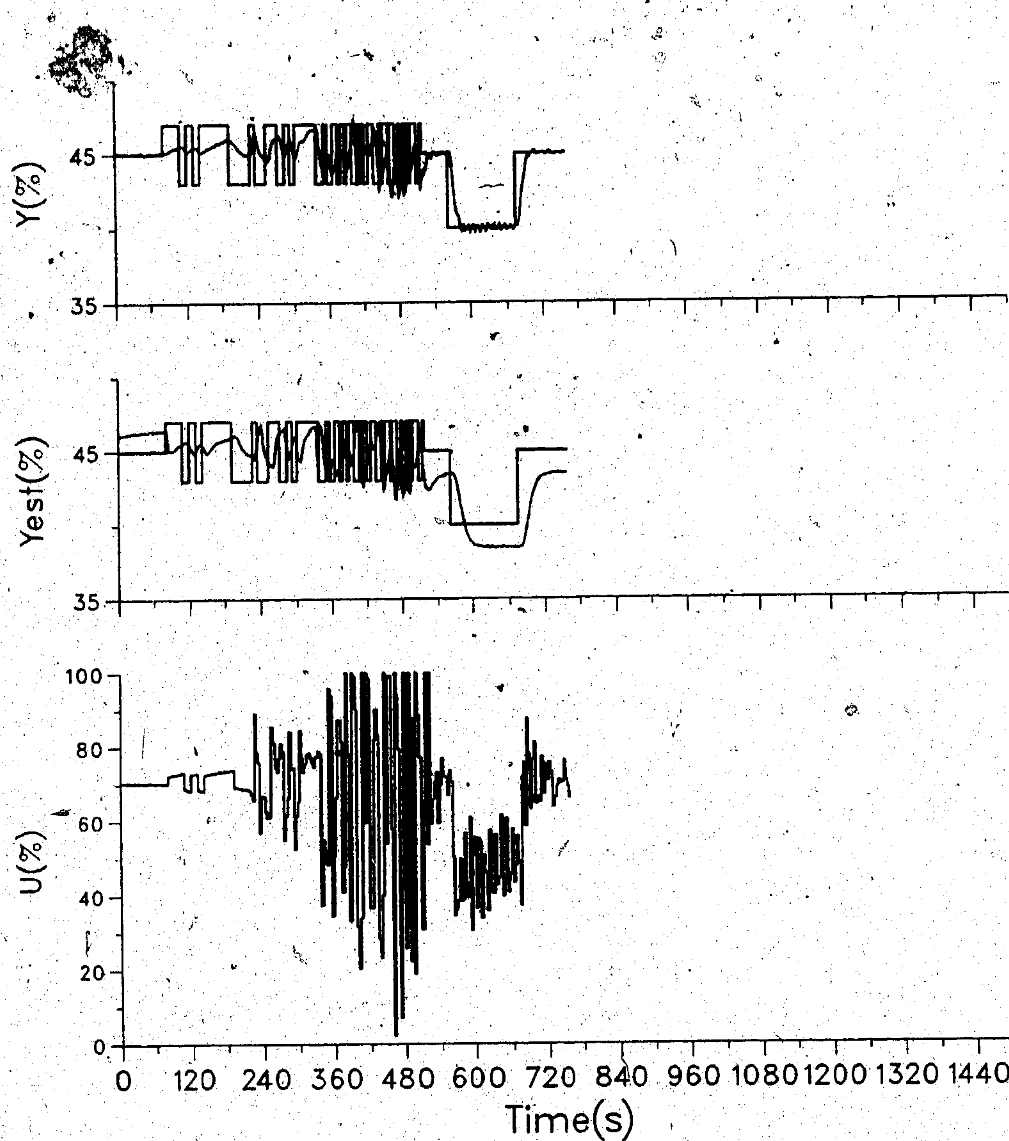


Fig. 7.3 Use of Automatic Set Point Perturbations to Give Converged PID Parameters (Continue Tuning Option - Tc1_5.5)

Initially, due to poor estimates of the process model, the recommended PID parameters were oscillatory and the estimated output of the model did not track the true process output very well. After ≃1 minute, the estimated process parameters began to stabilize and the confidence factor rose. At t=445s the recommended PID parameters were transferred to the controller to give PID values of

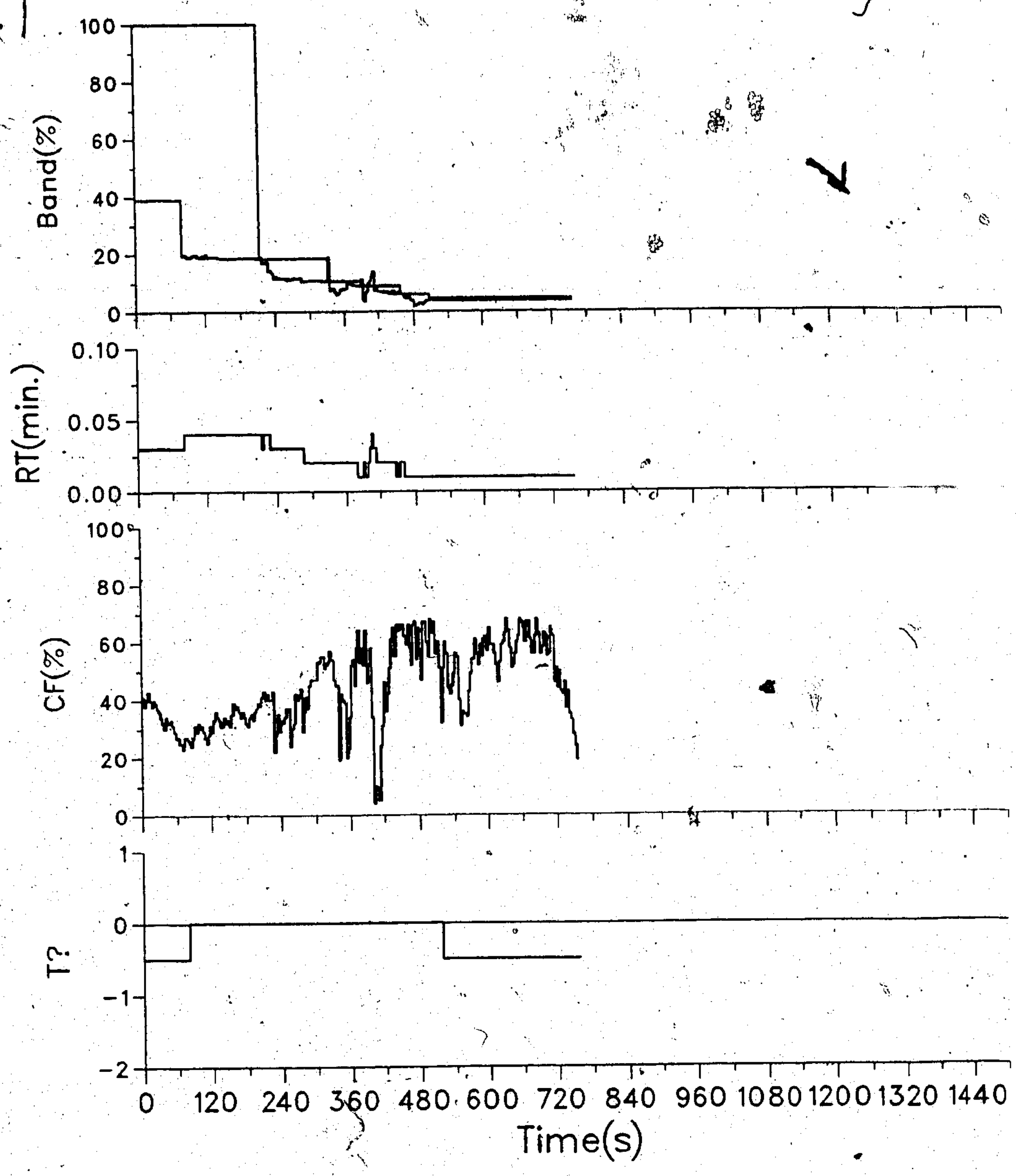


Fig. 7.4 Convergence PID Parameters Using the Continue Tuning Option (Tc1_5.5)

XP=16.1%, TI=00.21, TD=00.06 and TT=00.02. Subsequent set point perturbations resulted in little change in the recommended PID parameters so the tuner was turned OFF. The resulting closed loop control was stable with sustained oscillations of ±0.5%.

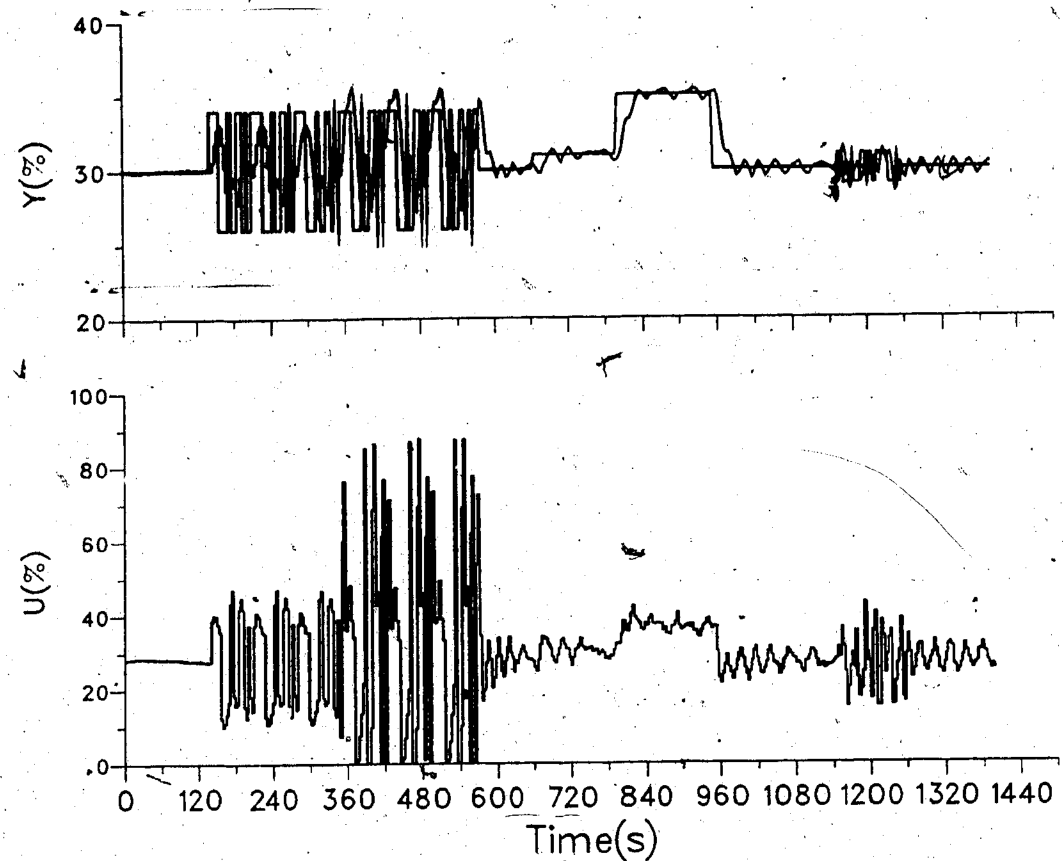


Fig. 7.5 Use of Automatic Set Point Perturbations to Give Converged PID Parameters (Retune Option - Tc1_6.5)

7.3.3 Tracking Ability of Self-Tuner

In order to evaluate the auto-tuning controller's ability to track large changes in process dynamics, a set point change from 45% to 30% span was implemented for two runs.

The first run (Tc1_7.5) used a single set point change of -15.0%. Figure 7.7 contains the transient response for the run which lasted approximately twenty minutes. The following initial parameters were specified (from Tc1_5.5):

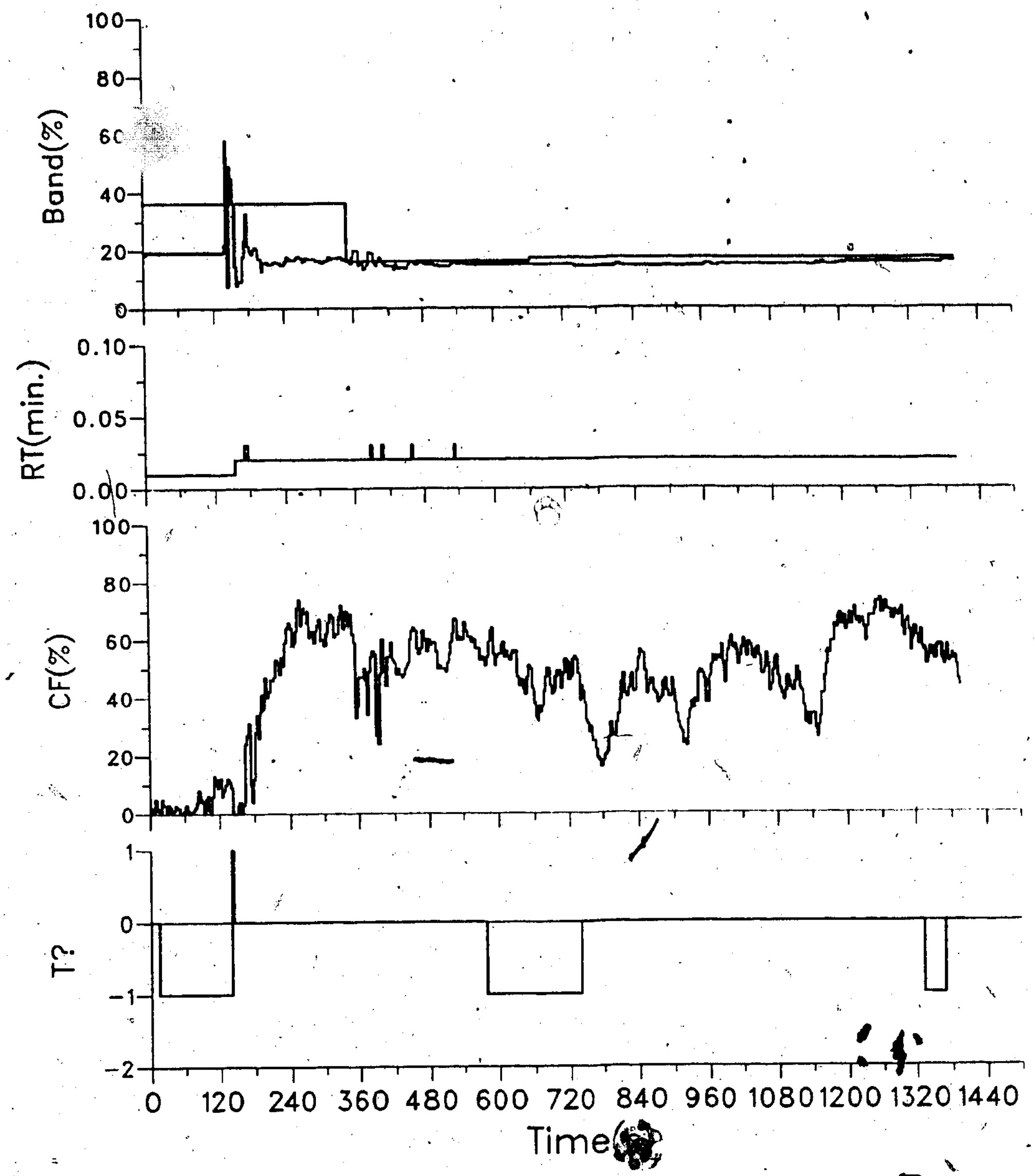


Fig. 7.6 Convergence of PID Parameters Using the Retune Option (Tcl 6.5)

proportional band, XP=6%, TI=0.18 minutes, TD=0.04 minutes, tuner sample time TT=0.01 and process delay time 0.02 minutes.

The "continue tuning" option was used initially during the run. The PID parameters were updated on-line by the

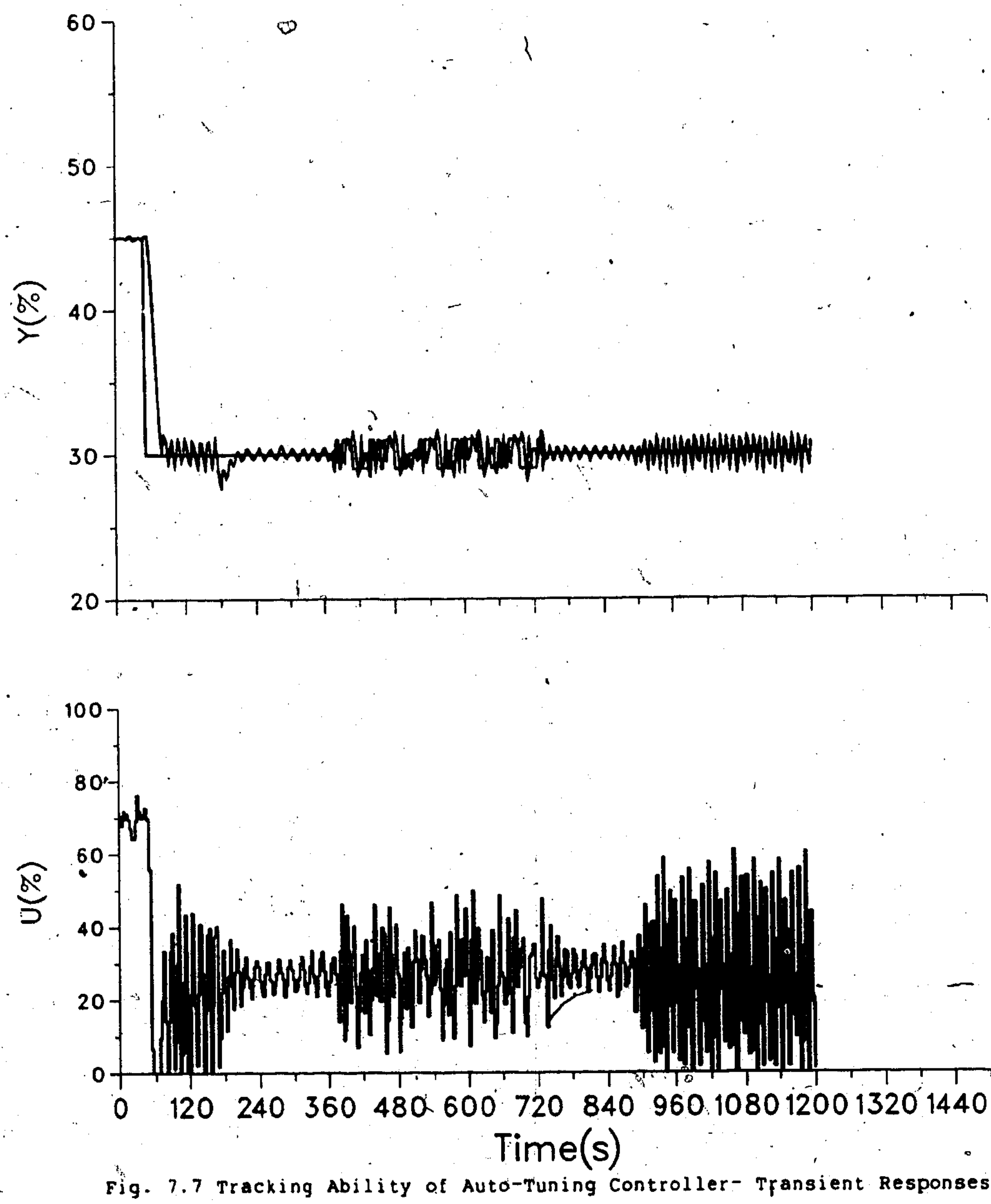


Fig. 7.7 Tracking Ability of Auto-Tuning Controller- Transient Responses with Single -15% Set Point Change (Tc1_7.5)

operator whenever the specified confidence factor reached an "acceptable" level (50-75%). Although this update procedure is somewhat subjective, it is typically how the controller would be used in industry.

Figure 7.8 graphically shows the performance of the 6355 controller's tuner, during the run. Variations in confidence factors, recommended proportional band and tuner status(T?) are plotted as functions of time. Recommended values for integral and derivative times are not shown since decisions to update PID parameters were based primarily on the confidence factor, CF, and the "trend" of the recommended proportional band, RP, itself.

Following the -15% set point change, the value of RP quickly increased once the process measurement reached the new set point of 30%, to a value of approximately 15.0% (t=170s). The confidence factor was at a level of ≃70% and the controller had to be detuned as evidenced by the sustained oscillations in u and y. Following the update of parameters (from XP=6 TI=.18 TD=.04 TT=.01 to XP=15 TI=.20 TD=.07 TT=.02), the confidence factor dropped to a value of 10% due to the change in tuner sample time, TT. It then climbed to ≃60% (t=275s) and parameters were again updated (14.2%/.18/.06/.02). Shortly after this, the self-tuner's confidence factors suddenly dropped to 5-10% and recommended constants took on "ridiculous" values (42%/51.2/0.00/2.07) due to the large changes in recommended tuner sample time.

At t=380s, a random set point perturbation (±1%) was

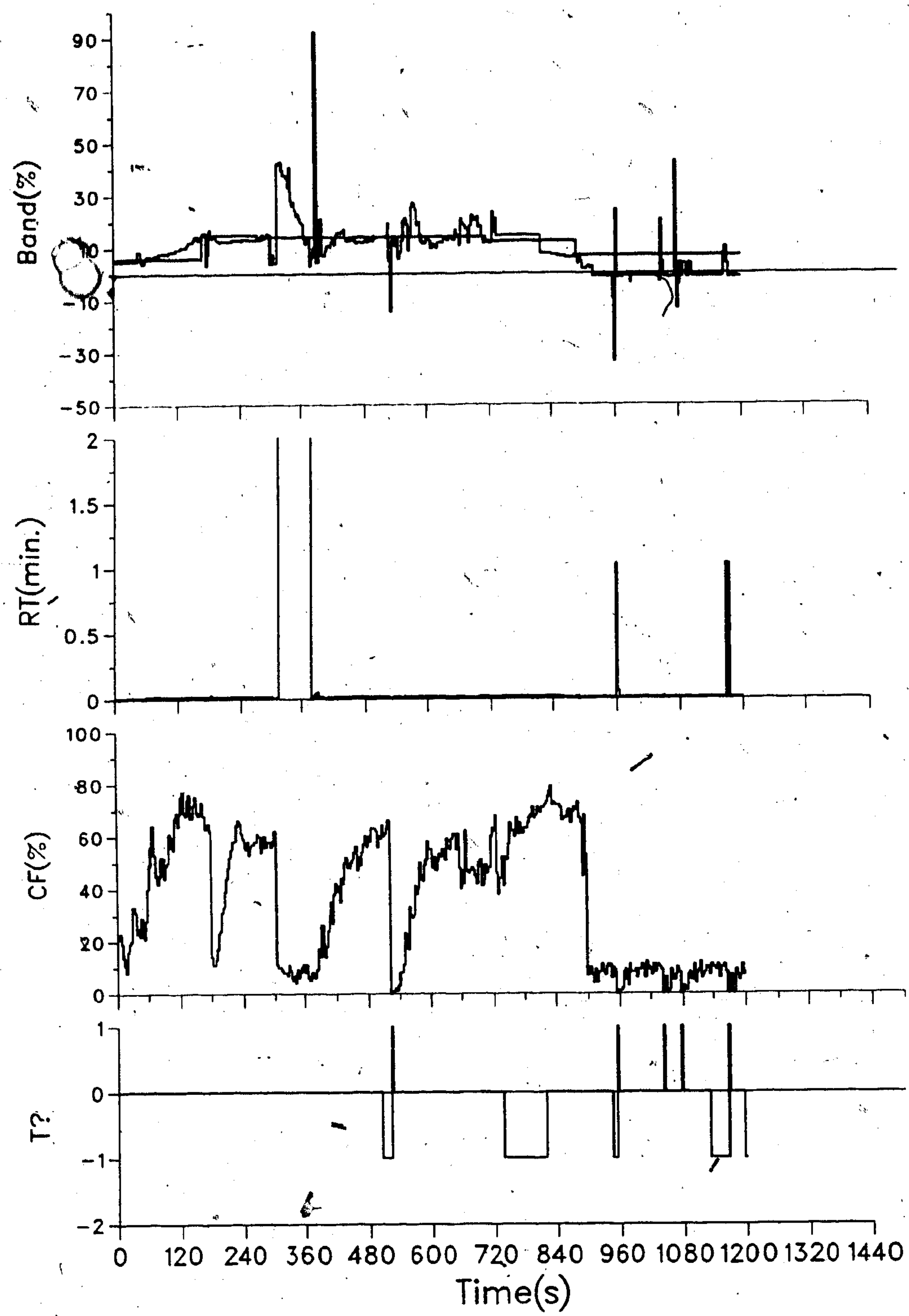


Fig. 7.8 Tracking Ability of Auto-Tuning Controller - Tuner Performance with Single -15% Set Point Change

introduced in order to provide excitation (through feedback) for the tuner. The recommended PID and RT were gradually adjusted to more "reasonable" values of 14.1/0.19/0.05/0.02 with a confidence factor of ≃60% by t=500s.

For demonstration purposes, at t=525s the "retune" option (T?=1) was specified so that the auto-tuner would discard the old model and "start fresh". Initially, CF drops to ≃zero and then rises to ≃60% (t=650s) at which time the PID parameters are updated (XP=14.0 TI=0.18 TD=0.05 TT=0.02). Parameters were again updated at t=710s (12.3/0.17/0.05/0.02). Tuning was suspended (T?=-1) at t=740s, and closed loop performance was oscillatory due to interaction with the inner flow control loop.

At t=820s the tuner was again turned on with no set point perturbations. Confidence factors remained high (≃70%) however and at t=890s the PID parameters were updated to values of 6.8%/0.15/0.04/0.01. Following this update, control became very oscillatory and the tuner "failed", giving negative values for RP and low confidence factors (≃5%).

For this particular run, the controller was not able to track the change in process dynamics fast enough to maintain smooth control. Once at the desired 30% operating point, the controller could not detune itself in order to give non-oscillatory control using either the "continue tuning" or "retune" options.

The results of this run indicate the amount of supervision and effort required for operation of the Auto-Tuning controller. Recommended PID values are very sensitive to the performance of the model and changes in tuner sample time. The results also show that the operator can not make safe update decisions on the basis of the confidence factor alone. When changes in recommended tuner sample time resulted in low value for RP (1%), which when updated to XP gave unstable control, the confidence factor had given no indication of trouble remaining at ≈40%.

In the second run, Tc1_8.5, the -15% change in operating point was implemented using a series of -5% step changes in set point. This procedure resulted in much better control performance compared to that of Tc1_7.5. Figure 7.9 contains the transient response of the system to the series of set point changes and the subsequent use of set point perturbations to improve the PID parameters about the operating point of 30%. Following this tuning period, several set point changes were implemented to demonstrate resulting closed loop performance. Figure 7.10 shows the behavior of CF,RP, and XP for the above run. The status of the tuner is plotted, as well.

Following each -5% step change in set point, the recommended PID parameters and the corresponding confidence factor were monitored. Recommended PID were explicitly transferred to XP, TI, TD and TT once during each transient in order to keep the controller parameters "up to date". Following the set point changes to 35% the process variable was somewhat oscillatory after the first update likely because the derivative constant was set to zero by the tuner. Following the set point change to 30%, and several PID updates, control was very sluggish. A subsequent set point change to 35% was aborted because the current PID parameters gave poor closed loop performance.

With such sluggish control action, the only practical way to improve the PID estimates is with automatic set point perturbations. At time t=570s, AD=1 was specified followed by an update at t=695s resulting in XP=45.1%, TI=0.40, TD=0.12 and TT=0.04. The set point perturbation parameter was then re-set to AD=2% in order to speed up parameter convergence. Several additional PID updates were implemented and the adjustment of PID parameters then suspended. Several operator specified step changes in set point were implemented to demonstrate the resulting closed loop performance.

With the highly nonlinear temperature controlled process, the "auto-tuning" controller demonstrated a limited ability to "track" changing process dynamics. Run Tc_1_7.5 indicates that large step changes in set point should be avoided (particularly when moving into regions with higher open loop static gains). Using guidelines for operation, attempts to get the self-tuner to detune the loop and stabilize control failed at the lower operating point of 30%.

The use of a series of smaller step changes in set point towards the desired operating point of 30% proved more

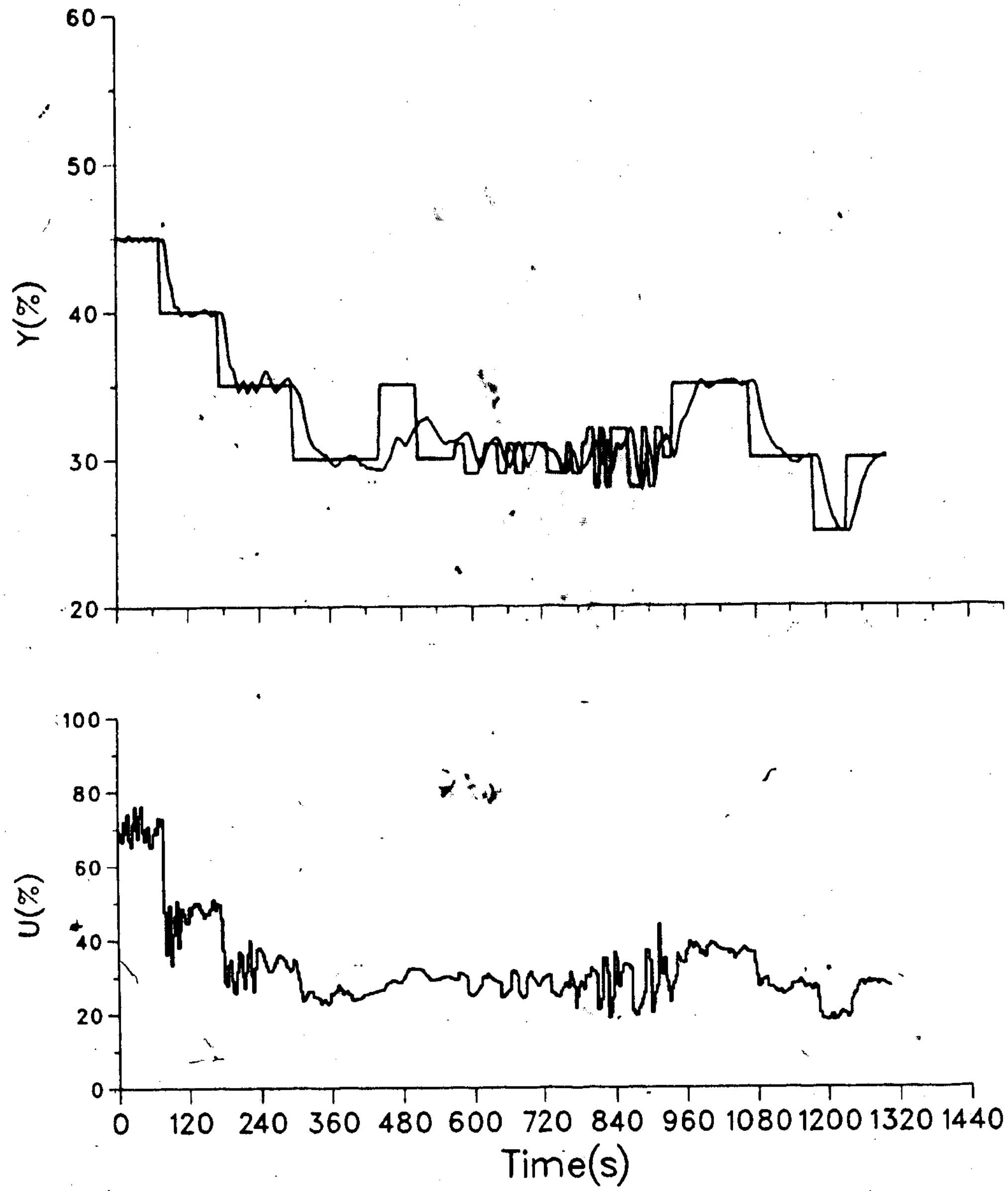


Fig. 7.9 Tracking Ability of Auto-Tuning Controller- Transient Responses with Series of -5% Set Point Changes (Tc1_8.5)

successful. Like the EXACT controller, it appeared to over detune, so much so, that control became very sluggish. The use of the automatic set point perturbation term AD was used to provide system excitation so that the tuner could adjust

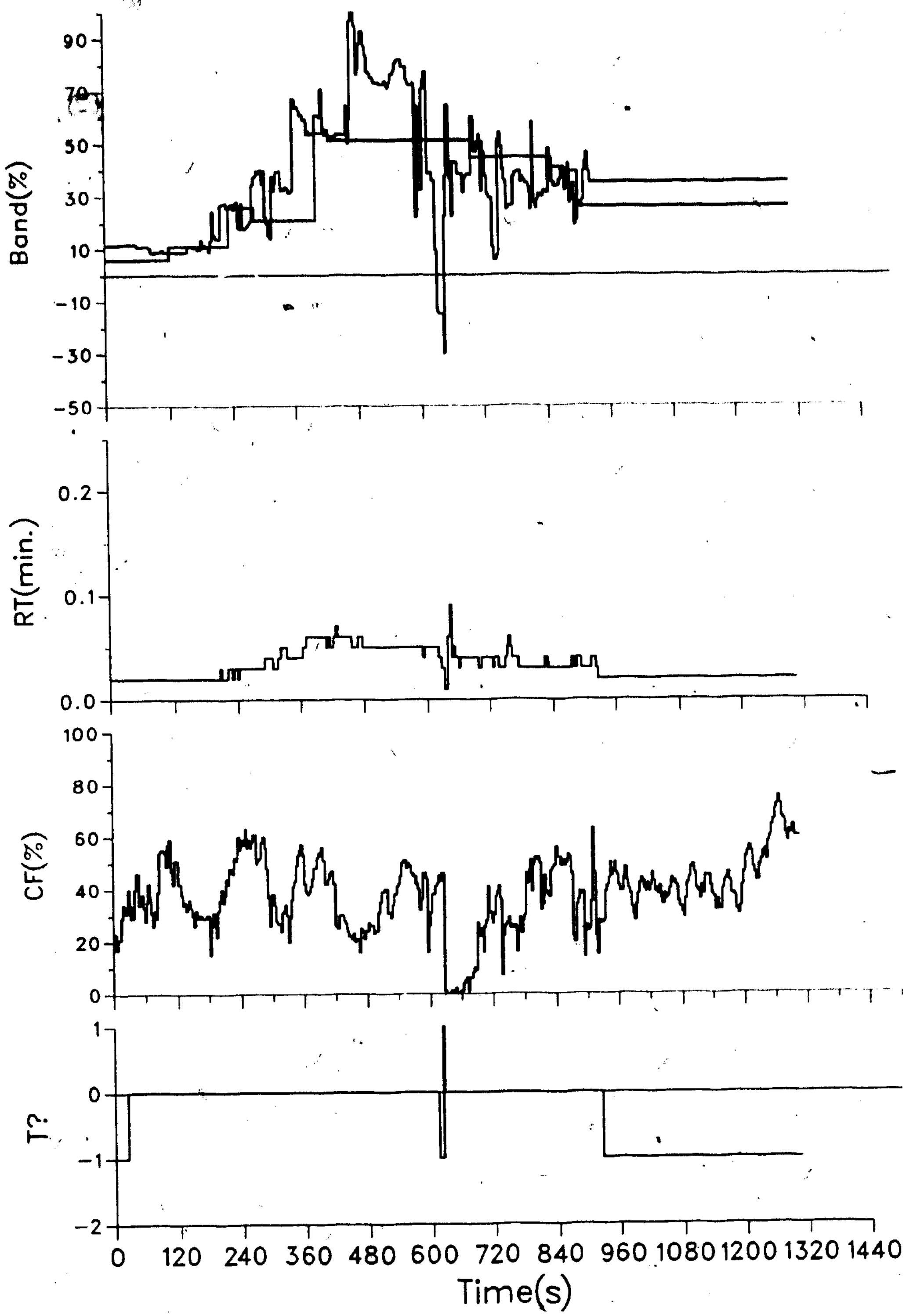


Fig. 7.10 Tracking Ability of Auto-Tuning Controller - Tuner Performance with Series of -5% Set Point Changes

the recommended PID parameters.

In addition, it was required that the self-tuner be "reinitialized" using the "retune" option of identification because the tuner had essentially lost its ability to adapt. The value of AD also had to be increased to 2.0% because a value of 1.0% gave slow adaptation. Following this considerable amount of supervision, the TCS controller gave good control in the 25-35% operating range.

7.3.4 Tailoring of Closed Loop Response - Error Limit Parameter

As discussed in section 7.1.3 the Error Limit parameter can be used to maintain the controller output at saturated levels for longer periods of time in order to increase the closed loop system's speed of response to set point changes. It can also be used, indirectly, to provide overshoot for set point transients.

The first order liquid level controlled process was used for this demonstration. Based on the closed loop performance with the nonlinear process it was decided that the first order process model would be easier to control and therefore demonstrate specific features. Two closed loop runs were performed each using the following initial conditions:

XP=-10%   TI=0.19 minutes   TD=0.05 minutes

IF=0.50s

y=30%   w=30%   u=45.9%

The first run (Tc2_5.2) using an error limit of 5% is shown in Figure 7.11. The first two set point changes were large enough that the controller output saturated at 0 and 100% respectively. The resulting overshoots were 11 and 18%, respectively. The subsequent ±5% step changes in set point illustrate that the Error Limit only takes effect when the controller output saturates due to large errors and/or high controller gains for a given process.

The second run (Tc2_6.2) using the default value for the Error Limit of 80%, gave corresponding overshoots of 7.7% and 7.9%, respectively. The difference in results is not large because the closed loop system responds very quickly after the controller output is allowed to come out of saturation. Nonetheless, it demonstrates that the Error Limit can be used to decrease rise times at the expense of larger overshoots.

Table 7.1 summarizes the effect of the Error Limit on closed loop performance.

TABLE 7.1 Effect of Error Limit on Closed Loop Performance (Liquid Level Control 30-50% Operating Region).

	Run	EL	Rise Time (s)	Overshoot (%)
Δw +20%	Tc2_5.2	5	25	11.0
	Tc2_6.2	80	28	7.7
-20%	Tc2_5.2	5	22	17.5

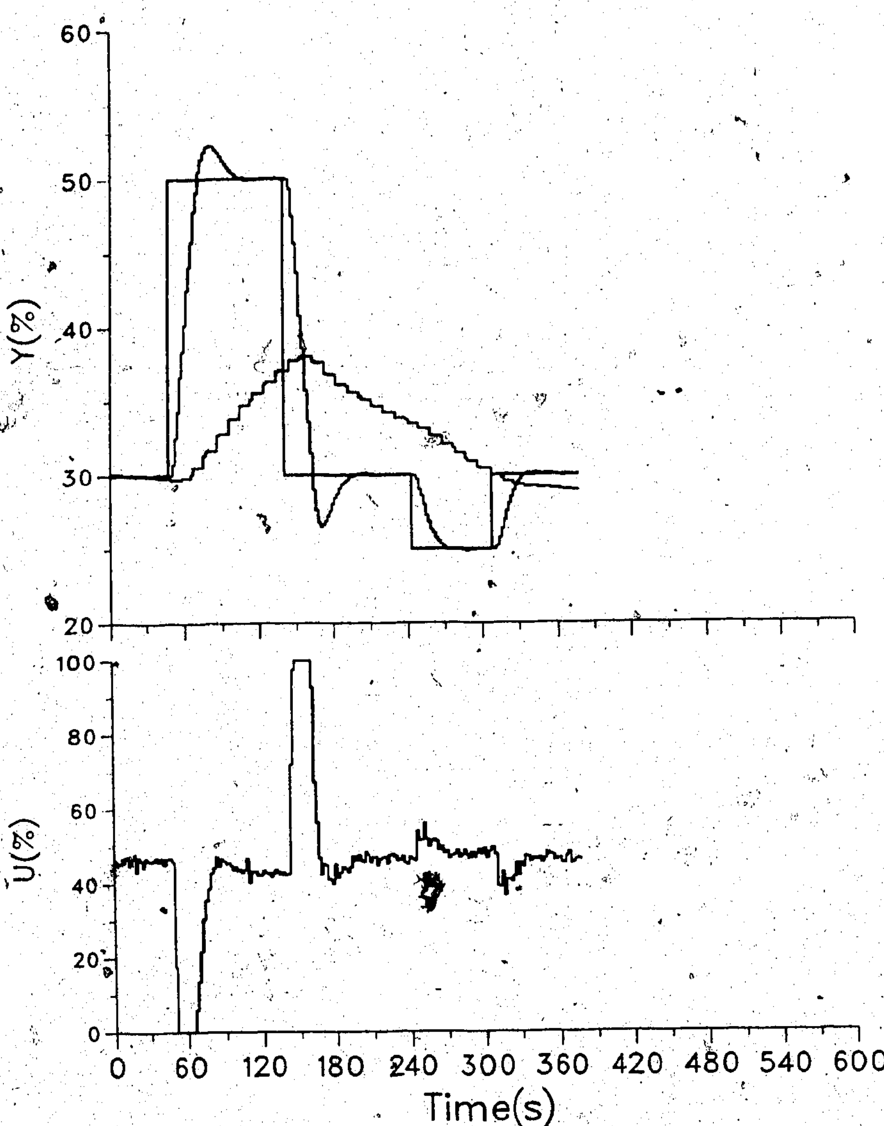


Fig. 7.11 Use of the Error Limit Parameter to Modify Closed Loop System Speed of Response - Tc2_5.2 EL=5%

Tc2_5.2	80	24	7.9

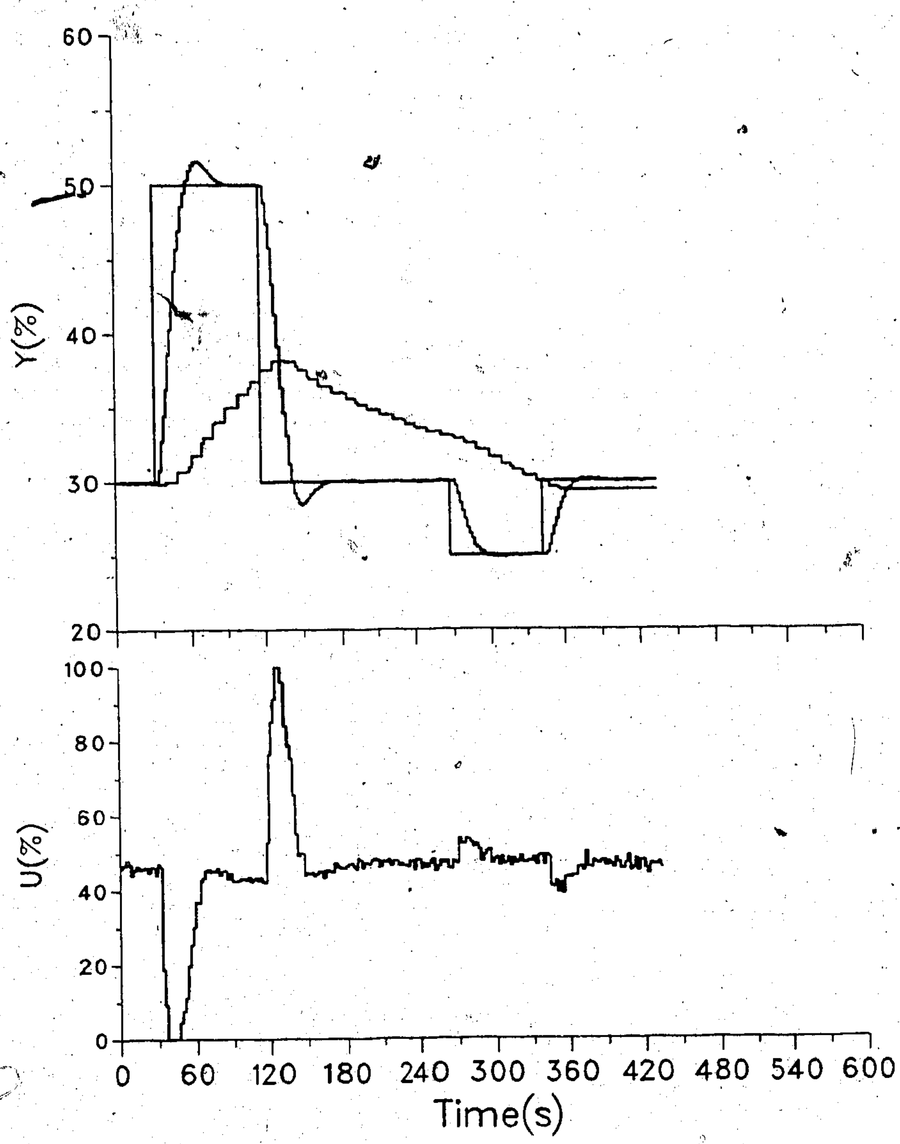


Fig. 7.12 Use of the Default Er [illegible] imit Parameter: EL=80% Tc2 6.2

7.4 Evaluation Under Selected Process Conditions

7.4.1 Overdamped Processes

Unlike the EXACT controller, the 6355 auto-tuning controller does not suffer from the gain wind up problem when controlling dominantly damped processes. Although the closed loop performance criteria are unknown, results indicate some form of "minimum variance" type control that gives damped closed loop response to set point changes.

Results below will show that for this simple process, the auto-tuning controller gives fast, almost damped transient responses to step changes in set point. Tuning was performed using the automatic set point perturbations. With measurement filtering, the controller output variance is acceptable at steady state (see section 7.4.2 for a discussion of Digital Filtering). The continue tuning option was started at t=50s and set point perturbation turned on (AD=2.0%) at t=82s. During the ≃4 minutes of random set point perturbations the PID parameters were updated four times. Following the suspension of self-tuning, several set point changes were implemented to demonstrate closed loop performance. Figures 7.13 and 7.14 on the following pages graphically illustrate the closed loop performance.

Figure 7.13 demonstrates one real drawback of the use of automatic set point perturbations. As PID parameters are updated (controller gain increases), the controller output becomes very oscillatory, likely due to the derivative action. Such control action might not be acceptable in industrial applications.

7.4.2 Measurement Noise

The "Auto-Tuning" controller's control law equation includes derivative action on process measurements. If process measurements become corrupted by random noise, the

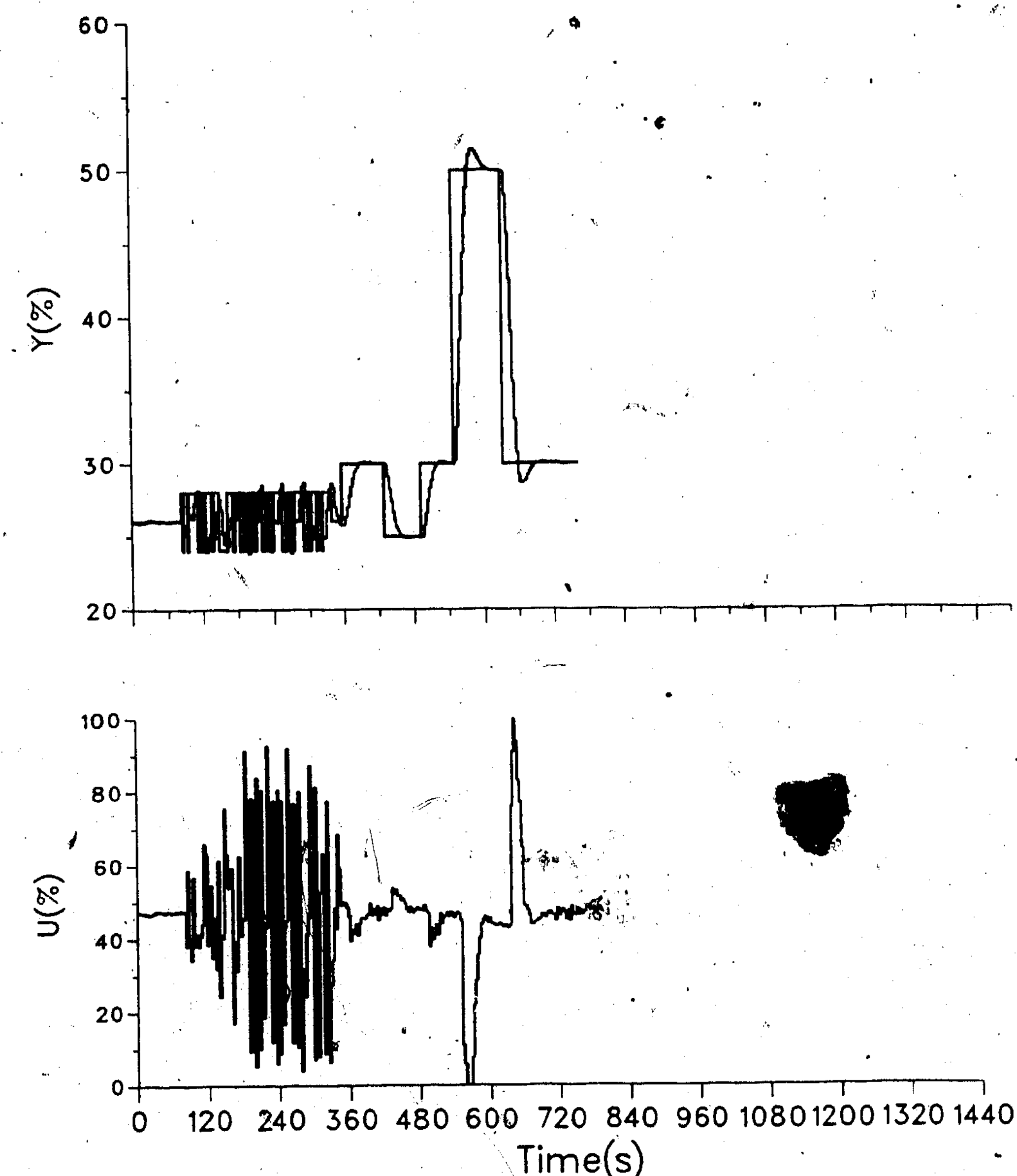


Fig. 3.13 Control Performance for the First Order Process - Transient Response
Tc2_3.2

feedback of these signals can result in corresponding variations in controller outputs.

These corrupted process measurements are also used within the self-tuning algorithm for modelling purposes. The feedback of measurement noise makes it more difficult for

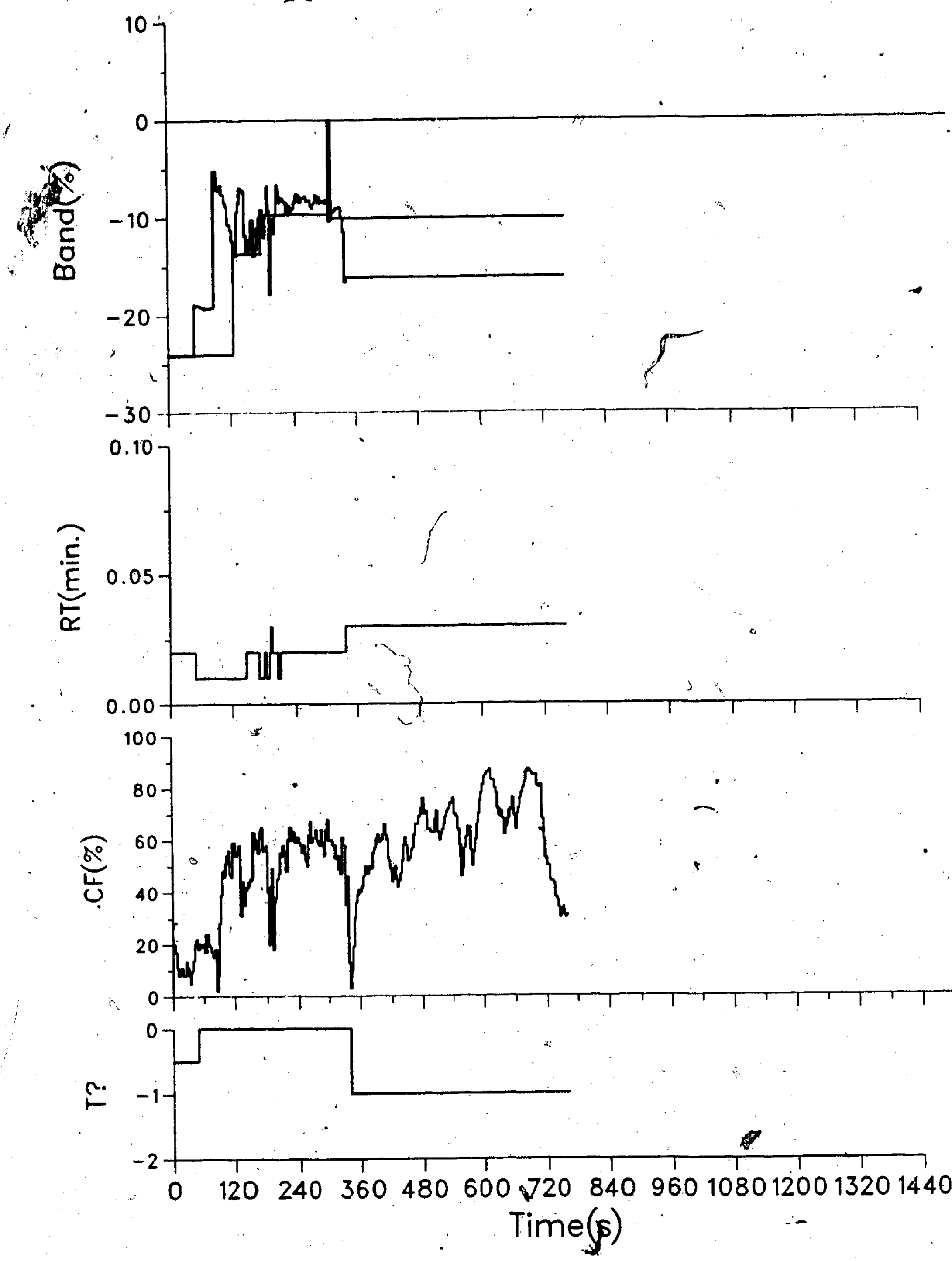


Fig. 7.14 Tuner Performance for a Dominantly Damped First Order Process - Tc2_3.2

the tuner to give converged PID parameters. The recommended

PID constants will tend to vary unnecessarily and due to modelling errors, the confidence factors will be lower. This will make it difficult to decide when to transfer recommended PID values. Figure 7.15 demonstrates how the measurement noise affects the variance in measured process outputs. The filter constant was set (t=550s) to IF=0.25s and the noise level to 63% at t=600s. The filter decreased the effect of noise on the measured process output.

Figure 7.15 demonstrates the effect of noise (12.5%) on control performance using the first order process. At t=270s the noise generator signal was increased to 12.5%. The controller output became more oscillatory due to the effects of derivative action on noisy measurements. Subsequent set point transients produced larger variance in controller output, as well.

At t=510s the filter constant was increased to IF=0.25s to demonstrate the effect of filtering on closed loop performance. Controller output became less oscillatory until filtering was disabled at t=595s. Noise was removed from the measurement signal at t=725s.

For the above run Figure 7.16 illustrates the effect of measurement noise on tuner performance. Recommended PID parameters and the corresponding confidence factor are plotted as a function of time. When the level of noise was increased to 12.5%, the recommended constants suddenly began to oscillate and the confidence factor decreased. When the measurement filter was "turned on", the PID parameters

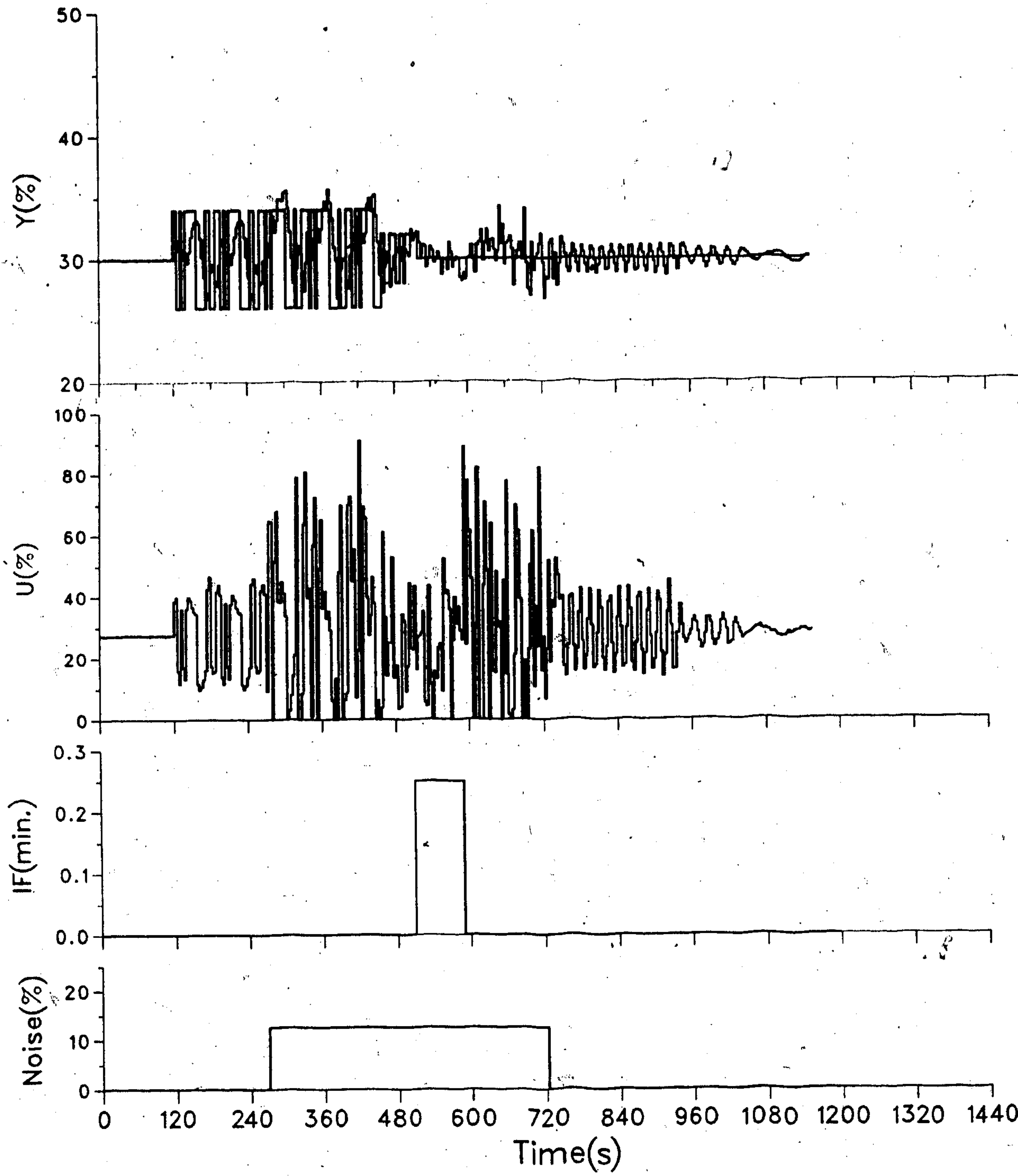


Fig. 7.15 Effects of Measurement Noise on Controller Performance

stabilized but confidence factors remained low. When filtering was disabled and following several PID updates, the tuner went unstable due to poor modelling (t=1080s).

The above results indicate that the 6355 Auto-Tuning controller is very sensitive to measurement noise both in terms of derivative action on process measurements and on tuner performance. When measurements are corrupted by noise the tuner gives wildly varying recommended PID parameters and low confidence factors. The tuner also loses its ability to adapt much sooner than when process measurements are not corrupted by hoise or process measurements are digitally filtered.

7.4.3 Slowly Drifting Disturbances

With the temperature controlled process the output of the autotransformer can be interpreted as a disturbance variable. When the autotransformer output to the heating coil is changed, the dynamic characteristics of the process change.

With its continual model identification, the Turnbull controller's ability to adapt in the presence of slowly varying dynamics was investigated. Following an experimental design similar to that work done with the EXACT controller, the following series of runs were performed about an operating point of 37.5%.

1. With the autotransformer at 50% output an Initial Test

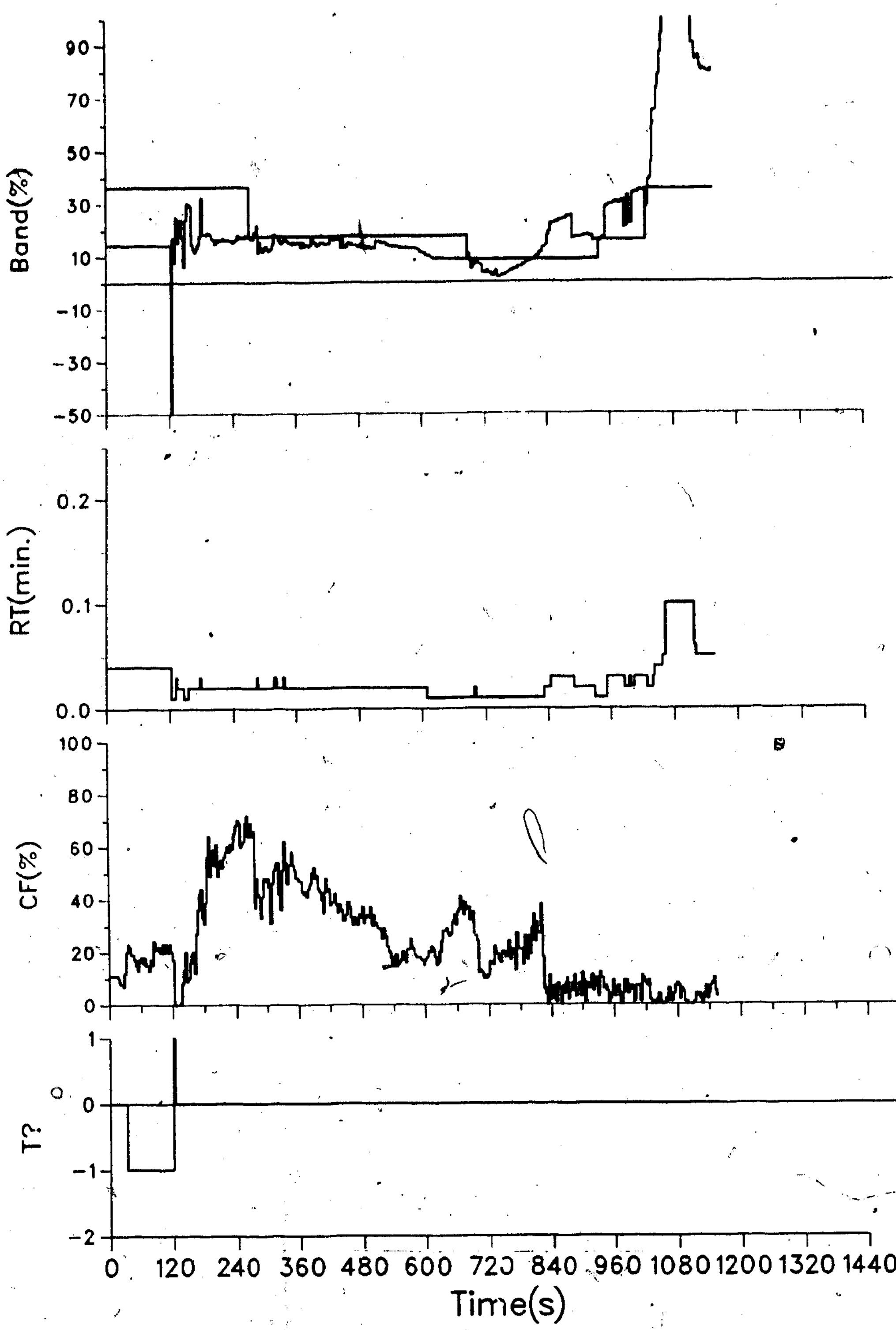


Fig. 7.16 Effect of Measurement Noise on Tuner Performance - Tc1_9.5

(OD=10% ID=5%) followed by "continued tuning" with automatic set point perturbation (AD=2%) was performed in order to obtain converged PID parameters (XP=10.7%, TI=0.20 TD=0.06 TT=0.02). Several set point transients were then used to demonstrate closed loop control performance (Runs Ti3_1.5/Tc3_1.5).

2. While running at steady state temperature corresponding to 37.5% of measurement span, the autotransformer output was changed from 50 to 70%, effectively increasing (asymptotically) the process gain from 0.6 to 2.0. The effect of this change on closed loop performance was observed and the ability of the tuner to readjust recommended PID parameters studied. Automatic set point perturbations were used to provide the necessary excitation for tuning.(Run Tc3_2.5)
3. Step 1 above was repeated with the autotransformer at 70%. (Runs Ti3_3.5/Tc3_3.5)

Table 7.2 summarizes the Initial Test results from steps 1 and 3 with different autotransformer outputs. As expected, the second run (transformer at 70%) gave larger proportional band since the static process gain is higher. The "converged" parameters are those values obtained following additional tuning.

Figure 7.17 shows the Initial Test for step 1 (Ti3_1.5) and Figure 7.18 the dynamic response of the system to automatic set point perturbations. Figure 7.18 also shows

TABLE 7.2 Initial Test and Continued Tuning Results about 37.5% Level for Configurations 1 and 2

Step	XP (%)	TI (min)	TD (min)	TT (min)	DT (min)	CF	Run
1	36.5	0.43	0.0	0.05	0.00	12	Ti315
3	51.0	0.34	0.0	0.03	0.03	33	Ti335
1	10.7	0.20	0.06	0.02		56	Tc315
2	48.6	0.36	0.00	0.03		47	Tc325
3	72.9	0.30	0.00	0.02		73	Tc335

how the tuner performed with plots of XP, RP, CF and the tuner status as well. When the tuner was turned off (T?=-1) at t=364s, closed loop control was oscillatory. Prior to turning the tuner off, the recommended PID parameters had been "switching" between two set of values: 13%, 0.27, 0.00, 0.02 versus 10.0%, 0.18, 0.05, 0.02. One set contained a recommended derivative constant and higher gain. When constants had been updated at t=352s, recommended values of RP=12.5%, RI=0.28 and RD=0.00 were transferred which resulted in the oscillations about 37.5%. When the "other set" was implemented (RP=10.7%, RI=0.20 and RD=0.06) at t=619s, subsequent controller performance was satisfactory. The corresponding values of confidence factors for the two sets of parameters gave no indication of which set might have given better control performance.

When the autotransformer output was increased to 70%

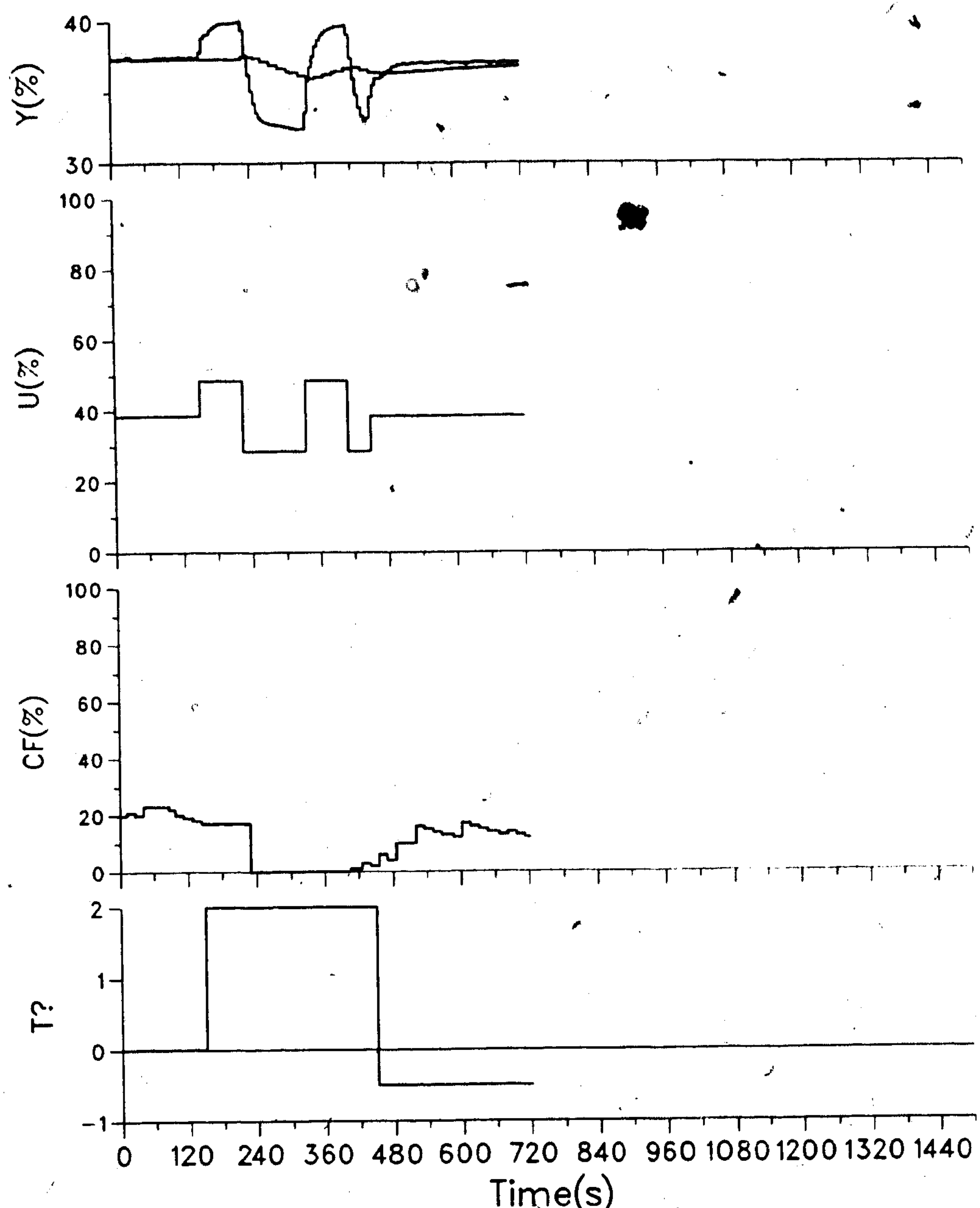


Fig. 7.17 Initial Test for Configuration #1 about 37.5%

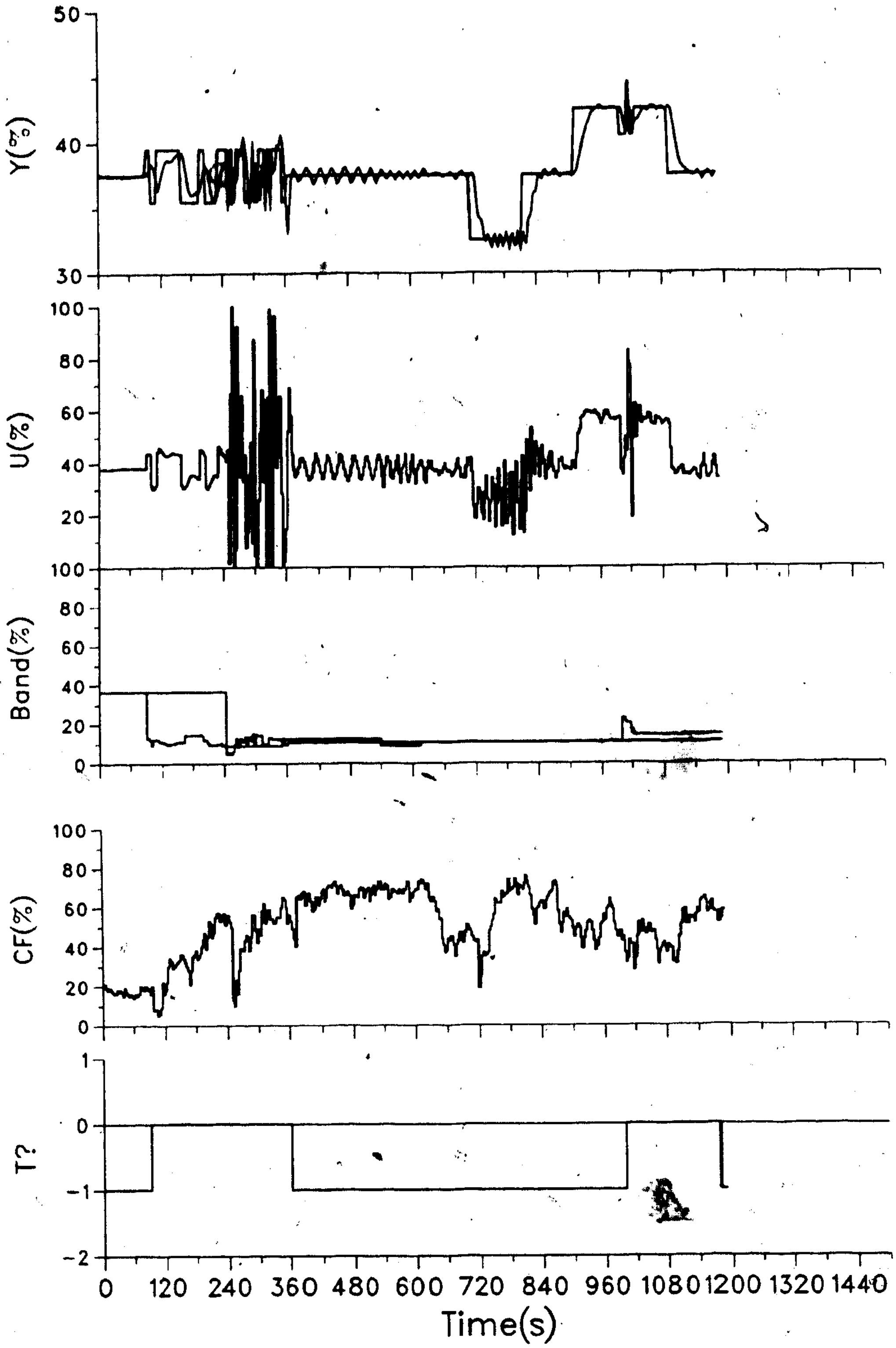


Fig. 7.18 Continued Tuning for Configuration #1 about 37.5%

following run Tc3_1.5, the implemented PID constants started to produce oscillations due to interaction with the increasing process static gain. Figure 7.19 shows the oscillatory behavior of the closed loop system. As the values of the confidence factor decreased, the model output also started to deviate from the true process measurement a significant amount. The resulting tuner performance is illustrated in Figure 7.20 with plots of RP, XP, CF and T? as a function of time. The tuner had to be reinitialized three times (at t=443, 657 and 1176s) in an effort to get higher confidence factor[illegible] Note that when T?=-1 in Figure 7.20, the tuner is OFF. [illegible]=1754s, the PID parameters were updated using RP=184%, [illegible]7, RD=0.26 and RT=0.10 even though the corresponding confidence factor was only 5%. The closed loop system was then able to stabilize.

Figures 7.21 and 7.22 on the following pages contain the results for run Tc3_3.5. Following the initial test (Ti3_3.5), automatic set point perturbations (AD=4%)were used to provide excitation for the tuner and allow for good adaptation. At t=173s the recommended parameters were transferred to give RP=31.9%, RI=0.28, RD=0.07 and RT=0.02. Subsequent closed loop performance was oscillatory when the tuner and set point perturbations were disabled. At t=365s the "continue tuning" was used (AD=1.0%) and parameters updated at t=493s to XP=72.9, TI=0.30, TD=0.00 and TT=0.02 with a confidence factor of 69%. When the tuner and set point perturbations were turned off, closed loop control was

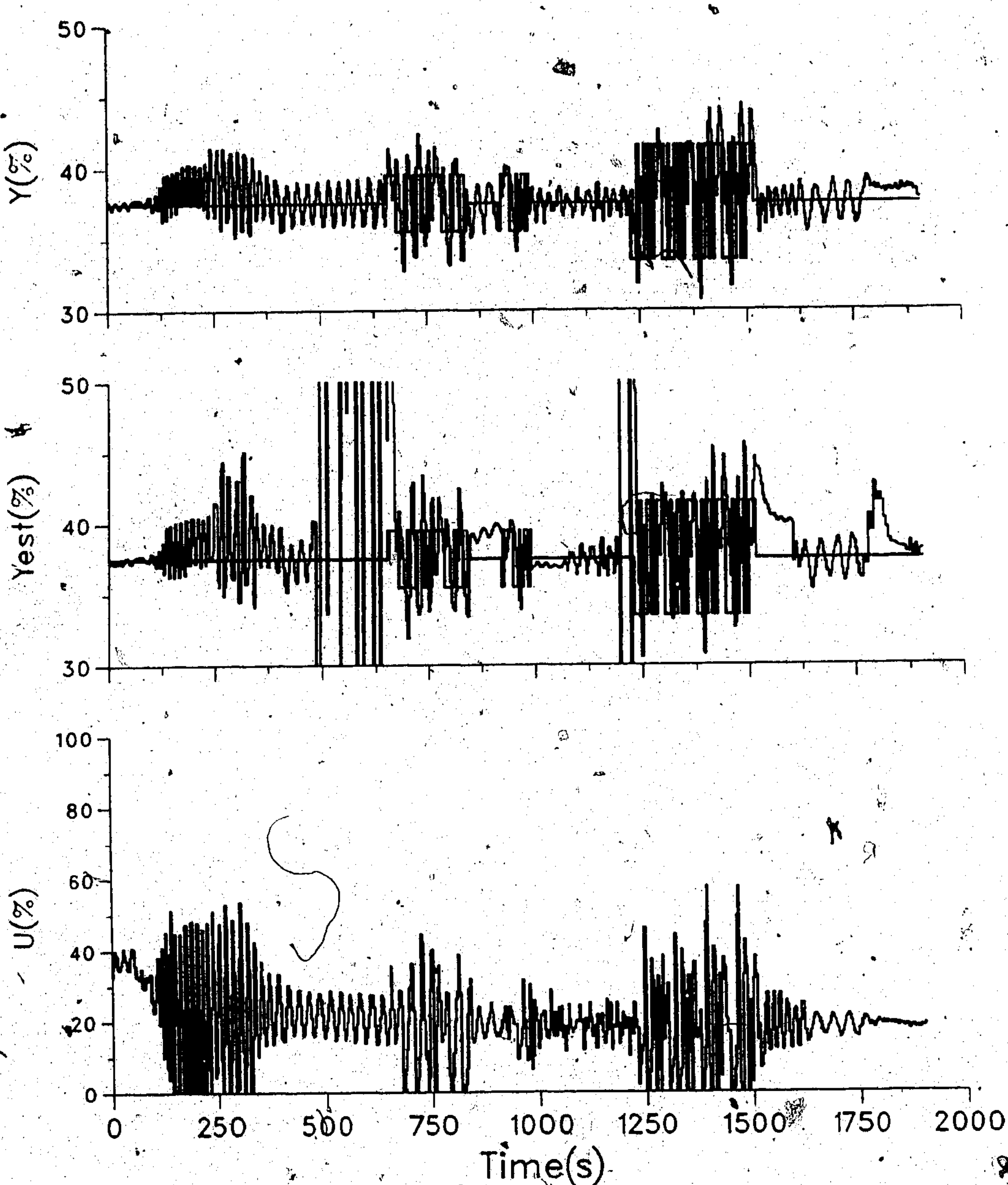


Fig. 7.19 Effect of Drifting Disturbance on Closed Loop Performance - Tc3_2.5 Operating at 37.5% Transformer 50→70%

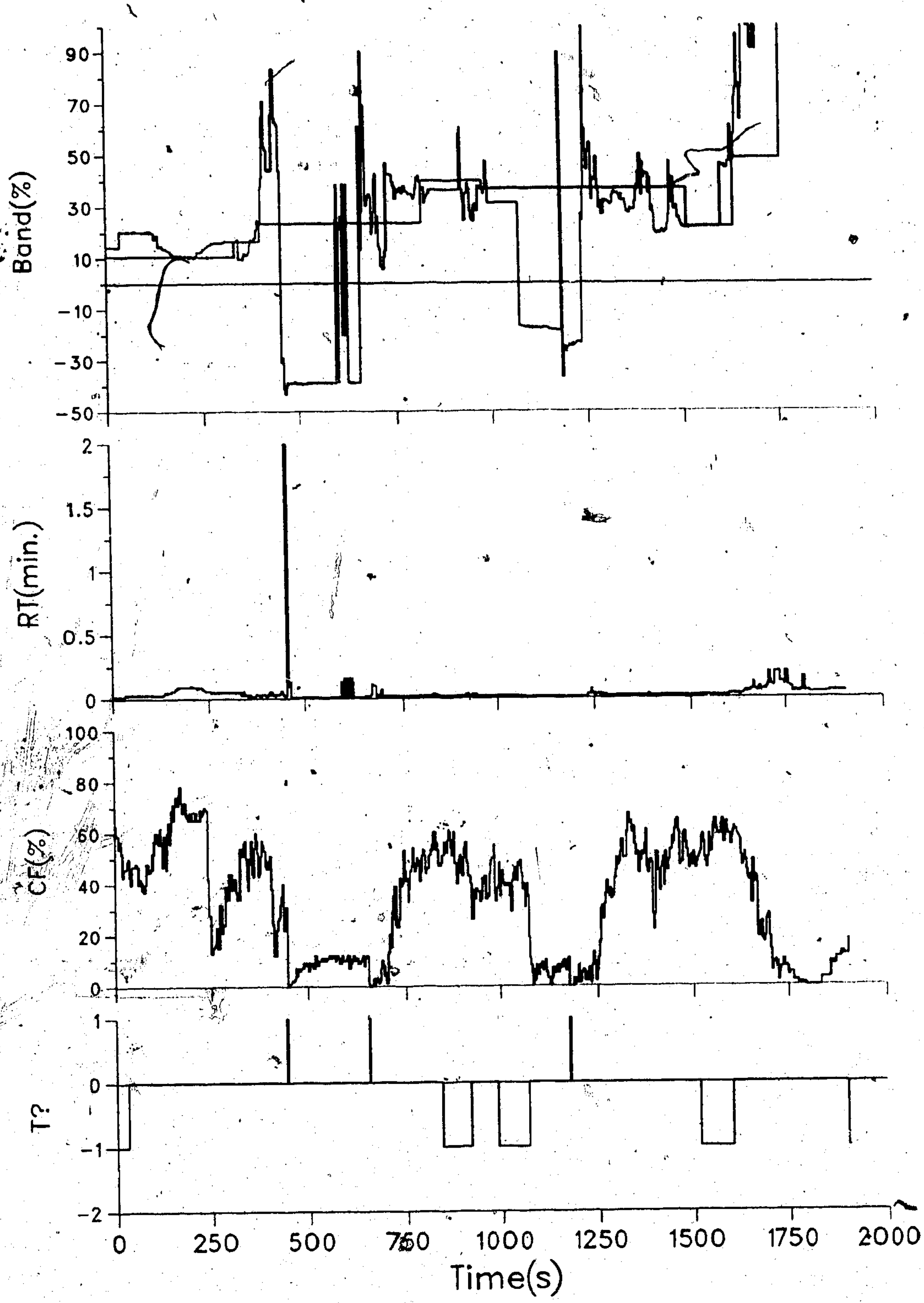


Fig. 7.20 Effect of Drifting Disturbance on Tuner Performance Tc3_2:5

still oscillatory.

At t=670s control was stabilized by adjusting the PID constants in the slave loop controller from values of 130/0.15/0.30 to 75.0/0.90/0.25 in order to increase the inner loop's speed of response. This effectively decreased the interaction between the two loops. At t=1170s the original values of 130/0.15/0.30 were again implemented into the slave controller.

The tuner's inability to give smooth control and converged PID parameters in run Tc3_3.5 is likely due to its sensitivity to incorrect specification of the process delay time, DT. From Figure 7.19 when the autotransformer output was increased to 70% the steady state air flow rate required to maintain the temperature at 37.5% of span decreased from ≃38.5 to ≃18.2%. These lower air flow rates will result in larger transport delays which was indicated by the Initial Test where the delay time changed from 0.0 to 0.02 minutes. This mismatch in DT resulted in low confidence factors and poor adaptation. The tuner quickly lost its ability to adapt and gave unreasonable PID recommendations.

If proccess delay times change, the tuner should be reinitialized with an initial test or the known time delay respecified. It is difficult to make conclusions about tuner performance because the criteria used for calculating/adjusting PID parameters is unknown.

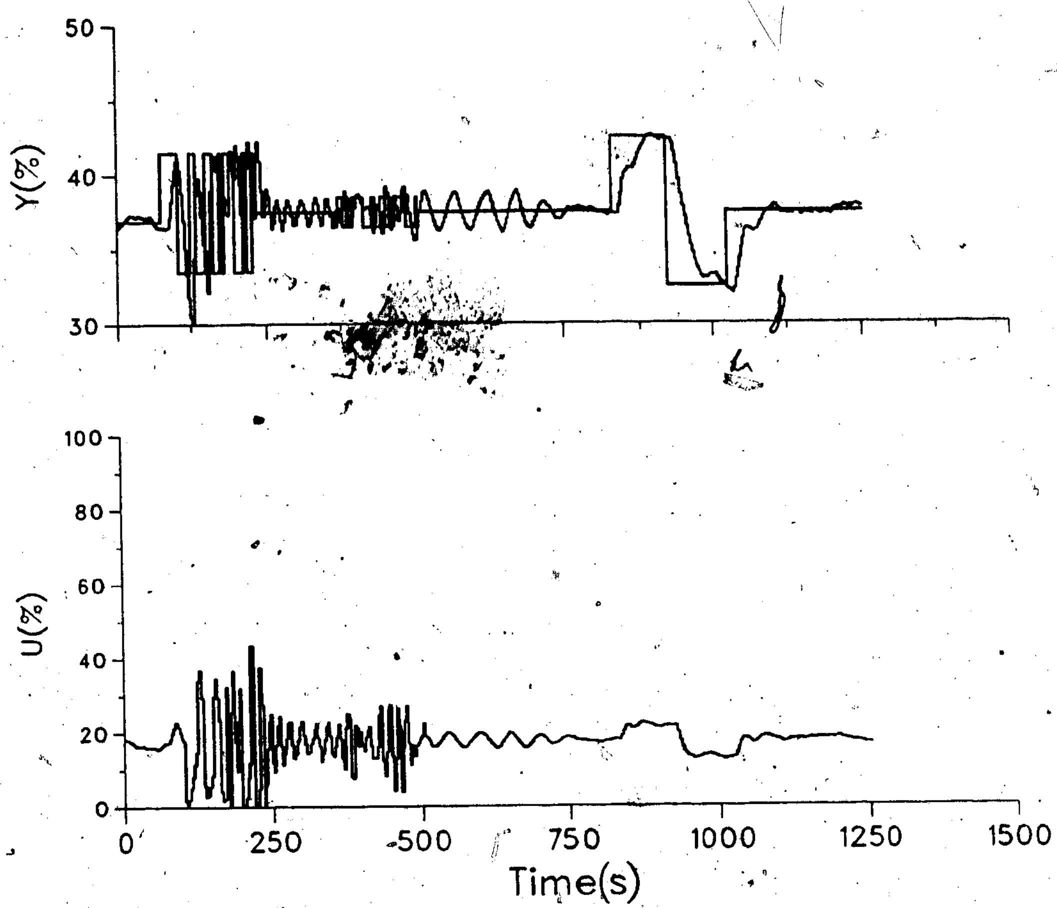


Fig. 7.21 Transient Response Using the Continue Tuning Option - Tc3_3.5 Operating at 37.5% Transformer at 70% following an Initial Test

7.4.4 Oscillatory Disturbances

The quality of recommended PID parameters from the auto-tuner is dependent on the performance of the modelling mechanism. If the dynamic behavior of the measured process output is affected by unmeasured disturbances and the tuner continues to model the process, the confidence factor will quickly drop. Because the changes in process output are no

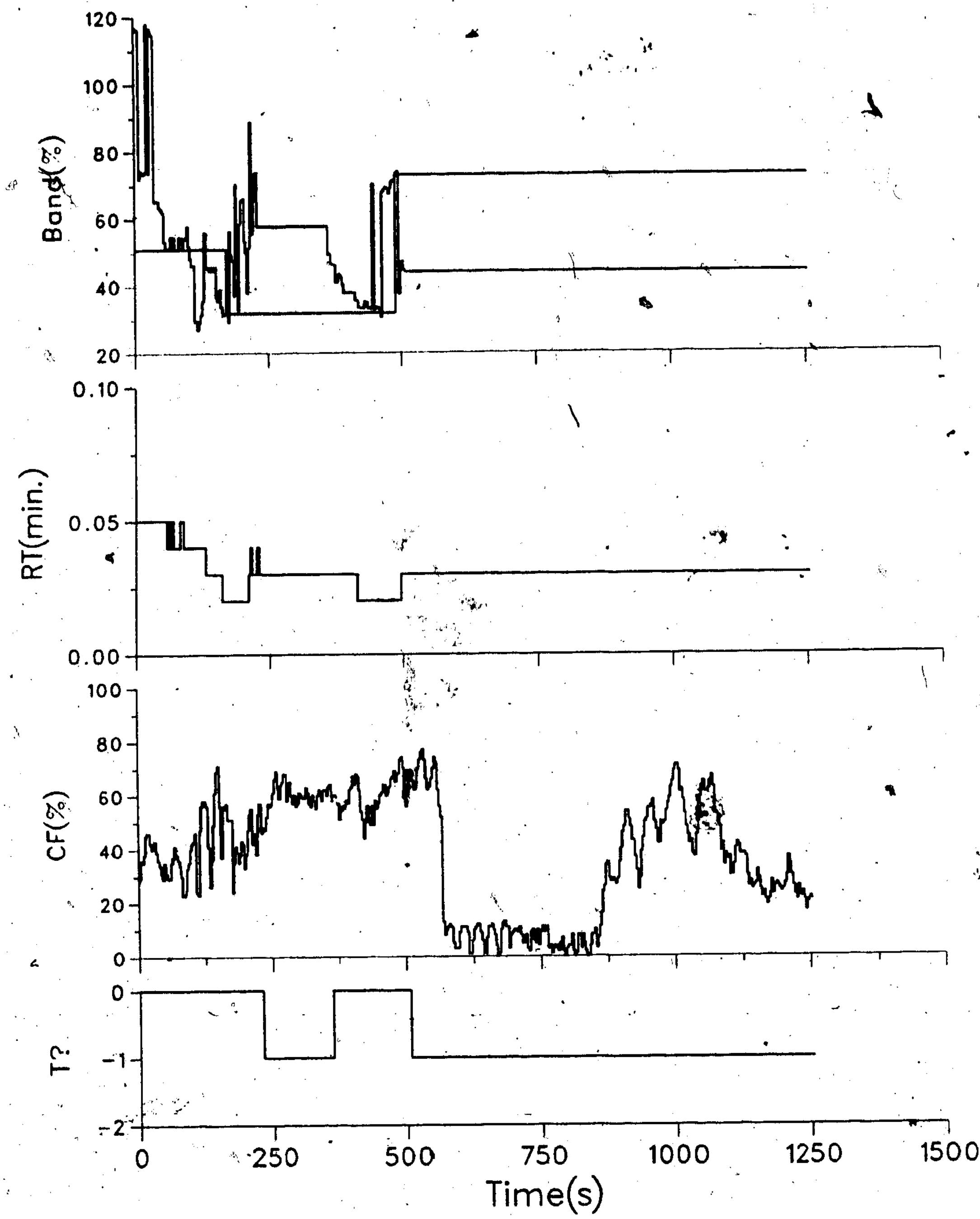


Fig. 7.22 Tuner Performance at 37.5% Operating Point - Tc3_3.5

longer correlated with controller output, the "u-y" model will become corrupted. Identification should not proceed when a process is subjected to unmeasured disturbances.

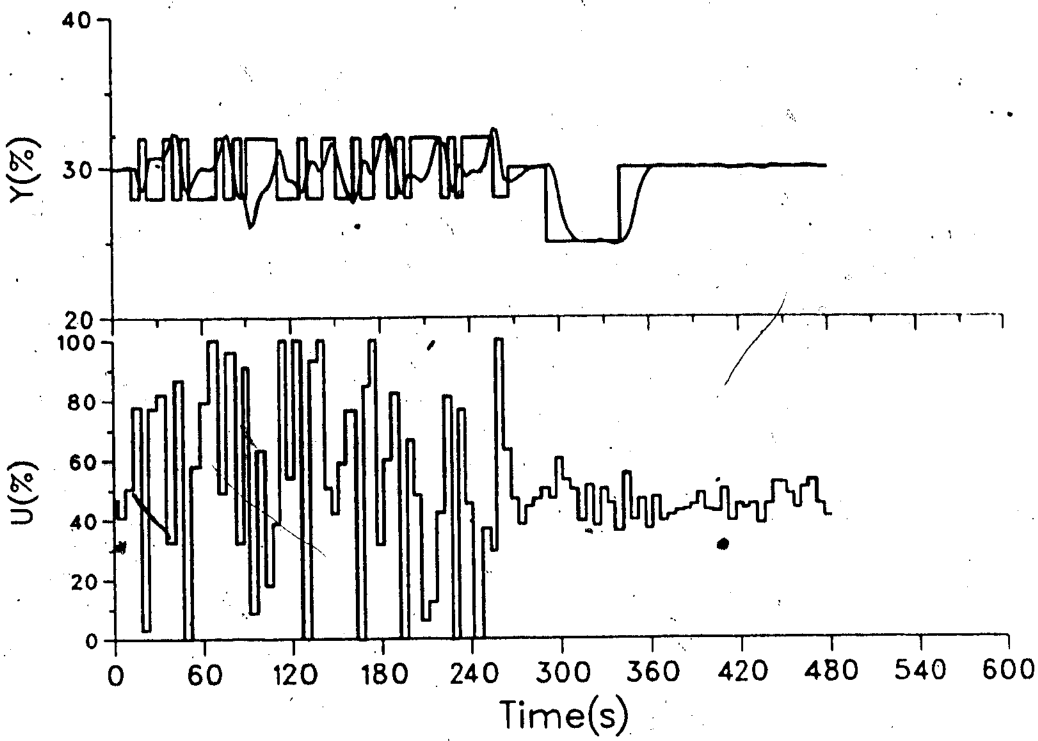
7.5 Evaluation of Controller Features

Because the "Auto-Tuning" controller does not automatically transfer its recommended PID parameters to the control law it has not been necessary to provide safeguards for poor tuner performance like retuning when controller output cycles or clamping of adapted PID constantsare not necessary.

An important option that is available is the use of digital filtering that was briefly discussed in section 7.4.2. Measurement noise can seriously affect tuner and controller performance if not removed from measured process output signals. The use of measurement filtering and its effect on performance was evaluated using the first order level controlled process.

Figure 7.23 contains transient responses for set point changes from a tank level of 30% to 25% and back to 30%. In run Tc2_3.2, a filter constant of IF=0.50s was used while for Tc1_0.2 no filtering was used. Control action in the latter case is much more oscillatory partly because of higher controller gains but mainly due to derivative action on noisy process measurements. Figures 7.24 and 7.25 contain corresponding plots of CF and the modelling errors $(y-\hat{y})$ for each run.

• When filtering is used, the recommended controller gains are lower resulting in less variance in control action. Confidence factors remain higher for longer periods of time and the tuner maintains its ability to adapt.



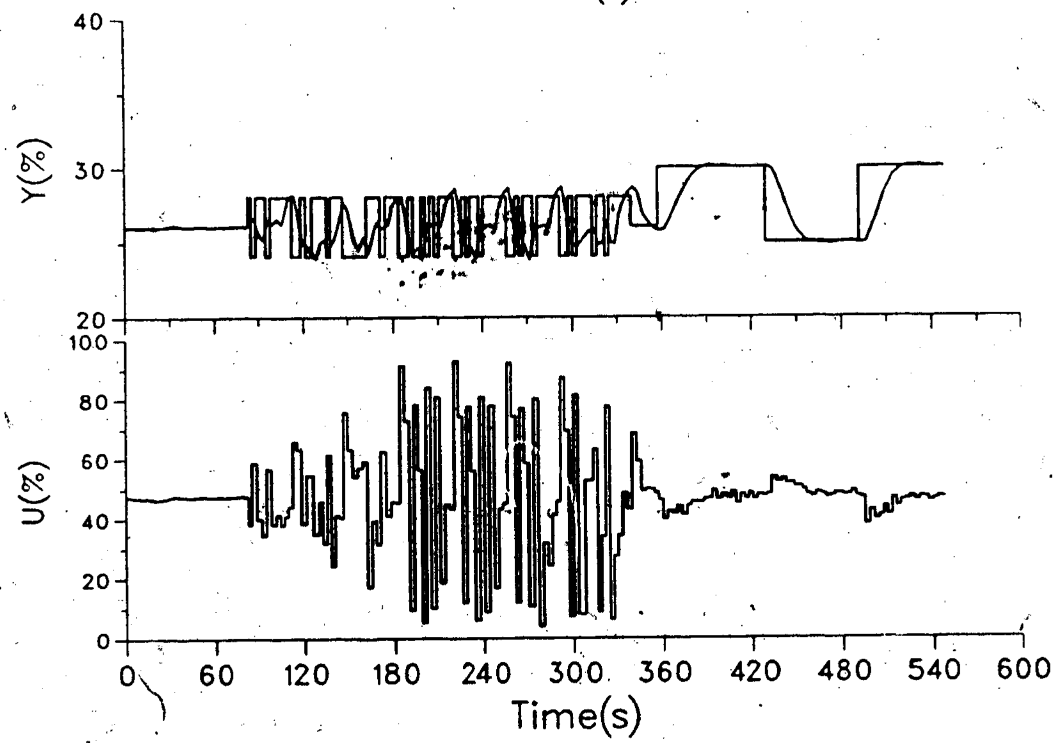


Fig. 7.23 Use of Digital Filtering to Improve Control Performance in the Presence of Measurement Noise - Runs Tc2_3.2 (IF=0.5s) and Tc1_0.2 (IF=0.0s)

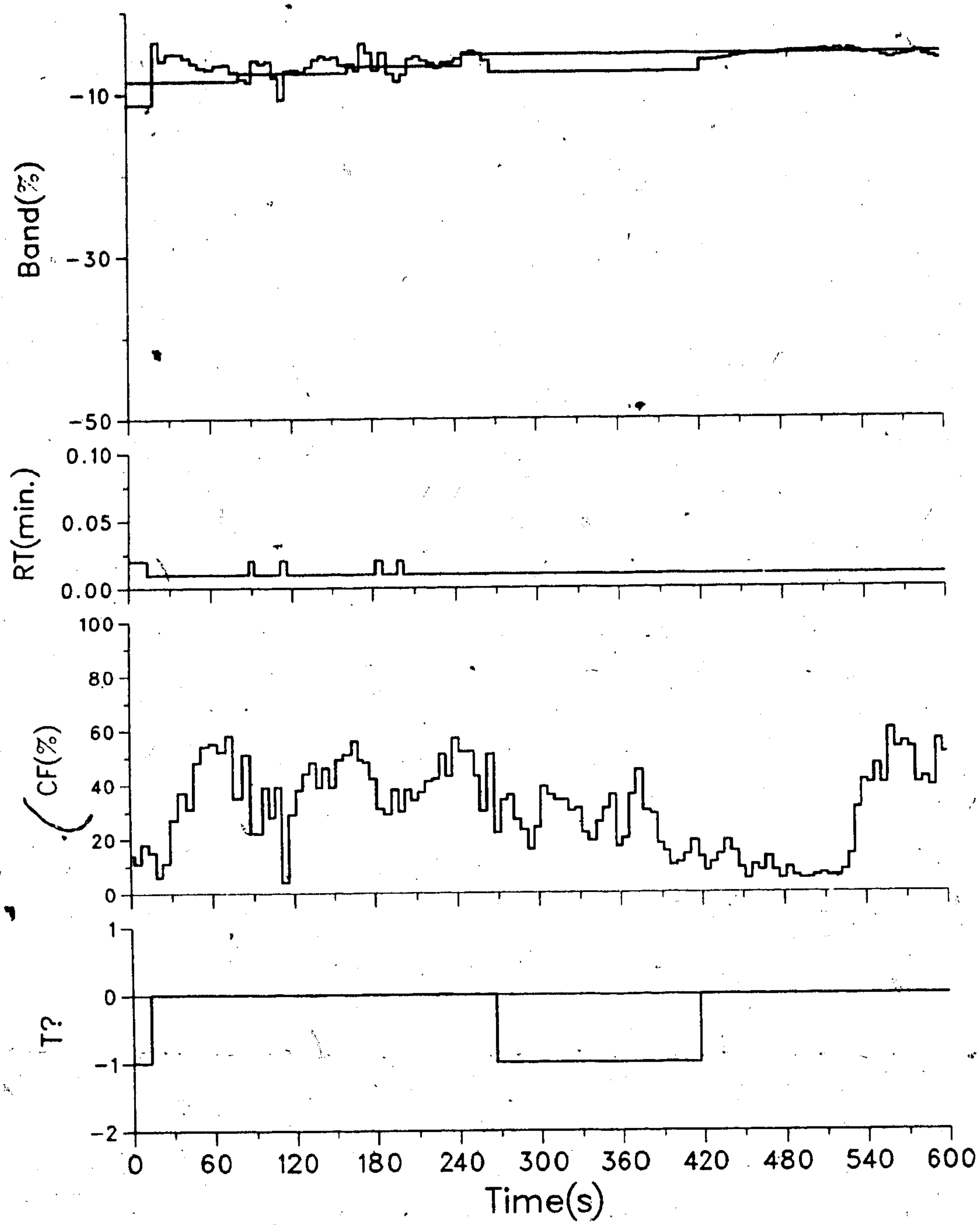


Fig. 7.24 Effect of Measurement Noise on Tuner Performance - Run Tc1_0.2 (IF=0.0s)

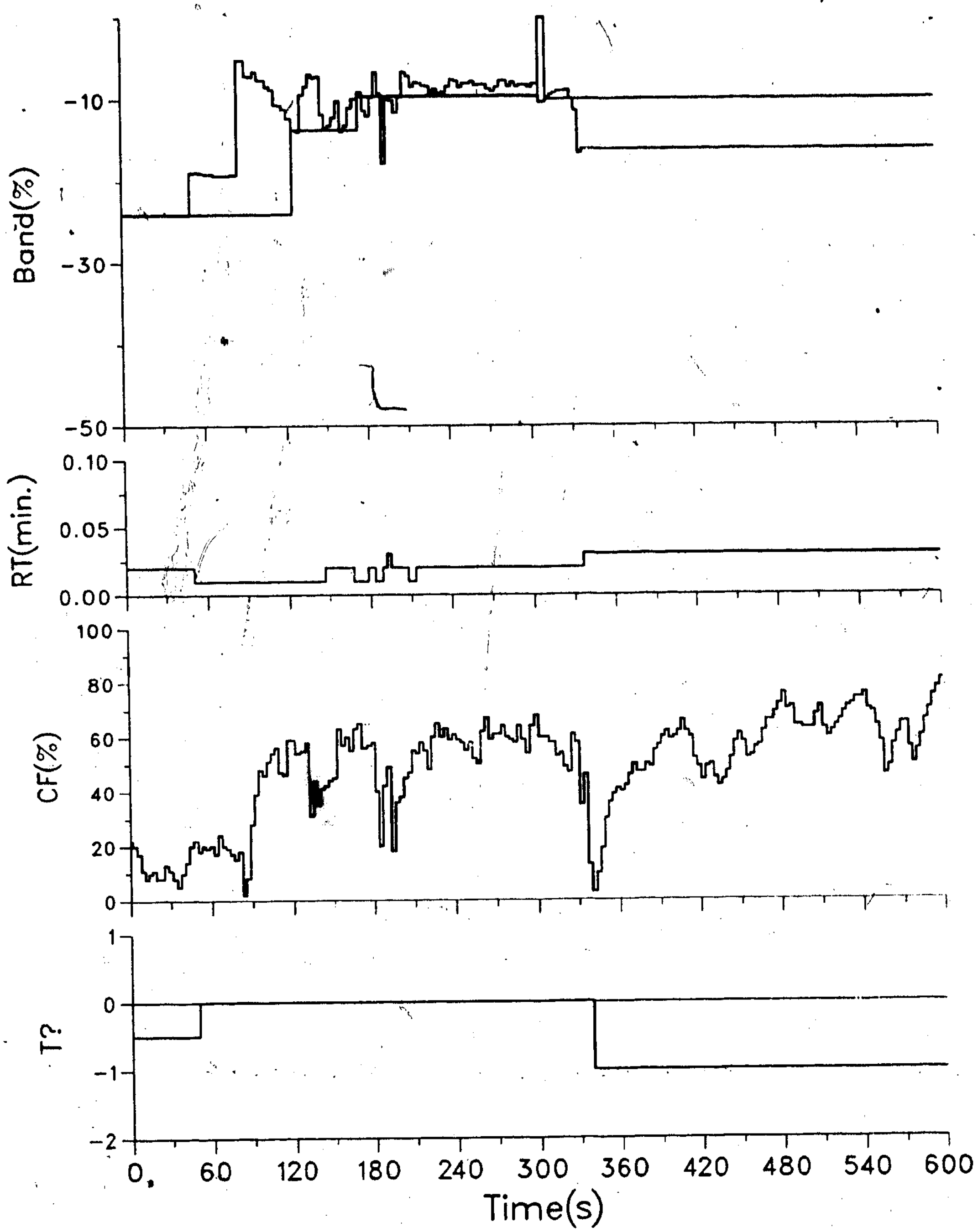


Fig. 7.25 Effect of Digital Filtering on Tuner Performance

7.6 Summary of Controller Features and Performance

Control Law

The 6355 Auto-Tuning controller uses a typical PID control law structure with derivative acting on the process measurement. There is no special filtering of derivative action so the variance of u(k) is sensitive to measurement noise.

Self-Tuning

The tuner uses measured process I/O data and user specified process delay time to generate a statistical model for predicting future outputs. Based on these predictions the tuner then calculates the optimum PID parameters needed to drive the measurement to the set point.

These newly calculated constants are not automatically transferred into the control law. The user must issue an "update" command to explicitly transfer these "recommended" values into the actual control algorithm.

The performance of the predictive model is continually monitored and a confidence factor is derived based on errors in predictions. The value of tnis confidence factor (0-100%) is meant to be a guideline for deciding how valid recommended PID parameters are.

The criteria on which the recommended PID parameters are based is not specified. Under ideal conditions, the auto-tuning controller gives critically damped closed loop responses to set point changes. The user can not specify desired closed loop performance.

During self tuning operation recommended values for PID parameters and tuner sample time are displayed. In the control of the two pilot plants used in this work, the tuner would sometimes change the recommended tuner sample time by a factor of three. This resulted in corresponding changes in the recommended PID values as well as the confidence factor. Sudden, large decreases in the confidence factor made decisions to update PID parameters difficult (Although the 6355 hardware samples at a fixed rate, the adaptive tuner samples at a slower rate dependent upon the dynamics of the process being controlled). For the control of a nonlinear high order process the calculated confidence factor rarely exceeded 50%. When recommended PID values were transferred with a confidence factor of 30-35%, closed loop control sometimes became unstable. If the operator inadvertantly updated the controller with an unstable set of PID parameters he had to manually modify the PID constants or switch to manual mode. An option should be provided for the transfer of stable backup constants in such an event. The tuner can not be used for long periods time. The confidence factor becomes very low ($\simeq$3%) and recommended PID parameters drift. In this work the recommended controller gains became very large $\simeq$100 as the confidence factors dropped. The generation of a predicted statistical model requires that the process delay time be known and fixed. When this assumption is violated, the performance of the model is poor (low confidence) and recommended PID values should not be used.

Assuming that the statistical model does give accurate predictions, the auto-tuning controller can give converged values for recommended PID parameters. Unlike trial and error tuning the 6355 delivers optimum control as soon as optimum prediction is achieved.

In order to obtain an accurate model, the 6355's tuner requires richly excited input output signals. Normal closed loop control with periodic step changes in set point is not enough. The use of automatic set point perturbations is virtually required for good model identification. Depending on current controller gains and or process sensitivity these perturbations result in unacceptably high variance in u(k).

The performance of this device as an adaptive controller is only as good as the prediction model obtained. Prediction performance was good for the linear pilot plant but suffered in the evaluation work with the nonlinear process.

Initialization and Start-Up

For normal operation, the operator must specify five parameters: initial PID values, the process delay time and the tuner's sample time. If reliable estimates are not available, an "Initial Test" can be used in which a square wave perturbation signal u(k) is used to excite the process.

The operator specifies the maximum allowable deviation in y(k) from its initial value and the magnitude of the change in controller output.

Parameters from the initial test results in damped closed loop response to set point changes.

Meaningful Feature, Options and Safeguards

Because the Auto-Tuning controller functions as an advisor and is designed for periodic retuning rather than continual adaptive operation, TCS has not provided many practical features.

The one much needed feature that is available is measurement filtering. Both the performance of the tuner and control law are sensitive to measurement noise. The use of filtering results in less variance in recommended PID parameters and more accurate predictions. The PID algorithm uses direct differentiation of process measurements making the use of filtering mandatory. Filtering significantly reduces the variance of u(k).

Based on the experimental evaluation of this controller several additional features should be provided:

A set of backup PID parameters should be available that can be easily transferred into the control algorithm. If an operator "updates" the controller with unstable PID parameters from the tuner he must modify the PID constants or switch to manual mode. The "update" of recommended PID and tuner sample time should be allowed only if the confidence factor exceeds some user specified lower limit.

Ease of Use

Information within the Auto-Tuning controller is accessed using a hand held configurator. Parameter values are retrieved from several database tables using two character mnemonics. The operator must keep a listing of these table so that he can determine in which table the desired parameter is located since he must "move" to that table before the particular parameter can be retrieved.

In this work a multi-tasking supervisory system was implemented so that large amounts of relevant data could be simultaneouly displayed and easily modified without requiring that the operator know the structure of the controller's database or parameter mnemonics.

Amount of Expertise Required for Operation.

The Auto-Tuning controller requires a considerable amount of supervision when used for periodic retuning. A reasonable value for the magnitude of automatic set point perturbations must be chosen. The derived confidence factor must be monitored so that a decision by the operator to transfer the recommended PID constants can be correctly made.

The amount of time and effort needed to obtain reliable PID constants and control increases for non ideal processes. It was difficult to get converged PID parameters for the nonlinear plant.

8. A^3 Experimental Evaluation

This Chapter presents an experimental evaluation of a control system based on Clarke and Gawthrop's GMV controller using a nonlinear process and a simple first order plant.

8.1 Functional Description

The A^3 controller is an adaptive predictive controller based on the Generalized Minimum Variance law of Clarke and Gawthrop (1979). GMV is essentially a non adaptive control law (1/Q) acting on predicted error. Predicted outputs are determined using recursively estimated parameters whose values vary with changing process dynamics. The recursive estimation of the predicted output makes up the current adaptive part of GMV.

The actual implementation of this GMV is completely general allowing a control engineer to easily specify P,Q and R transfer functions as desired. Parameters within these filters can be easily modified on-line. However, the supervisory task or "Process Operator's Communications" task automatically initializes the GMV to give a PI control law acting on predicted control error. Set point changes are exponentially filtered by default, for tailoring purposes. The degree of set point filtering is determined by the user specified desired closed loop time constant.

8.1.1 Adaptive Prediction and Self-Tuning Mechanism

The A^3 controller contains two adaptive schemes. A recursive algorithm is used to estimate predicted outputs (see Chapter four) with an implicit scheme. Parameters in a predicted output equation (3.12) are estimated on-line and used to calculate future process measurements. For noise free processes, classical RLS with UDU^T factorization and Fortescue's variable forgetting factor can be used. The improved least squares algorithm, ILS, described in Chapter four with its useful ON-OFF criteria is a more practical scheme that minimizes unwanted drift in prediction estimation due to numerical effects, poor conditioning and noisy I/O data.

A second ad hoc tuning scheme has been designed and implemented to modify the degree of filtering on controller outputs based on user specified overshoot (see section 3.5). When the A^3 controller is operated with P or PI Q-filtering, a tuning mechanism is used to modify the effective controller gain following set point changes. Without this adaptation scheme, the GMV controller remains fixed gain and therefore unsuitable for highly nonlinear processes. This second tuner is designed to operate when set point filtering has been disabled. The tuner requires an initial estimate of the open loop process time constant in order to know how long it will wait for peaks following a step change in set point.

At any time during its operation, the PI control parameters can be replaced by user specified values and/or this tuner can be turned off. Likewise, a fixed parameter PID controller is available as a backup controller. The fixed gain controller allows for smooth closed loop startup of the GMV controller by first allowing it to perform identification while the conventional one carries out control.

8.1.2 Required Inputs

Ease of use and simplicity are two factors that will contribute to the success of any controller. Historically, self-tuning controllers have been considered too complicated for practical use. Process model structures must be specified, PQR filters chosen, initial parameter estimates for predicted outputs selected, and so on.

The previous discussions on Q weighting and set point filtering have led to the selection of default values for Q and R. Likewise, the choice of a default process model structure with second order denominator and first order numerator dynamics gives good tracking for a wide range of processes.

The following information is required for normal operation and being process dependent, must be specified by the operator:

Kc,Ki Initial values for desired PI controller parameters. These are used to specify the value of $Q(z^{-1})$. If these parameters are not available, use the open loop identification test discussed in section 8.1.4.

τ_p Open loop process time constant (approximate). The A^3 controller requires τ_p as an estimate of the time scale of closed loop dynamics. When the second tuner is used, it will search for peak heights until $4 \cdot \tau_p$ units of time have expired since the set point change. It is critical that τ_p is not underestimated since this may result in unreasonably high controller gains.

τ_d Process delay time. The GMV controller's adaptive prediction algorithm must have an accurate estimate of the total delay time in the closed loop. Inacurrate values may lead to poor estimation of predicted outputs and possibly unstable control.

For the purposes of predicting process measurements there are additional parameters that must be initialized. Default values for these parameters have been provided to minimize the amount of information needed for startup.

Process Model Structure

The assumed orders of $A(z^{-1})$, $B(z^{-1})$, $C(z^{-1})$ and $L(z^{-1})$ must be specified. If feedforward compensation is not used, NL=-1 since for NL=0, $L(z^{-1})=l_0$. The default values: NA=2, NB=1, NC=0 and NL=-1 specify a process with second order denominator and first order numerator dynamics whose processs output is corrupted by white measurement noise and not influenced by any measured disturbances.

P(0)

Initial covariance matrix is specified as 10·I. The value of the diagonal elements in P(0) directly affect the gain in the recursive estimation scheme. Larger values of P(0) indicate greater uncertainty in the initial estimates of θ and result in more active identification. The operator can override the default specification at initialization time. During closed loop operation a covariance multiplier can be used to inflate P(k). The covariance matrix is automatically inflated following a set point change by some user specified factor (default 10) for six sampling periods and then deflated by the same amount. Covariance inflation increases the RLS algorithm's sensitivity to changes in $\theta(k)$ and allows for faster adaptation.

$\theta(0)$

Initial estimates of parameters used to calculate th predicted output, $\hat{y}(k+d)$. These parameters are not explicit estimates of the process model and can not be directly initialized using estimates of $A(z^{-1})$, $B(z^{-1})$ and $C(z^{-1})$ in (3.1). Default values of 1.0 are used.

$\lambda_u(0), \lambda_y(0)$

Forgetting factors for mean estimation in ILS are initialized to 0.75. The operator may specify other values for either factor. Smaller values will result in faster discounting of I/O data when determining means. If these factors are too small the mean deviation data used for RLS identification will lose useful dynamic information. If the factors are too large $\simeq 1$, the tracking of d.c. levels will be too slow. This will result in slow offset rejection since poor mean estimation will give inaccurate predictions.

IOTA

User specified lower limit on $\|P(k)x(k)\|$ (used for ON-OFF criteria) is initialized to 10^{-3}.

CONDP

User specified upper limit on condition number of P(k) (used for ON-OFF criteria) is initialized to 500.

The above default values are adequate for most applications. Results will demonstrate that the performance of the adaptive predictor is not sensitive to small changes in model structure. If desired, the structure can be modified prior to startup as can any of the other parameters.

8.1.3 Additional Features and Inputs

To widen the A^3 controller's potential range of application, additional features are provided that allow the user to make use of known process information, handle specific process conditions and minimize the need for operator intervention.

Maximum allowed overshoot. Default value 0.0 (OVR)

The operator can specify the degree of damping desired for closed loop responses to set point changes. In theory the cost function given by (3.16) attempts to minimize predicted control error, but since control weighting is necessary for most applications, minimum variance control is not realizable. Using an inverse PI control law structure for Q and a few heuristic tuning rules, the GMV can be tuned to give a particular overshoot following set point changes.

Exponential Filtering of Set Point Changes - TOWSP

Discrete filtering of set point changes can be used to produce smoother transitions between operating levels. A specified set point filter constant of 0.0 seconds results in no filtering. If set point filtering is enabled, a default value of $\tau_p/3$ is automatically selected. The user can respecify the value of TOWSP any time and $R(z^{-1})$ will be updated accordingly.

Noise band - NB

User specified limits on process measurement noise (white) can be used by GMV to determine a value of IOTA. Because the ILS algorithm uses mean deviation data, at steady state the only measured signals entering the recursive algorithm are due to white noise. Since $Tr[P(k)]$ is held constant and I/O data in $x(k)$ is due to noise, an approximate limit for $\|P(k)x(k)\|$ can be determined.

Digital Filtering of Process Measurements - TOW

A discrete finite difference approximation (bilinear transformation) of a second order Butterworth filter is used as a low pass filter (Phillips and Nagle, 1984).

Gain Compensation

This option has been provided to allow the GMV to adjust its effective controller gain automatically, following a set point change. Using an operator specified steady state open loop characterization, the controller will immediately adjust the degree of weighting on control following a set point change. The use of gain compensation allows the controller to maintain user specified closed loop response characteristics through the operating range of a nonlinear process.

Ideally, perfect gain compensation would result in convergence of adapted controller gain independent of operating levels. Internal adjustment of controller weighting prevents problems associated with moving from low process gain to high process gain regions.

With gain compensation in use, the second tuner (based on Overshoot) should only adjust Kc in response to changes in overshoot specifications and differences between actual and assumed nonlinearities.

8.1.4 Initial Identification Option

To minimize the amount of knowledge required for startup, an open loop identification option has been provided. The option involves the use of two sequences of controller perturbations. With the open loop system initially at steady state, the process is perturbed by a step change in controller output and the process reaction curve monitored. From the open loop response, the process delay time, open loop time constant and static gain are determined. With this data, the sampling interval, initial PI controller parameters and set point filter time constant are calculated. When this sequence is completed, and the process allowed to return to its original steady state, the second phase is started.

A square wave perturbation signal with user specified output deviations is used to excite the process. The period of the square waves is dependent on the maximum allowable deviation in process output specified by the operator prior to starting the open loop test. The resursive ILS algorithm is turned on and estimates for $\theta(k)$ calculated. User specified limits on allowable deviations in process output determine the period of the square wave perturbations. These limits allow the operator to specify the amount of process upset in the test. Figure 8.1 illustrates a typical open loop identification test performed using the nonlinear temperature controlled process. Note that at t=360s the controller was put into automatic mode using the PI parameters determined from the first phase of the open loop test. The process output was oscillatory due to the value of the controlle gain (25% overshoot and $\frac{1}{4}$ decay ratio specifications).

Recursive estimation of predicted outputs (open loop) can not be carried out using a single step change in u(k). The square wave signal is needed to provide sufficient excitation so that $\theta(k)$ can be estimated.

It should be noted that the test is completely automatic once started. Using TOWSP, PI estimates and the current sample time T_s, the initial Q and R filters are calculated and implemented. The second phase of the test is completed when the controller output is returned to its original steady state value following the fifth perturbation. When the test is completed, recursive estimation of θ is suspended until the operator proceeds with closed loop control or explicitly starts adaptive prediction via the

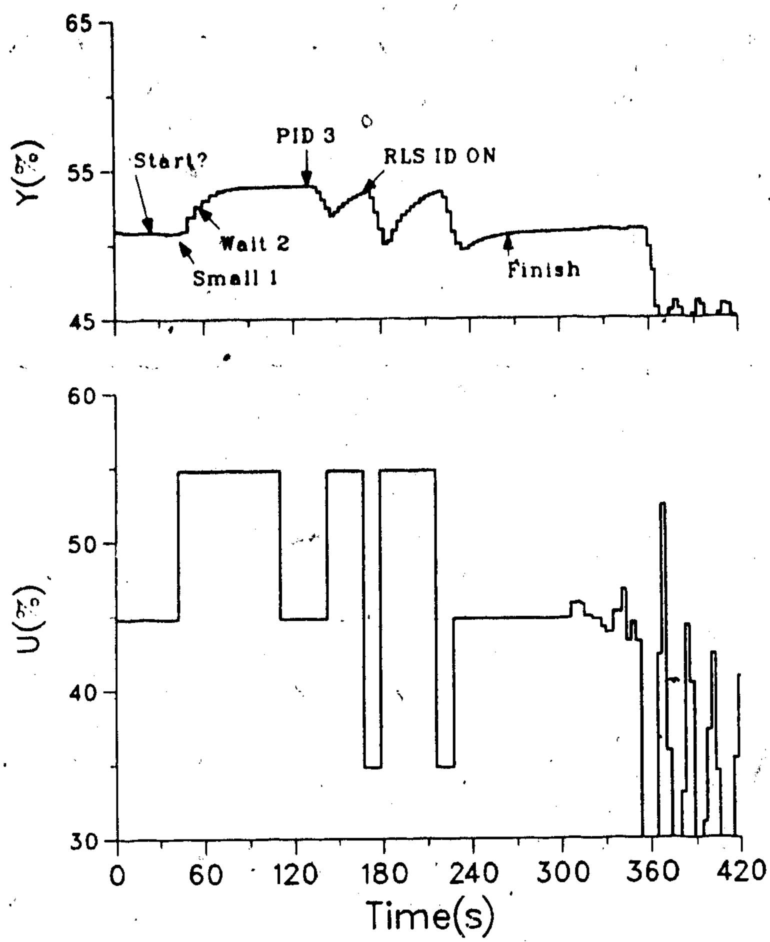


Fig. 8.1 Open Loop Identification Test for A^3 Control System

supervisory program.

Some of the Figures on the following pages contain plots of variables that indicate the status of the A^3 system during its operation. These parameters are defined below:

Acm Flag - Status of the GMV controller.

1- Has performed a complete initialization from disk files.

2- Is buffering data vectors with measured, filtered I/O data.

4- Is performing least squares identification (and 2 above).

5- Is calculating controller outputs (plus 2 and 4).

6- Control calculation is suspended by the operator (steps 2 and 4 remain on).

ALG - Control algorithm currently in use.

1- Conventional PID.

2- GMV.

Id Flag - Status of recursive least squares algorithm.

0 - ID is suspended.

-1 - ID is on.

8.2 Use of Controller Output Filtering - 1/Q Control

This section experimentally demonstrates the effect of several Q structures on closed loop performance and verifies the discussion in section 3.4.2. For results in this section the default second order model (NA=2 NB=1 NC=0 NL=-1) was used. P and R were held fixed with values of 1.0 and digital filtering of measurements was not used. Sampling and control intervals were fixed at two seconds.

8.2.1 Minimum Variance Control (P=R=1, Q=0)

With the above choice for P,Q and R, stable closed loop control is dependent upon the performance of the recursive estimation scheme. Calculated controller outputs depend upon the value of $1/\hat{g}_0$ which is an estimate of $1/(b_0e_0)$ and are extremely sensitive to variations in $\hat{g}_0$, particularly at fast sampling rates. The leading coefficient, b_0, of $B(z^{-1})$ is a function of static process gain and sampling time.

Due to sensitivity to estimates of g_0, the GMV can not be commissioned without first performing on line recursive identification of $y(k+d)$ in order to obtain "good" estimates of $\theta(k)$. With the A^3 control system, this was performed using: conventional PID closed loop control, operator implemented changes in $u(k)$ or the initial test.

Figure 8.2 illustrates minimum variance control. For this particular run the RLS algorithm was turned on while in manual mode. Using Kc=5/Ki=25s. conventional PI control and several set point perturbations were implemented to provide useful I/O data for parameter identification. P(0) and θ(0) were initialized to 100I and 1 respectively.

While at steady state, control was transferred to the minimum variance STC (t=280s). Due to either imperfect modelling or sample time effects the closed loop response began to oscillate. Nonlinearities in the process made it impossible to achieve perfect modelling and therefore good control with Q=0.

8.2.2 Proportional Control (P=R=1, Q=inverse PI)

The use of Q to stabilize closed loop performance is demonstrated in Figure 8.3. Following minimum variance control (90s<t<180s), Q was implemented as an inverse PI controller with Kc=4 and Ki=500s, effectively disabling the integral action. Upon addition of Q, the closed loop became stable but a steady state offset resulted. Note how the process measurement moved towards w(k)=50% at t=285s. At this point the integral constant was changed to Ki=40s and the controller offset was eliminated.

8.2.3 Integral Control (P=R=1, Q=1/λ)

Figure 8.4 illustrates typical performance using integral action alone. Following two set point transients using conventional PI control (Kc=5/Ki=20s) the GMV controller was activated with an integral time of 5 seconds at t=310s. Two subsequent set point changes demonstrate slow, but acceptable, servo control. The effect of nonlinear process

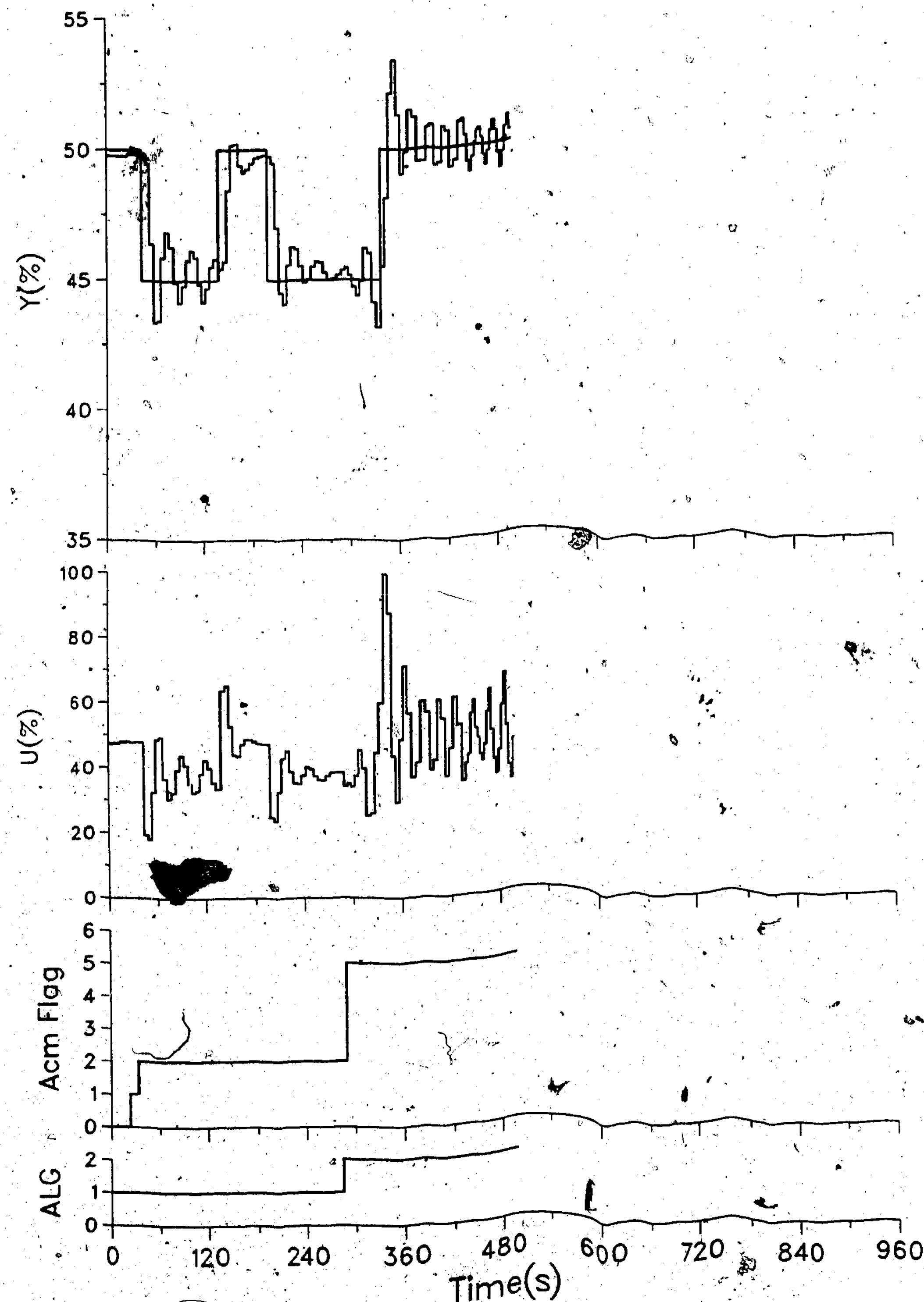


Fig. 8.2 Closed Loop Performance Using Minimum Variance Control

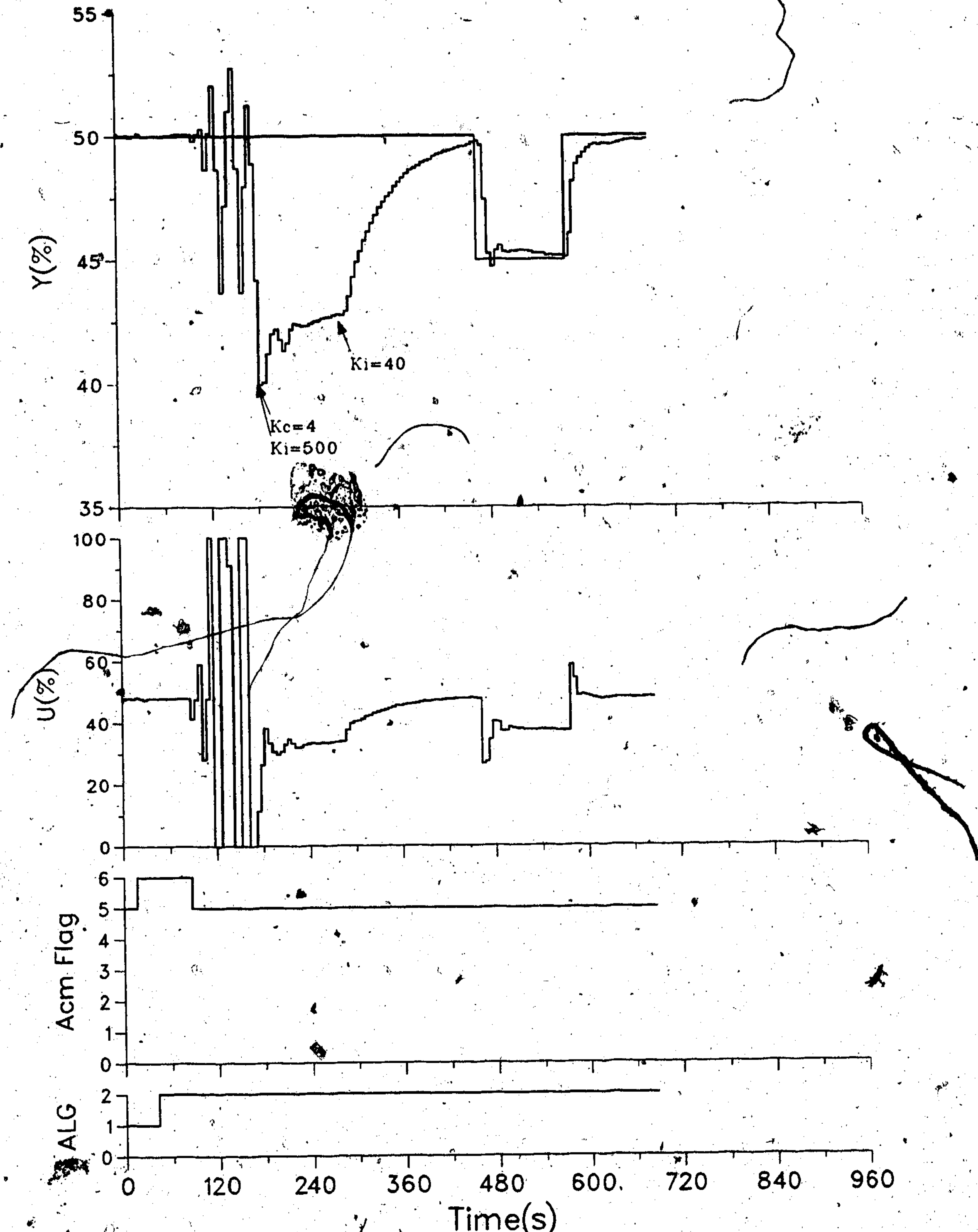


Fig. 8.3 Stabilization of Closed Loop Using Proportional Form of Q(z) (Kc=4.0 Ki=500s)

dynamics is evident from the slightly more oscillatory response to the negative set point changes.

Figure 8.5 clearly illustrates the effect of varying Ki on GMV closed loop responses to set point changes. The first two transients were oscillatory with Ki=1s. By changing Ki to 10s (higher penalty on u(k)) a significant change in closed loop performance was realized. Controller output was much less oscillatory and changes smaller, resulting in a damped transition to the new operating point. The negative set point changes result in less damped response due to higher process gains in the lower operating regions.

While the integral structure is necessary for offset elimination the effect of adjusting Ki is clear. There is a direct tradeoff between speed of resonse and degree of damping. The effect is even more pronounced for processes with large time constants.

For servo control, the integral only mode is not adequate since highly underdamped responses can only be prevented at the expense of sluggish control.

8.2.4 Proportional Integral Control (P=R=1, Q=Inverse PI)

The use of Q as an inverse PI controller allows for a variable degree of weighting which is a function of predicted error, $(w(k) - \hat{y}(k+d))$.

Figure 8.6 demonstrates how adjustment of effective controller gain and integral time affects transient responses to set point changes. As in conventional PI

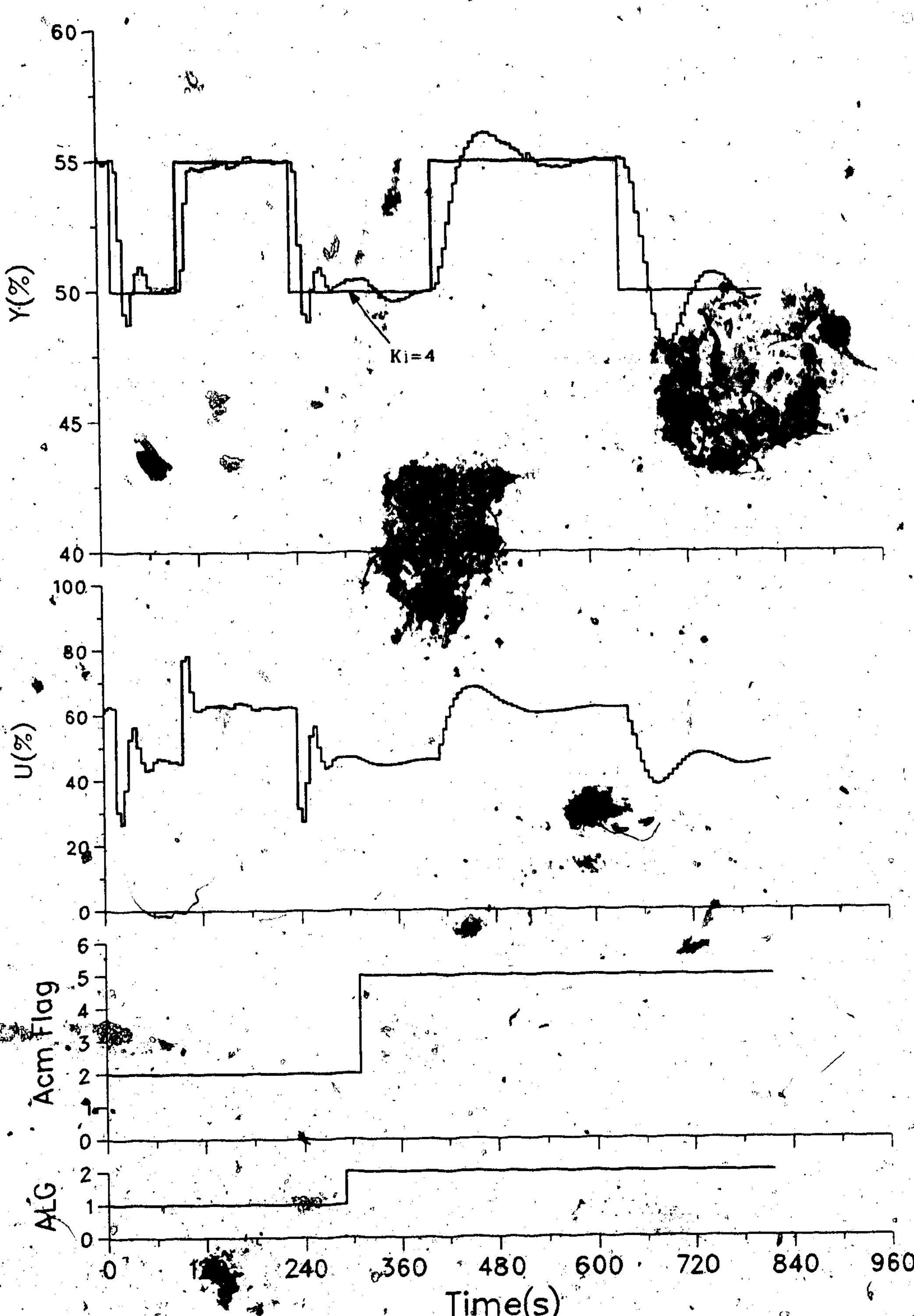


Fig. 8.4 STC-Integral Only Mode Ki=5s

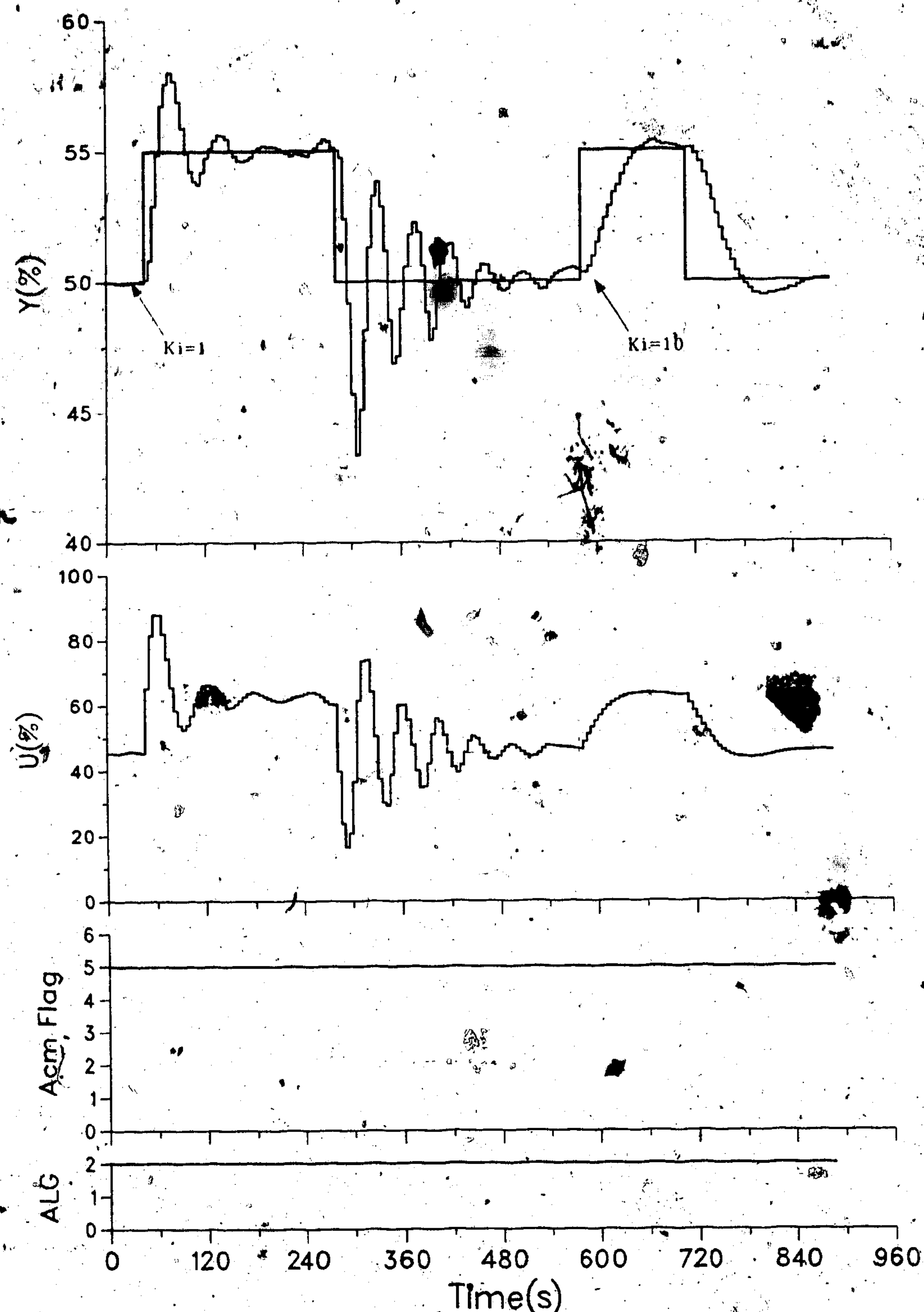


Fig. 8.5 STC - Integral Mode. Effect of Varying Ki on Closed Loop Performance

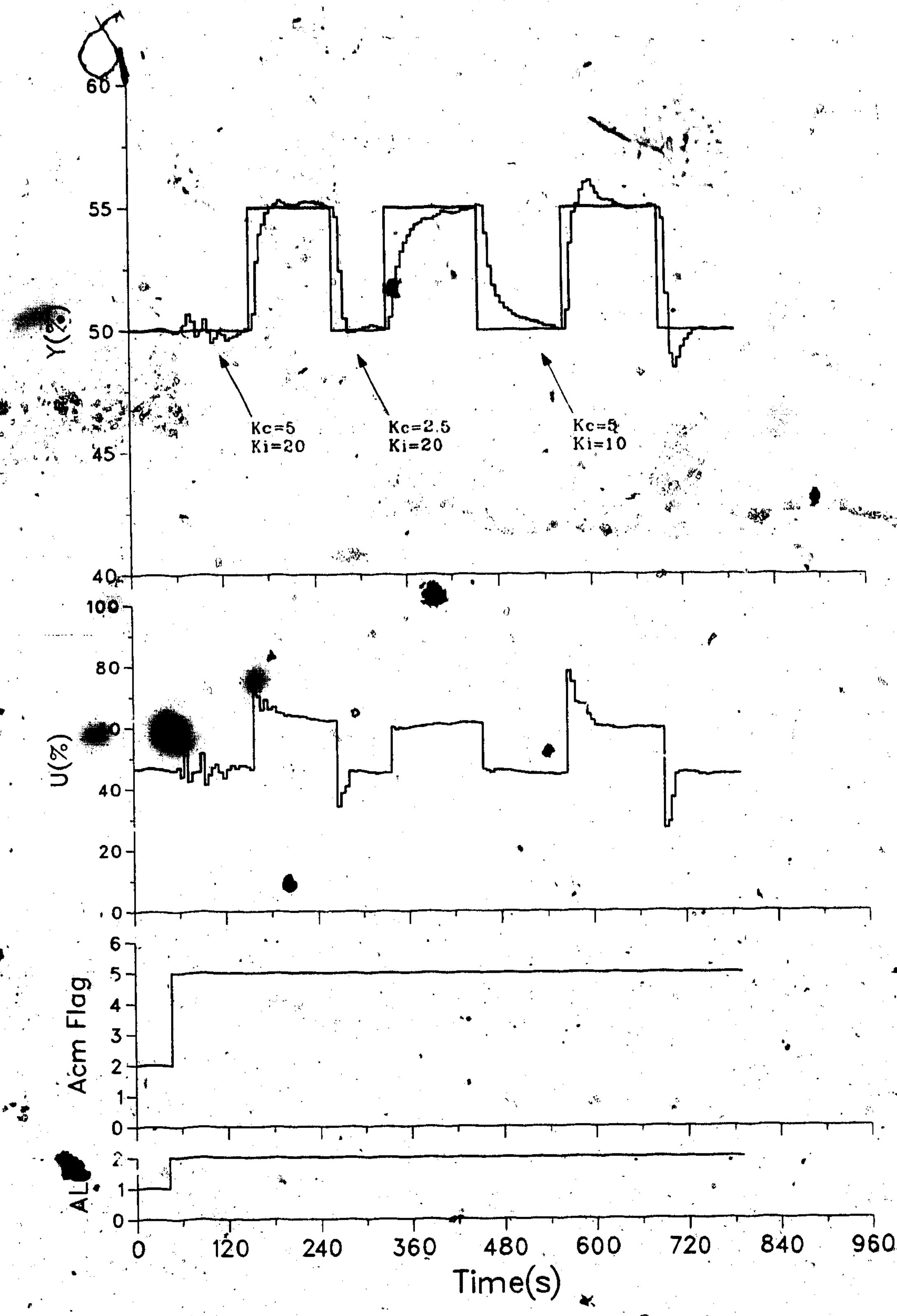


Fig. 8.6 Predictive PI Control - Effect of Tuning Kc and Ki

control, higher gains give faster response and eventually overshoot. Likewise, increasing integral action gives more underdamped behavior. The two transients in Figure 8.7 show the effect of increasing STC's gain on transient response to set point changes.

8.2.5 STC's Predicted Error vs Conventional Error PI Control

This section presents results that support the interpretation that predicted error effectively introduces phase lead into the closed loop system. This allows for the use of higher controller gains and more stable control as with derivative action.

Figure 8.8 illustrates that STC gives less oscillatory control than conventional PI control for equivalent Kc/Ki values. In order to stablilize the closed loop using conventional PI control both the controller gain and integral time had to be detuned.

Figure 8.9 demonstrates the effect of process nonlinearities on performance. Using conventional PI control, detuning was required after each negative set point change. Due to detuning, a subsequent positive set point change gave very sluggish control. At t=485s control was switched to STC (PI) using the same original PI constants. Much less detuning was required allowing for a narrower range of controller constants and better overall control.

The use of predicted control error gives better performance and <u>adapted</u> predicted error partially

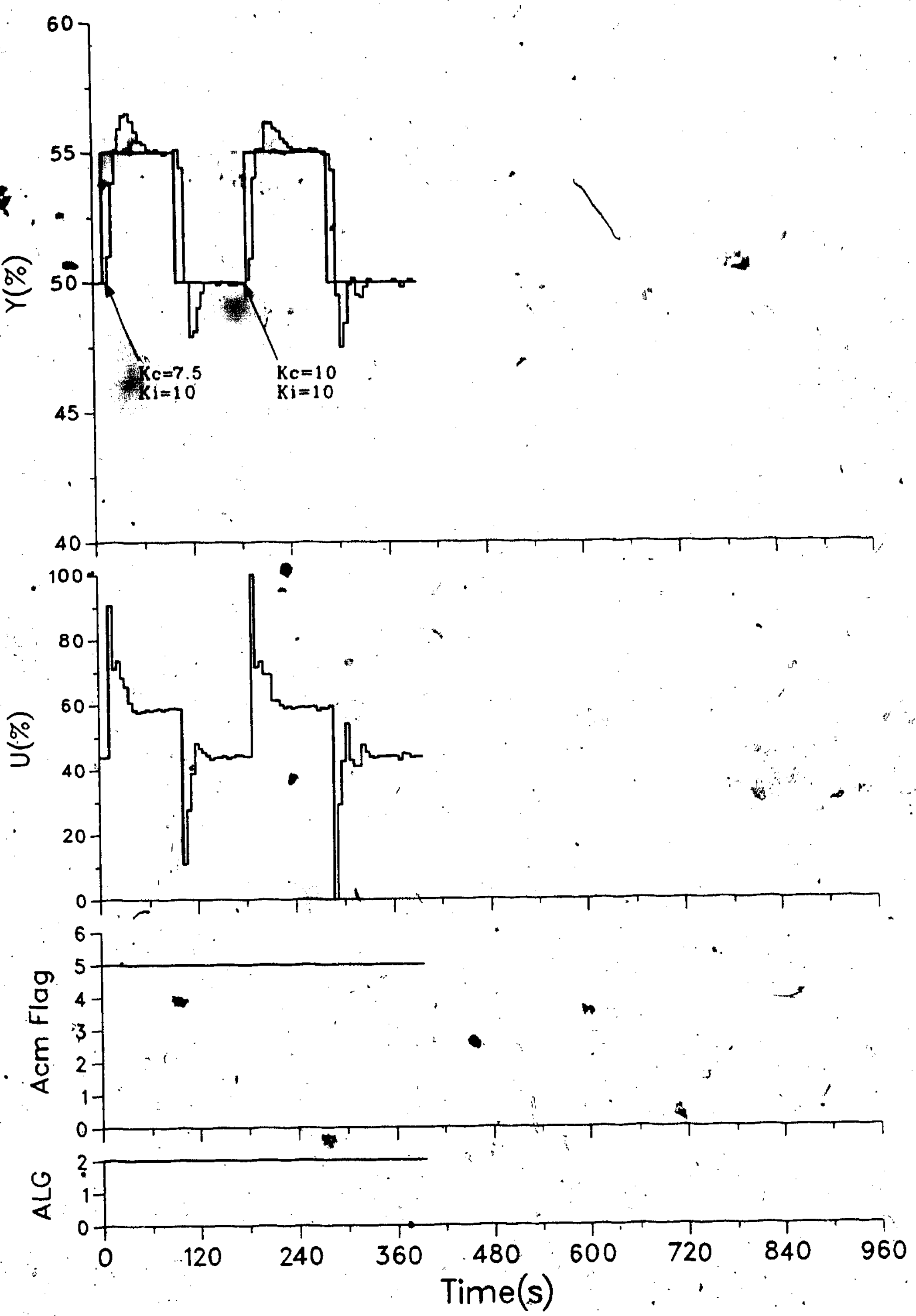


Fig. 8.7 Predictive PI Control - Effect of Adjusting Kc on Servo Performance.

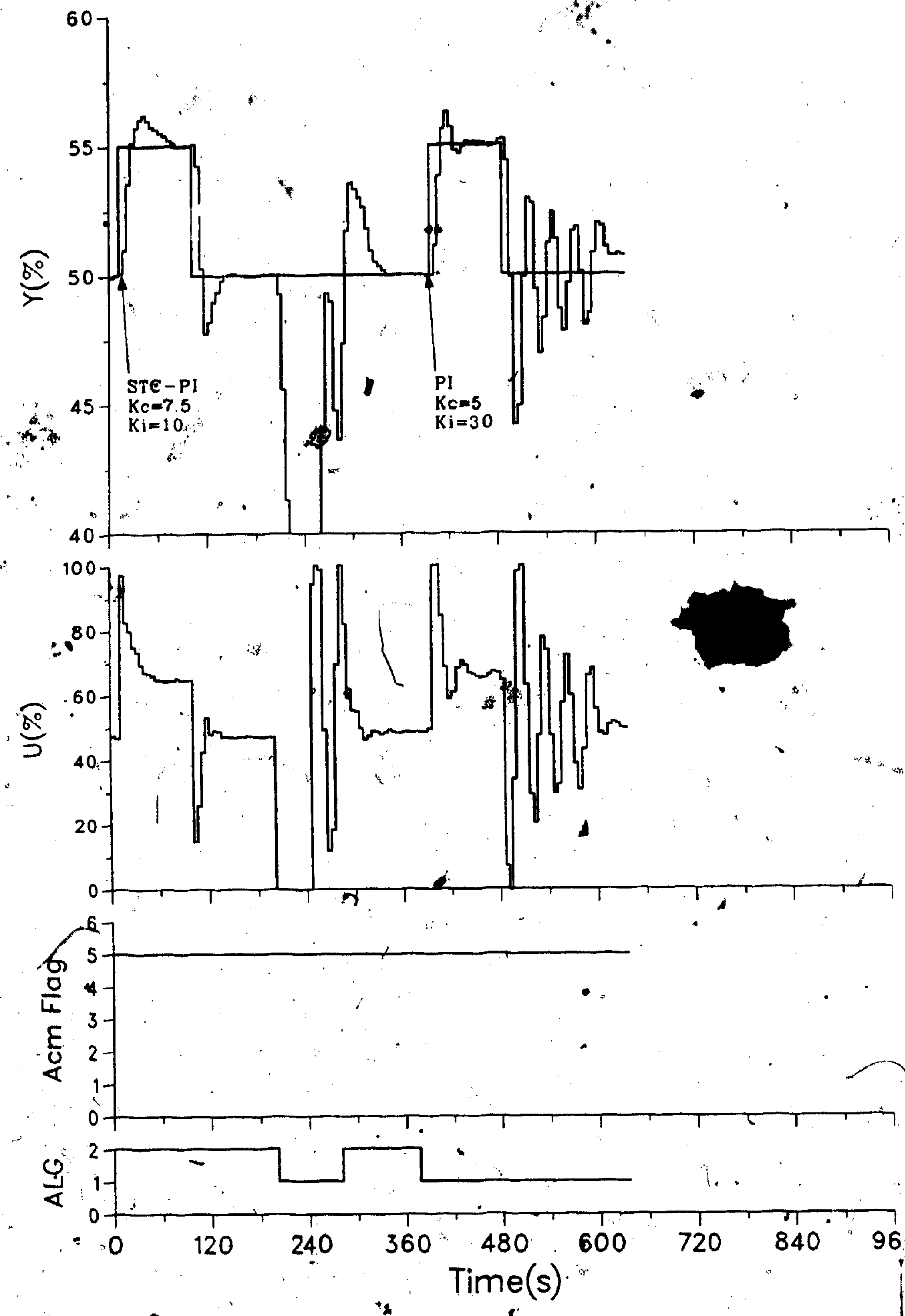


Fig. 8.8 Comparison of Conventional PI and Adaptive Predictive PI Control Performance.

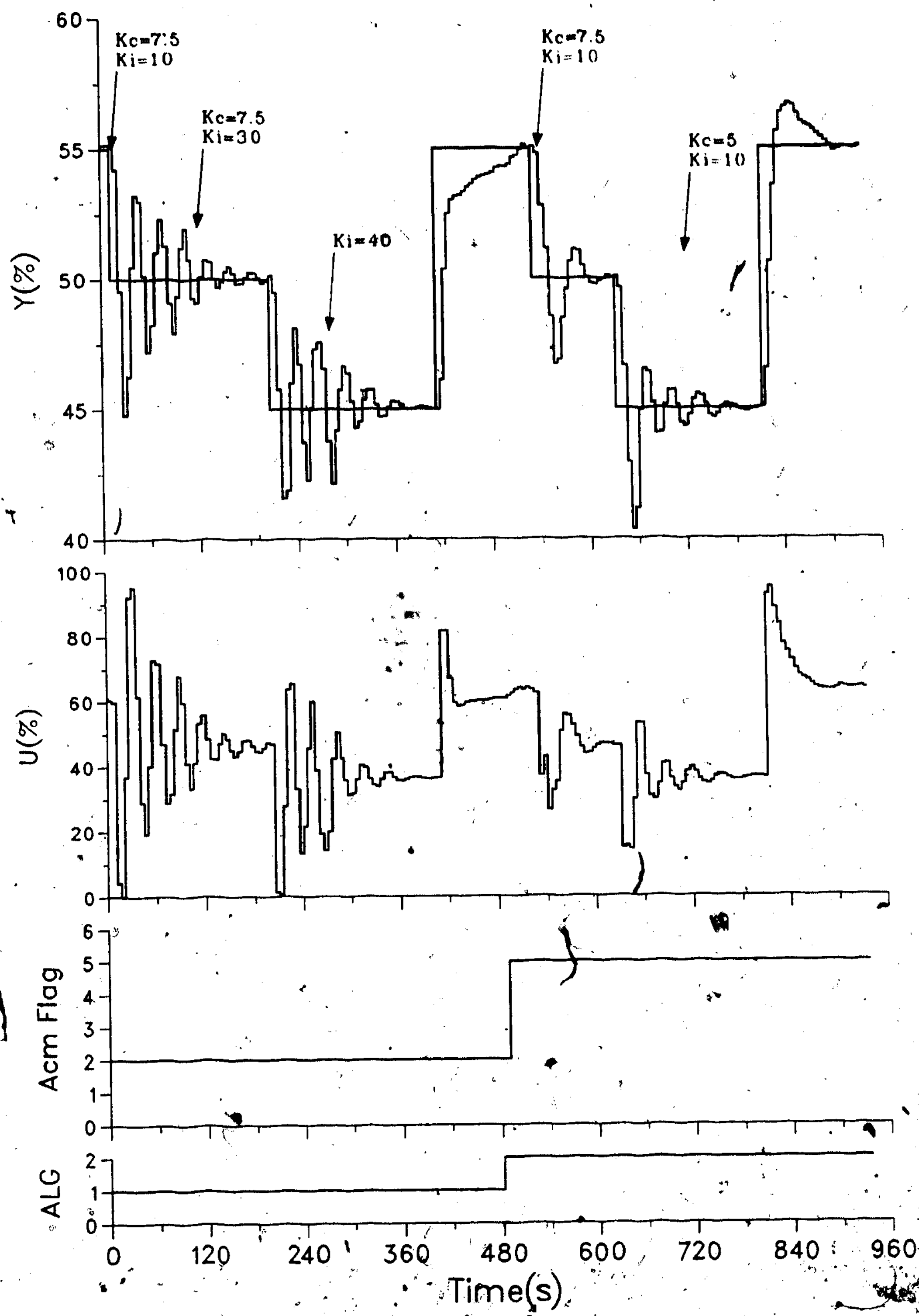


Fig. 8.9 Effect of Process Nonlinearities on Closed Loop Performance - PI vs. Predictive PI Control

compensates for process nonlinearities and changing dynamics. The use of predicted error results in less oscillatory control when compared with conventional error driven PI control. The use of predicted error is analagous to using derivative action. For conventional PD control with derivative acting on the measurement only:

$$u(k)=Kc\text{middot>}[e(k)+Kd[y(k-1)-y(k)]]$$

With proportional STC ($Q=1/\lambda$) acting on predicted error:

$$u(k)=1/\lambda[w(k)-\hat{y}(k+d)]=1/\lambda[w(k)-y(k)+[y(k)-\hat{y}(k+d)]]$$
$$=1/\lambda[e(k)] + 1/\lambda[y(k)-\hat{y}(k+d)]$$

For $Kc=\frac{1}{\lambda}$ and $d=1$, the GMV is equivalent to a classical proportional derivative controller albeit $\hat{y}(k+d)=f(u(k))$. The derivative term in GMV is based on predicted changes in process output rather posterior estimates.

8.2.6 STC is NOT a Closed Loop Adaptive Controller

Results with STC and Q=1/PI also indicate that there is no adaptation of controller parameters based on closed loop performance specifications. Although the control law is based on the minimization of 3.16, once Q is non zero GMV is no longer minimum variance control and the effect of Q on closed loop performance can not be predicted for unknown processes. The proper choice of Q is process dependent and for nonlinear processes also dependent on operating point. Regions with high static process gain require higher penalty (lower gain) on u(k) in order to maintain overall closed loop gain.

The previous results clearly demonstrate how on line adjustment of Q affects the characteristics of closed loop transient responses to set point changes. Through specification of gain and integral time, Q can be used to maintain stability of the closed loop, provided that predicted outputs are asymptotically correct. These statements clearly suggest a need for a proper adaptation loop that will adjust something in the STC's control law of 3.18 so that user specified closed loop performance specificiations can be realized. The present "adaptive" mechanism attempts to give accurate predictions, $\hat{y}(k+d)$ by adjusting $\theta(k)$ but does not provide any kind of closed loop adaptation.

The results of Figure 7.6 demonstrate how Q can be adjusted using heuristic knowledge to tune corresponding PI parameters. Figure 8.10 illustrates the proposed structure for a truly closed loop adaptive STC. Using such a design, a control engineer can implement a closed loop adaptation scheme of his choice. Given desired closed loop performance specifications and initial control parameters, the adaptive mechanism must compare current performance with desired. Based on the difference, the current control parameters are adjusted so that desired performance is achieved. When P=R=1

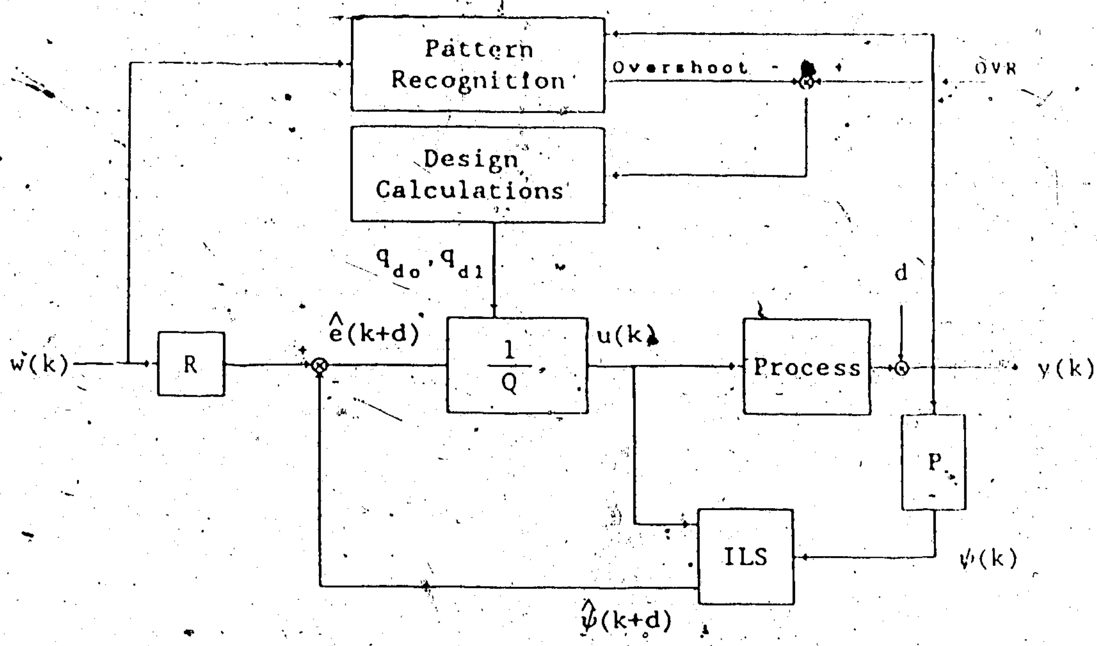


Fig. 8.10 Adaptive Predictive Control With Closed Loop Self-Tuning

and 1/Q=PI, this is equivalent to adjusting classical PI parameters to obtain the desired closed loop performance.

8.3 Effect of Set Point Filtering on Closed Loop Performance

Although set point filtering has no effect on closed loop stability it can be used for servo model following. In practice, step changes in set points are rarely possible. Controlled variables are often inputs to downstream processes and vigorous control action is not acceptable. Fast transitions in operating level therefore upset downstream processes.

Results demonstrating the practical use of filtering are presented in Figure 8.11. For a fixed set of PI constants (Kc=5.0/Ki=10.0 s.) several set point transients using different set point filter time constants are shown. As filtering was increased, changes in control action became smoother, approaching steady state levels exponentially rather than with saturation and oscillation. For τ_{sp}=30s the controller attempts to drive the process measurement to its new set point in $4 \cdot \tau_{sp}$ seconds or two minutes. The use of set point filtering prevents large errors from entering the closed loop, resulting in smoother control and transients.

At t=7[illegible]5s the controller gain was increased to 10.0 just prior to a -10% change in set point (τ_{sp}=15s) to 40%. The transient response was oscillatory about the set point of 45% demonstrating that the set point filter had no influence on steady state performance. With the proper choice of controller constants, exponential filtering gives the operator a means of specifying degree of damping for servo changes. Also since the GMV attempts to minimize the error $[w(k)-\hat{y}(k+d)]$, set point filtering is an indirect method of achieving the desired closed loop response to a set point change.

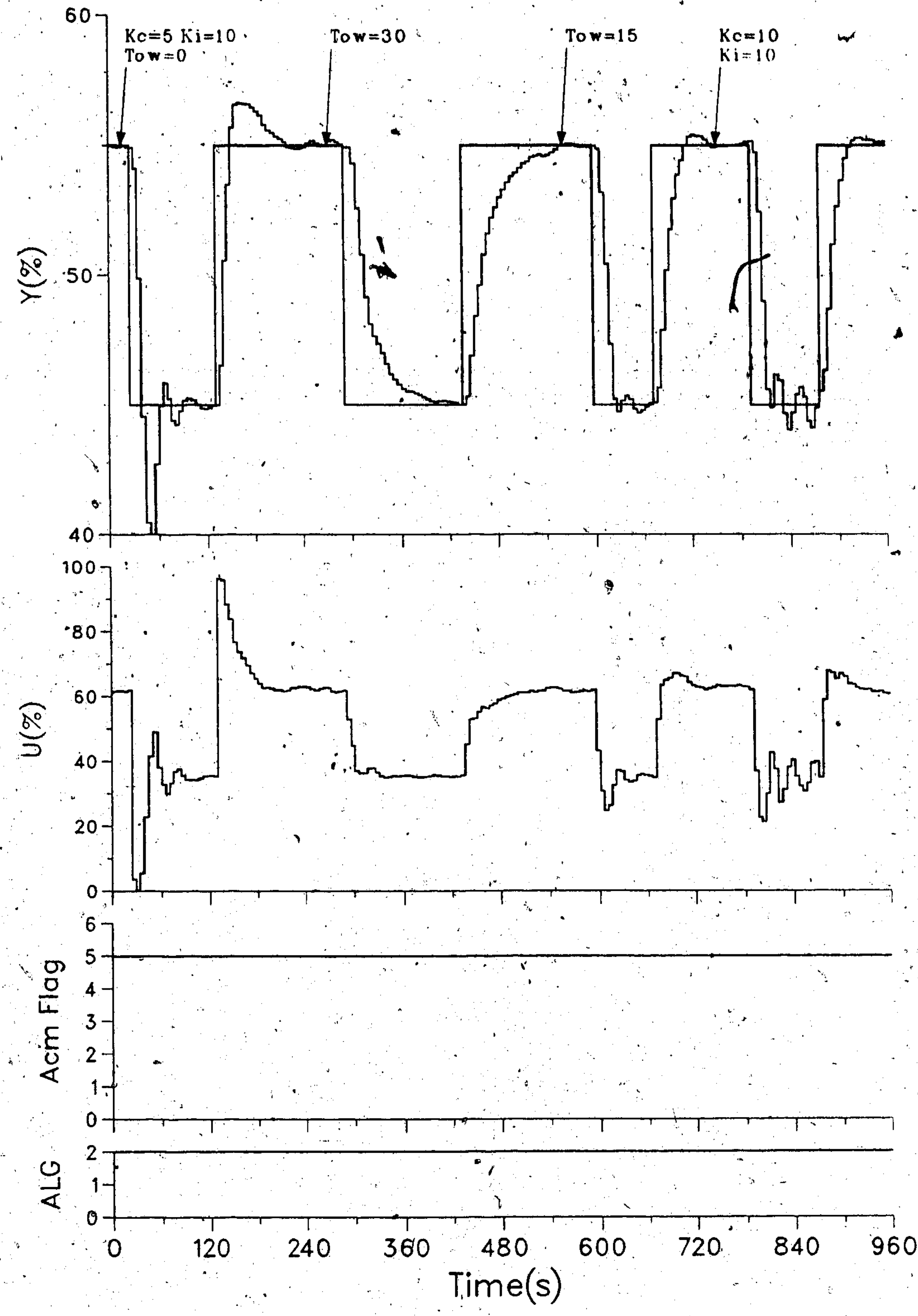


Fig. 8.11 STC - PI mode - Effect of Set Point Filtering on Closed Loop Performance

8.4 Selection of On-Off Criteria Limits for ILS

Improved Least Squares has incorporated two important On-Off criteria into its constant trace algorithm (see Chapter four). To make full use of these On-Off tests the user must specify an allowable upper limit on condition number of P(k) (Condp) and a lower limit on $\|P(k-1)x(k)\|$ (Iota) which is related to information content in measured I/O signals.

Selection of appropriate values for these limits is not intuitively simple, but can be accomplished by trial and error. In this work, Condp and Iota were determined using the following procedure. Using either GMV or conventional PI control, the predicted output terms were estimated using RLS with UDU^T factorization and Fortescue's variable forgetting factor. The same normalization on y(k) and x(k) was performed as in ILS and values for $\|P(k-1)x(k)\|$ and Cond[P(k)] calculated and displayed to the operator's console. Concurrently, the plotting task was used to display a graphic trend of parameter estimates so that the operator could observe how $\theta(k)$ behaved during closed loop operation. Following a user specified set point change, the calculated values for $\|P(k-1)x(k)\|$ and Cond[P(k)] were monitored. When the process lined out at the new set point, the mean deviation data in x(k) quickly approached zero so that $\|P(k-1)x(k)\|$ suddenly decreased in value. The condition number of P(k) tended to remain at a constant level provided that the process measurement was not corrupted by large amounts of noise. Limits were then selected based on the values calculated prior to the sudden drop in $\|P(k-1)x(k)\|$ at steady state. The value of Tr[P(k)] was initially set to 1000I for RLS identification.

Figure 8.12 illustrates the dynamic response of calculated values of $\|P(k-1)x(k)\|$ and Cond[P(k)] during closed loop conventional PI control. Values for Condp and Iota were chosen at 500 and 0.001 for subsequent use in ILS. The value of Tr[P(k)] was held at 2.0 (0.5I) for subsequent runs unless otherwise specified.

Figure 8.13 shows the resulting performance of ILS using these limits for a series of set point changes. With the chosen limits for $\|P(k-1)x(k)\|$ and Cond[P(k)], identification was suspended during periods of low excitation, thus preventing drift in $\theta(k)$ due to noise or high frequency dynamics.

The ILS algorithm maintains Tr[P(k)] at a user specified value by adjusting $\lambda(k)$. If the calculated value of $\lambda(k)$ is also used for mean estimation of $\bar{u}(k)$ and $\bar{y}(k)$, the chosen limits for On-Off criteria can affect closed loop performance during "steady state" operation. If Iota is specified as a very small number ($\simeq 10^{-9}$), and Condp very large ($\simeq 10^{8}$) then the update of $\theta(k-1)$ will not be turned off. As a result, the calculated values of $\lambda(k)$ approach values of zero so that Tr[P(k)] will be kept constant. From equation (4.32), the actual calculation of u(k) is a function of $\lambda_u(k)u(k-1)$ or $\lambda(k)u(k-1)$ if $\lambda_u(k)=\lambda(k)$. As a

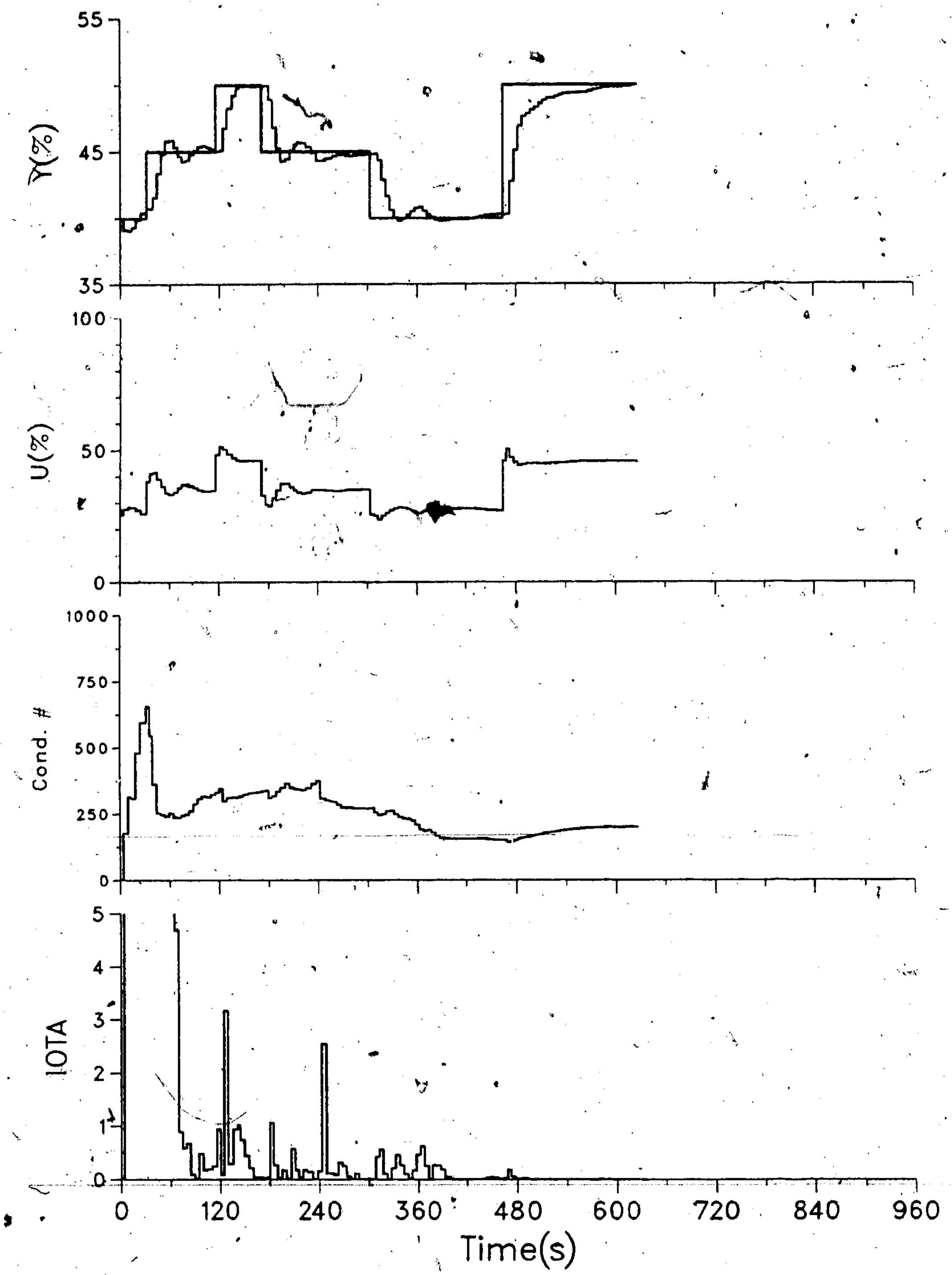


Fig. 8.12 Calculated Values of [P(k-1)x(k)] and Cond[P(k)] During Conventional PI Control With Ordinary RLS Prediction

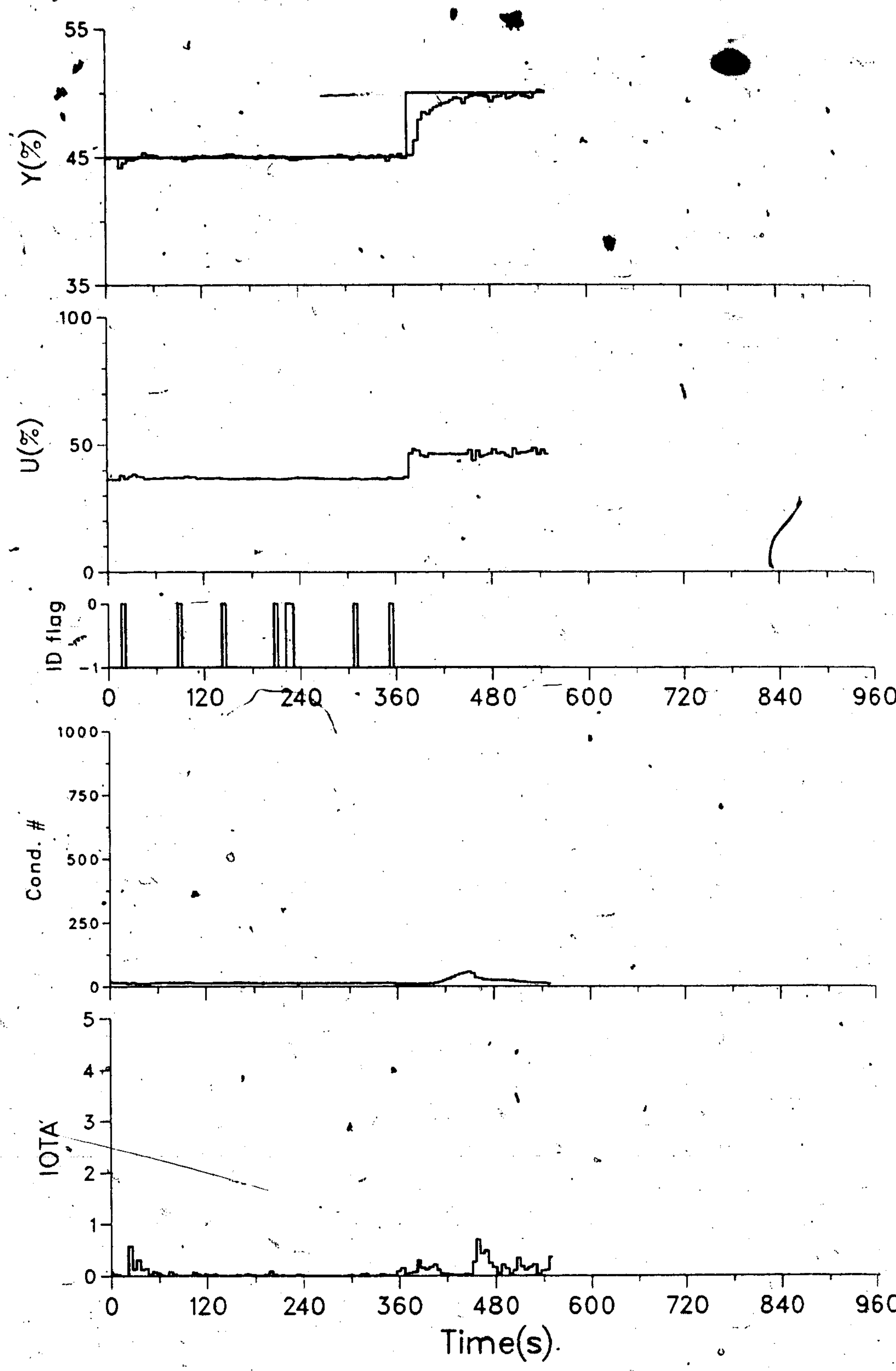


Fig. 8.13 Use of ILS and Demonstration of On-Off Criteria (Iota=0.001 Condp=500)

result, u(k) will be "perturbed" by λ(k)u(k-1) giving unsatisfactory control. Figure 8.14 demonstrates this problem. The transient response of y(k) and u(k) are shown. The values of Iota and Condp were 10^{-3} and 10_3. At steady state (t=325s) the perturbations in u(k) appeared. The problem was corrected by decreasing the desired value for Tr[P(k)] at t=415s. The value of λ(k) is also plotted. Following this, parameter identification was suspended due to a lack of excitation and λ(k) was fixed. At t=560s, Tr[P(k)] was inflated to 105.0, resulting in oscillations of λ(k). The controller eventually became unstable due to the effects of λ(k) on control calculations.

In this work, the forgetting factors for mean estimation, $\lambda_y(k)$ and $\lambda_u(k)$, were specified by the user (default 0.75) and not a function of λ(k) from the ILS algorithm.

8.5 Open Loop Identification

The initial identification option minimizes the amount of knowledge required for startup and closed loop operation. Figure 8.15 illustrates a typical test followed by several set point transients to demonstrate the resulting controller performance. The initial phase uses the Cohen and Coon method for calculating initial estimates for Kc and Ki following the estimation of Kp, τ_p T_s and τ_d. The initial values for Kc and Ki are based on 25% overshoot and $\frac{1}{4}$ decay.

The recommended sampling interval based on the estimated time constant, τ_p, gives results that compare with those

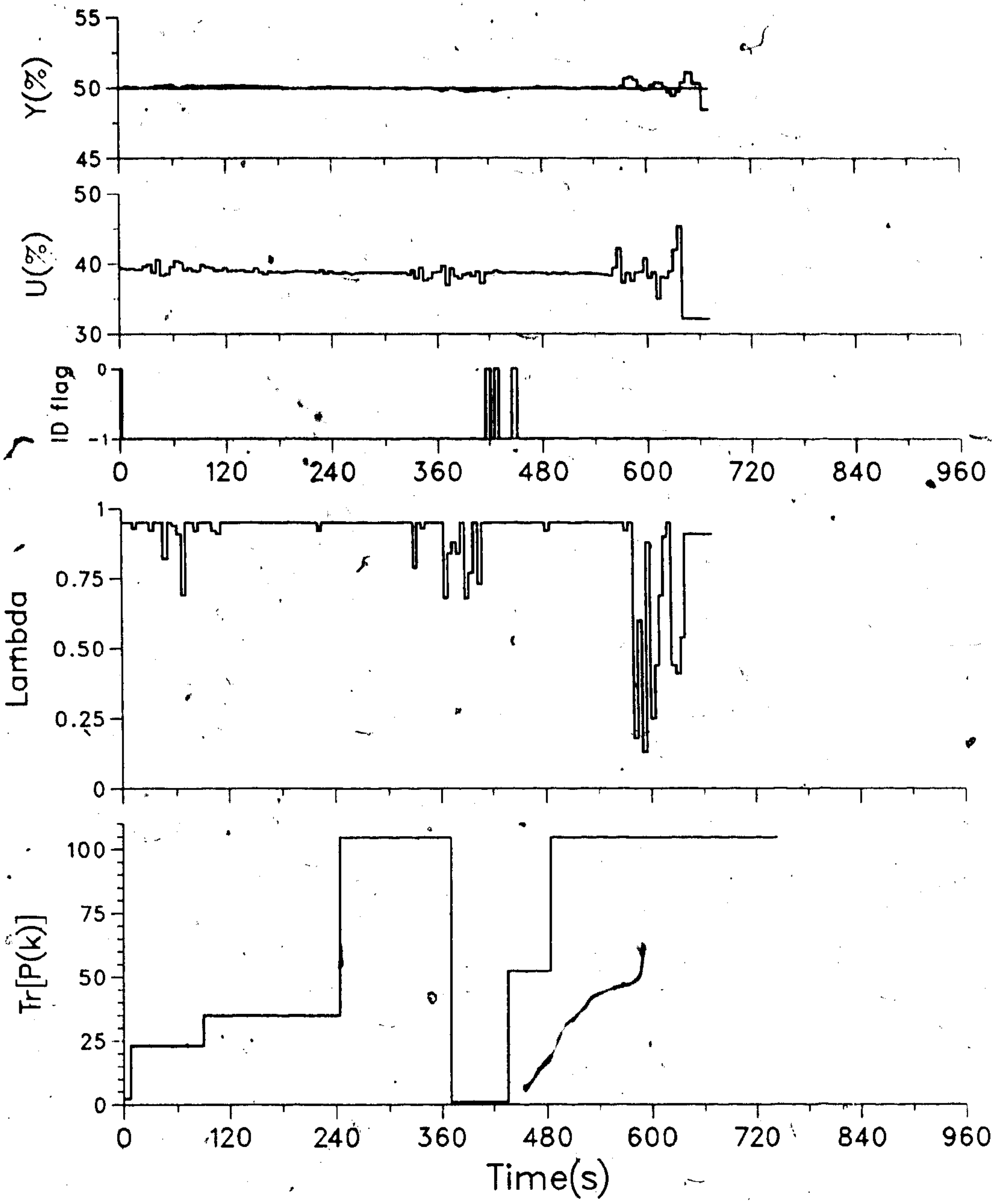


Fig. 8.14 Effect of Incorrect Specification of Limits on $P(k-1)x(k)$ and Cond[P(k)] When $\lambda(k)$ is Used for Mean Estimation

that had been arbitrarily used previous to the implementation of the initial test. The minimum allowable sample time is one second. This value increases to two seconds when all of the A^3's features are used.

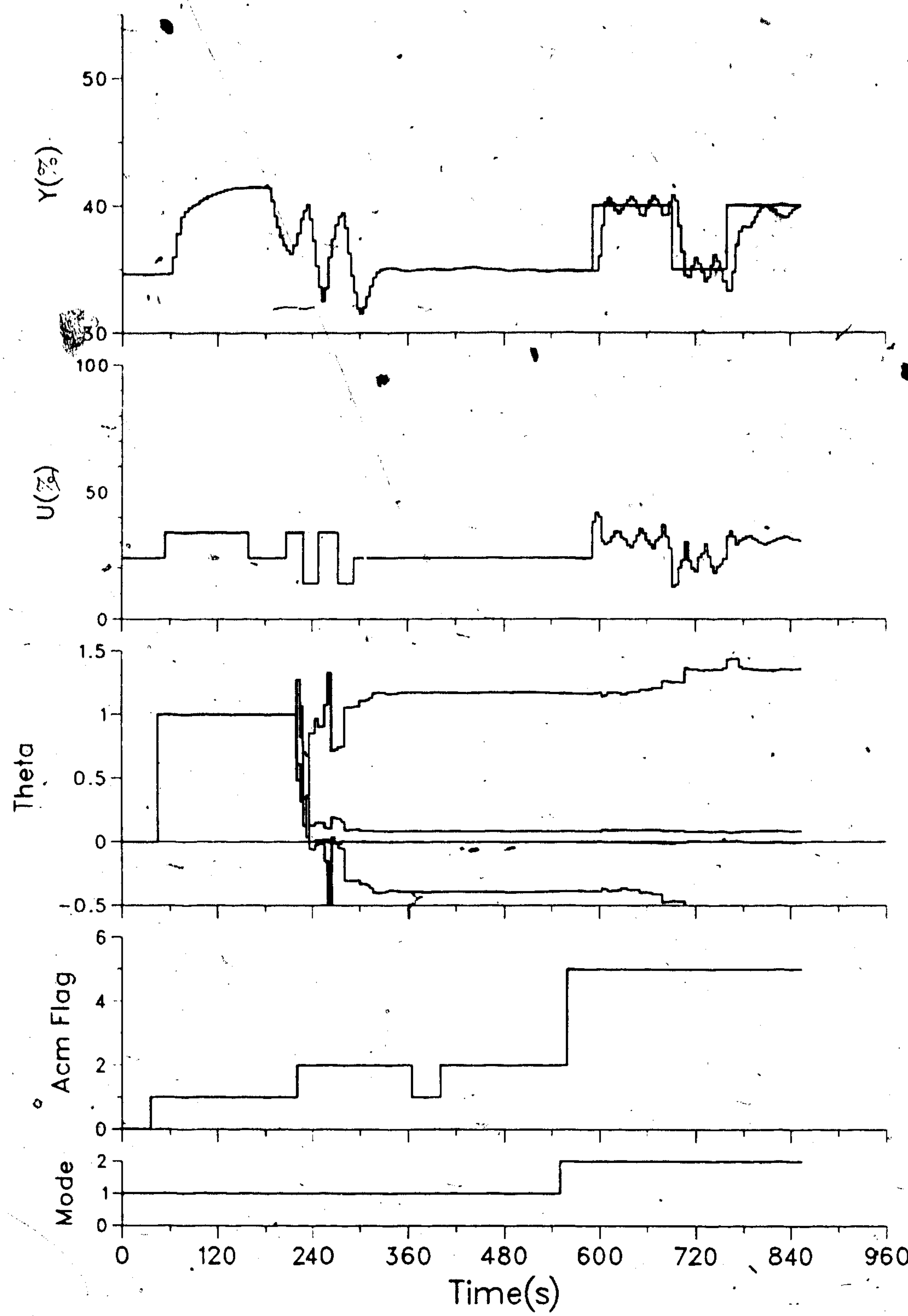


Fig. 8.15 Open Loop Identification Followed by Closed Loop Control

To appreciate the utility of the initial identification test, refer to Figure 8.16 in which the STC is commissioned with the same initial PI constants but no previous open loop estimation of predicted outputs $\hat{y}(k+d)$. Control is initially unstable due to poor prediction. As prediction estimates improve the controller manages to bring the process output to its desired level.

The use of an inverse PI structure for Q has allowed for its calculation based on well known, commonly used techniques for obtaining initial PI constants. Most applications in the past have used much more arbitrary procedures in which parameters in Q are directly modified and Q kept simple in structure.

Figure 8.15 demonstrates that GMV can give good control with a minimal amount of supervision for startup. Because the initial identification results are based on dynamic open loop response data, results were consistent and repeatable for a given operating point.

With the initial test, the operator must specify only two pieces of information: the magnitude of the step change in controller output and the maximum desired deviation from initial steady state for the process measurement. The specification of exact model structure is not critical for "non minimum variance" control since calculated controller outputs are no longer a function of $1/g_0$ alone.

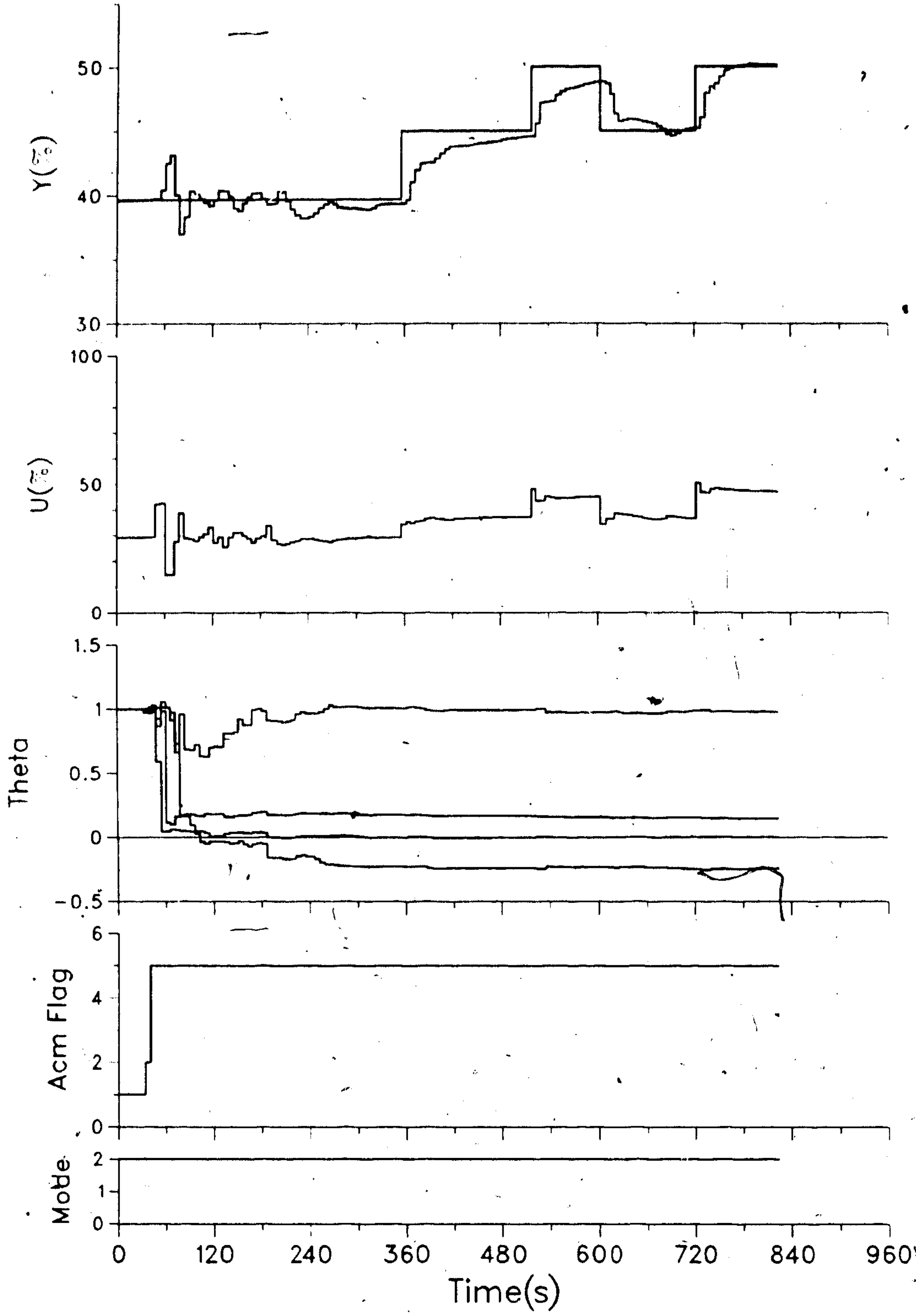


Fig. 8.16 STC Startup with no Initial Identification Period

8.6 Closed Loop Performance

8.6.1 Typical Controller Performance

For experimental purposes, controller performance must be defined. Strictly speaking, GMV's performance criteria is the minimization of 3.16. This cost function minimization is realizable only if perfect modelling is achieved. Furthermore even with perfect modelling the minimization is guaranteed in the discrete domain alone.

With the use of filtering on u(k), it becomes more difficult to interpret the performance criteria from a practical point of view. Q is used to offset the destabilizing effect of modelling errors. With the presence of modelling errors during transients, the only thing that J guarantees is that if $\hat{y}(k+d) \rightarrow y(k+d)$ and $Q(z^{-1})|_{z=1}=0$ then y(k+d) will equal w(k) at steady state. It makes no guarantee as to how set point error will be rejected.

More meaningful performance criteria are needed that will provide the basis for adaptation of adjustable parameters within 3.18. The criteria should uniquely determine the characteristics of the closed loop transient behavior to specific process upsets such as disturbances or set point changes. For example ISE is often used as a measure of performance but can not alone be used as the criterion for adaptation. Both an underdamped and overdamped transient response can give the same ISE values.

For the A^3 control system, overshoot of process measurement following a set point change was used as a measure of performance, which was then used to adjust controller gain on-line. A user specified maximum allowable damping was used as a constraint to override the overshoot specification. This prevented sustained oscillations in y following set point transients. Figure 8.17 shows a typical closed loop response to an unfiltered step change in set point for the A^3 control system. The cost function minimization provides the necessary second condition that will specify the time scale of the closed loop system through the choice of set point filtering .

8.6.2 Performance of the Adaptive Algorithms

As shown in Figure (8.10), there are two adaptation schemes running during normal closed loop operation. The first is the recursive estimation of $\theta(k)$ used for prediction of $y(k+d)$. The second is the scheme used to adjust parameters within the 1/Q control law based on the difference between desired and the latest measured overshoot.

The RLS scheme is essentially open loop identification while the second based on measured $w(k)$ and $y(k)$ is closed loop self-tuning. The estimation of predicted outputs is not based on the feedback of any closed loop performance criteria.

The main objective of the inner adaptive loop is to accurately predict the dynamic behavior of the process (a

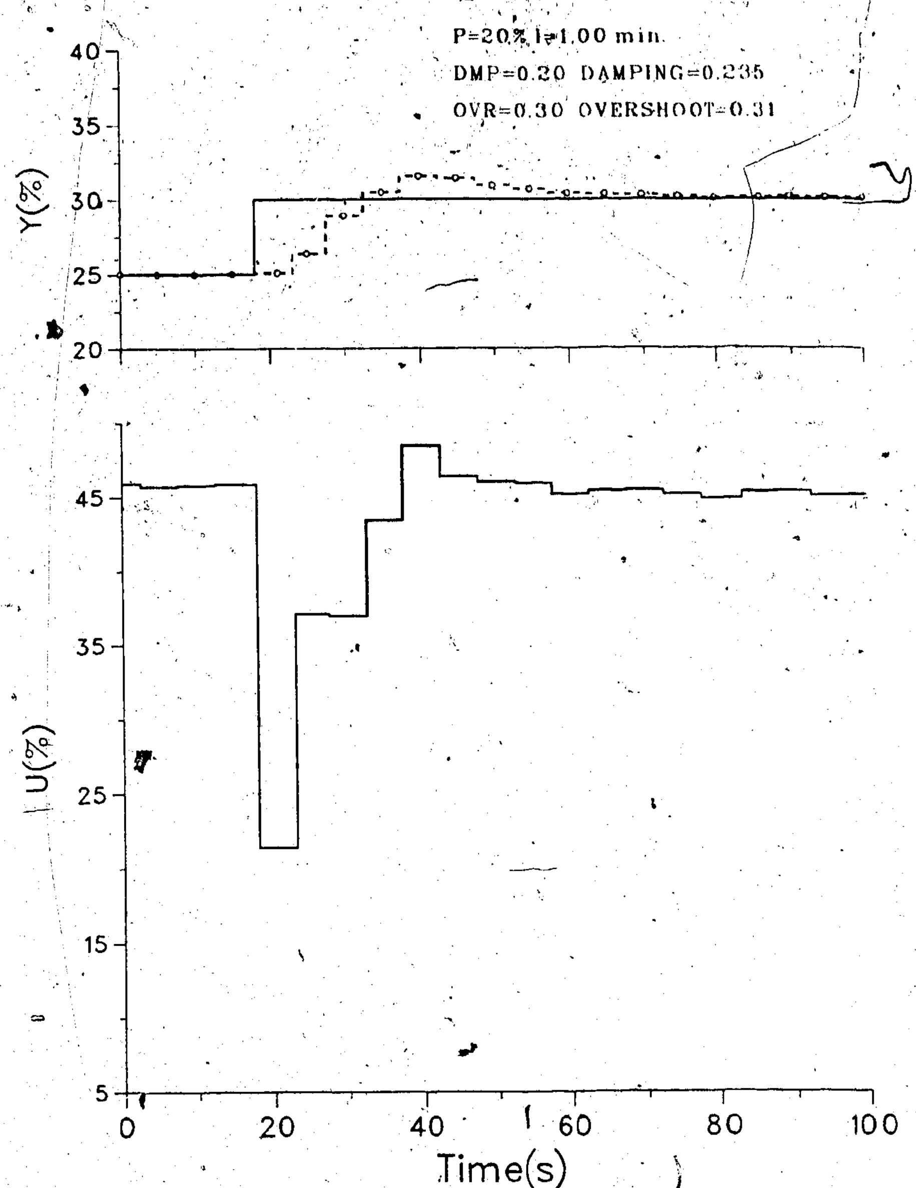


Fig. 8.17 Typical Closed Loop Response of GMV Control to A Step Change in Set Point

"u-y model"). With this implicit scheme, the values of parameters within $\theta(k)$ converge to some set, not necessarily the true values, so that $\hat{e}_0$ does not drift or oscillate. From (3.54), variations in $\hat{e}_0$ due to numerical problems will create undesirable fluctuations in u(k).

8.6.3 Automatic Adjustment of Kc $\rightarrow Q(z^{-1})$

A second adaptation loop based on closed loop overshoot and damping was implemented in an effort to make GMV directly self-tuning. The purpose of this section is to demonstrate that automatic adjustment of $Q(z^{-1})$ can be used to deliver user specified closed loop performance.

With the inverse PI control law formulation and some heuristic tuning rules, recommended controller gains were used to automatically adjust $Q(z^{-1})$ following set point changes. Using an underdamped, three peak characterization for closed loop responses to set point changes, a simple algorithm was designed to adjust Kc based on the difference between desired and measured overshoot. As well, a user specified limit on damping (as defined by Foxboro) was used as an overriding constraint on Kc. The performance of this tuner is documented in this section.

<u>Convergence of Kc</u>

Given initial control constants Kc=1 and Ki=30s., and steady state operation at 30% of span, the ability of the A^3's tuner to give a specified overshoot of 0.30 with a damping constraint of 0.10 was evaluated. As shown in Figure 8.18 the speed of response to set point changes improved with each tuning transient. After ≅5 transients, the controller gain had converged to a value of 6.0 delivering an overshoot of 25-30%. Subsequent set point changes about

27.5% resulted in constant controller gain. Figure 8.19 contains the trajectory of Kc as a function of tuning transient.

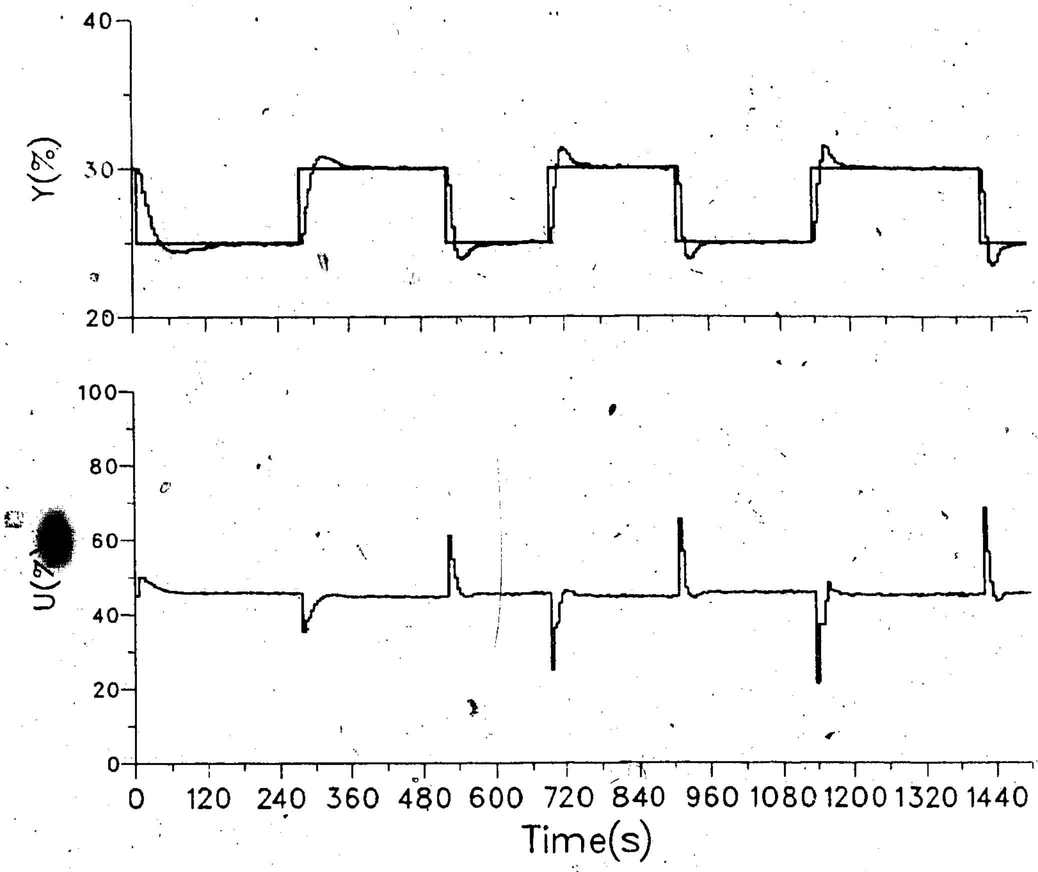


Fig. 8.18 Convergence of Specified Overshoot About 52.5% for the A^3 Tuner - OVR=0.3

Tracking Ability

An adaptive controller must be able to adjust controller parameters in response to changes in process

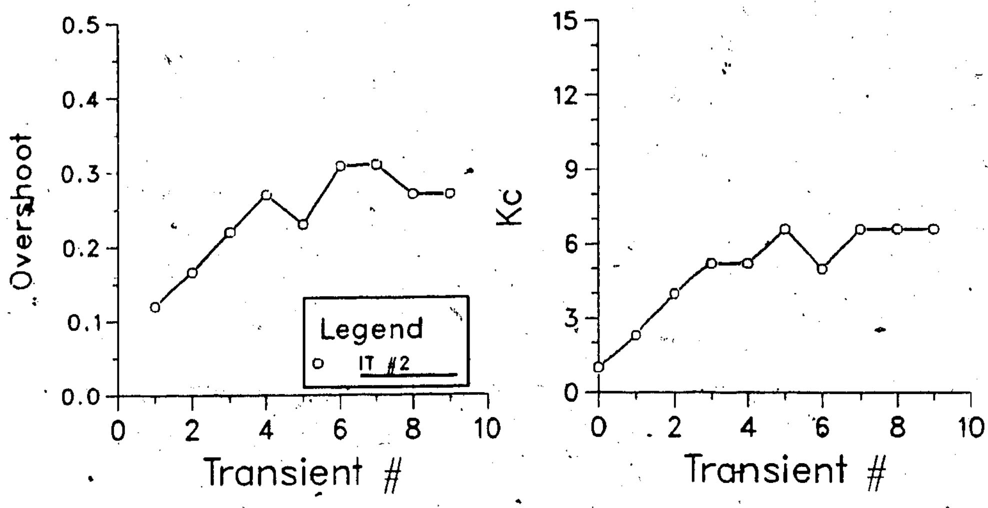


Fig. 8.19 Trajectories for Adapted Kc Demonstrating Convergence

dynamics. For an overshoot specification of 0.30, the GMV was operated about the 27.5% level to obtain converged Kc and $\theta(k)$. A series of +5% set point changes were then implemented in order to operate at a set point of 45-50%. Figure 8.20 illustrates the closed loop performance for the series of set point changes.

The effects of the process nonlinearities on the closed loop performance are clear. The A^3's self-tuner had to increase the controller gain after each transient in an effort to maintain the desired overshoot. The overall adjustment of controller gain was not as critical as in the Exact controller's case due to the advantage of using predicted error. Refer to Figure 8.21 for a trajectory of controller gains for the series of set point changes.

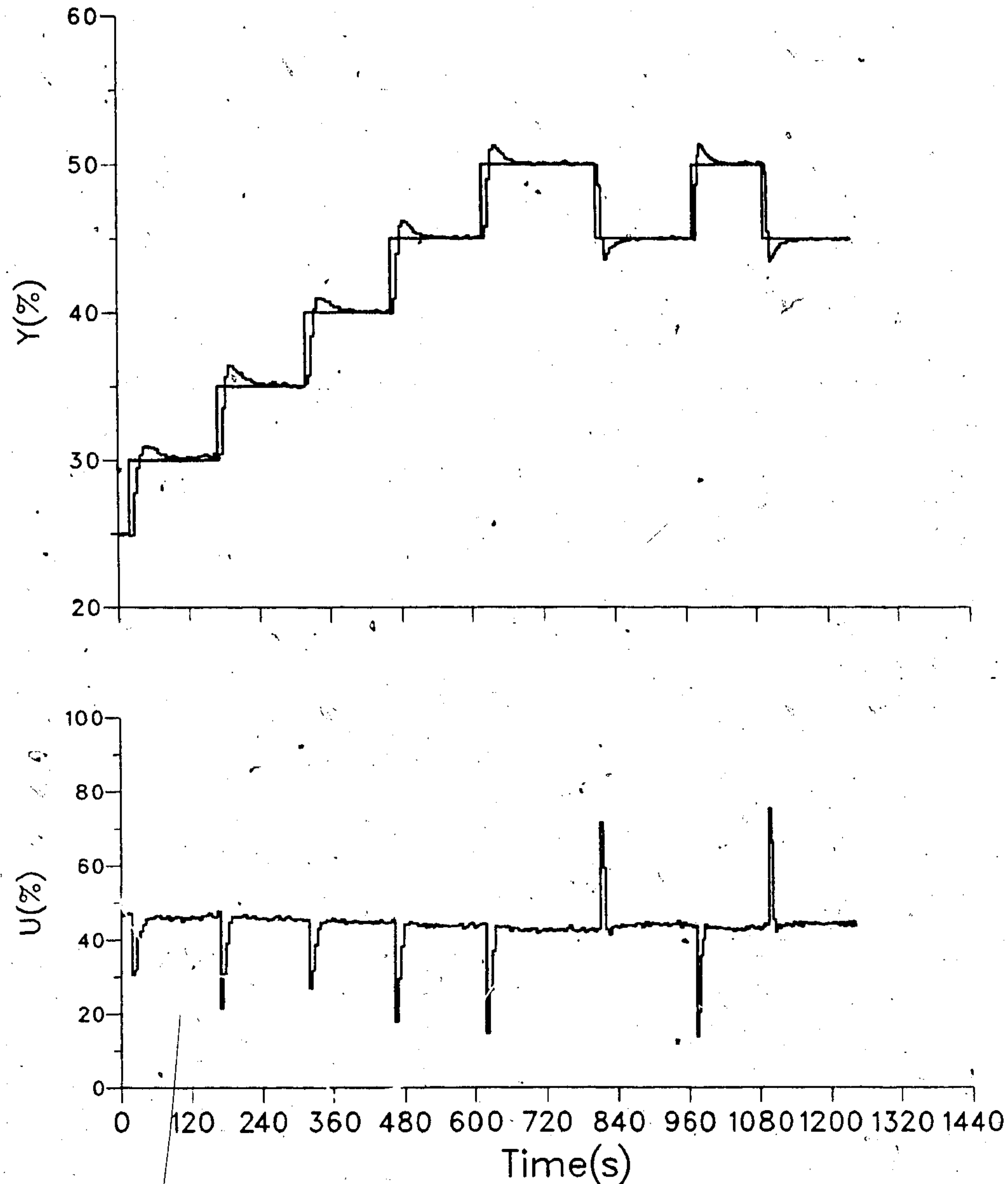


Fig. 8.20 Tracking Ability of the A^3 Controller (OVR=0.30)

Tailoring of Closed Loop Response

An adaptive controller must be able to adjust its control law parameters in response to changes in operator specified performance criteria. For user specified changes

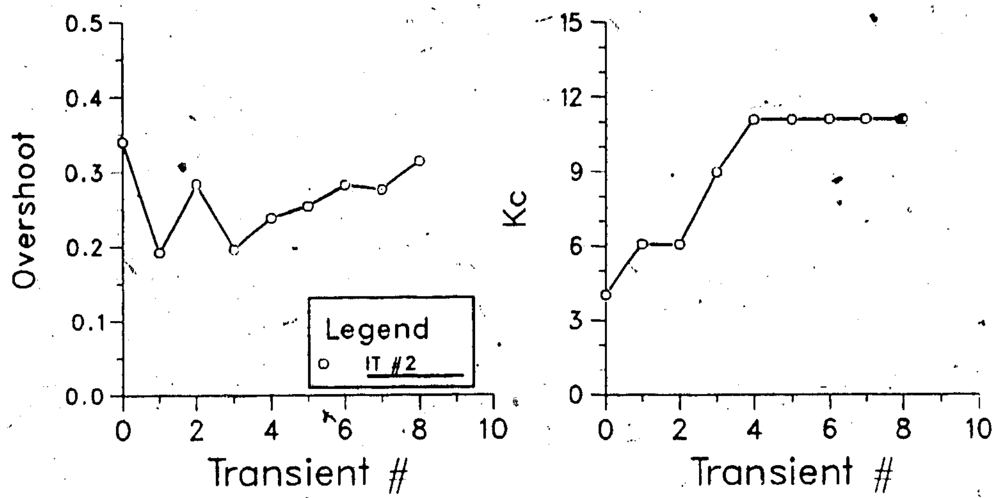


Fig. 8.21 Trajectories for Overshoot and Kc for a Series of +5% Set Point Changes to 52.5.% (OVR=0.30 DMP=0.10)

in overshoot at a given operating level, Figure 8.22 demonstrates the A^3 tuner's ability to adjust controller gain so that the specified overshoot is achieved.

Initially, for a specified overshoot of 0.30 and damping limit of 0.25, the controller gain was 6.5. At t=60s. those limits were changed to OVR=0.15 and DMP=0.1 and several set point perturbations used to generate transient responses. The tuner decreased the controller gain to 2.21 after 7 transients and subsequent set point changes had overshoots of ≃0.15. Figure 8.23 contains the trajectory for Kc for the series of set point changes following the overshoot specification of 0.0.

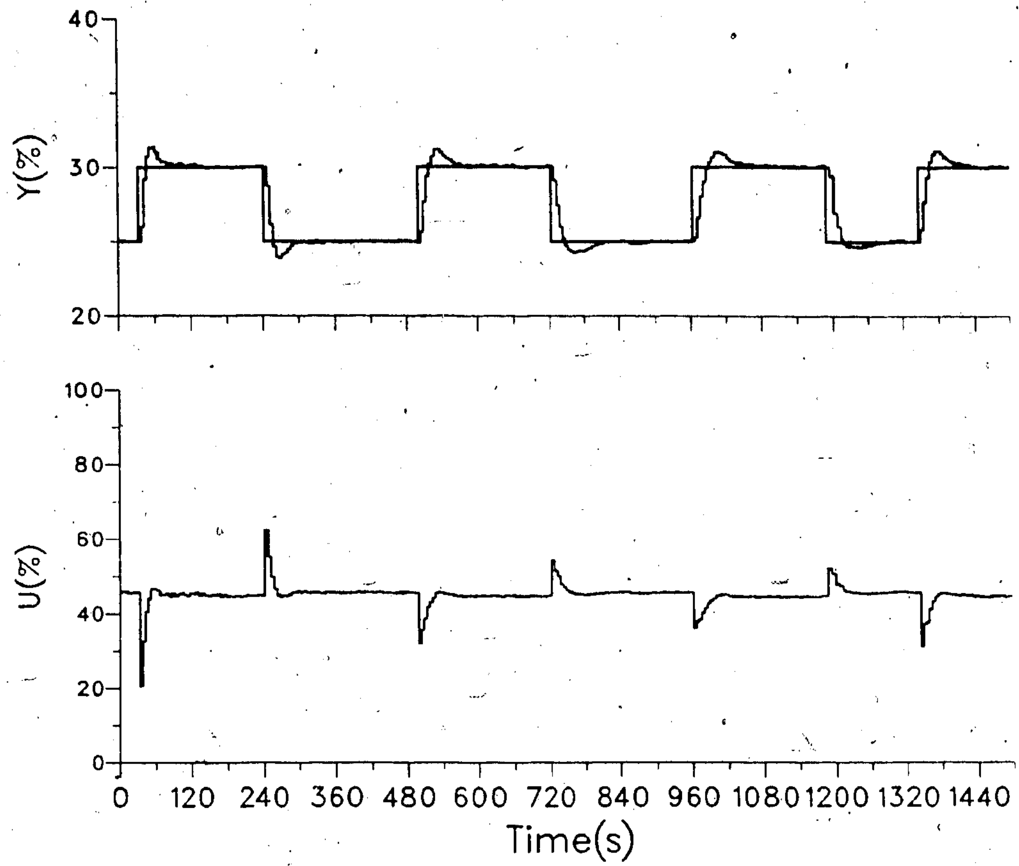


Fig. 8.22 Tailoring of Closed Loop Response Using A^3 Tuner

8.7 Evaluation Under Selected Process Conditions

8.7.1 Overdamped Processes

The specification of underdamped closed loop responses in the control of dominantly damped processes can result in the adaptive tuner giving high controller gains. For the first order liquid level controlled plant, Figure 8.24 demonstrates the gain wind up problem.

Following each transient, the self-tuner increased controller gain in an effort to force the process to

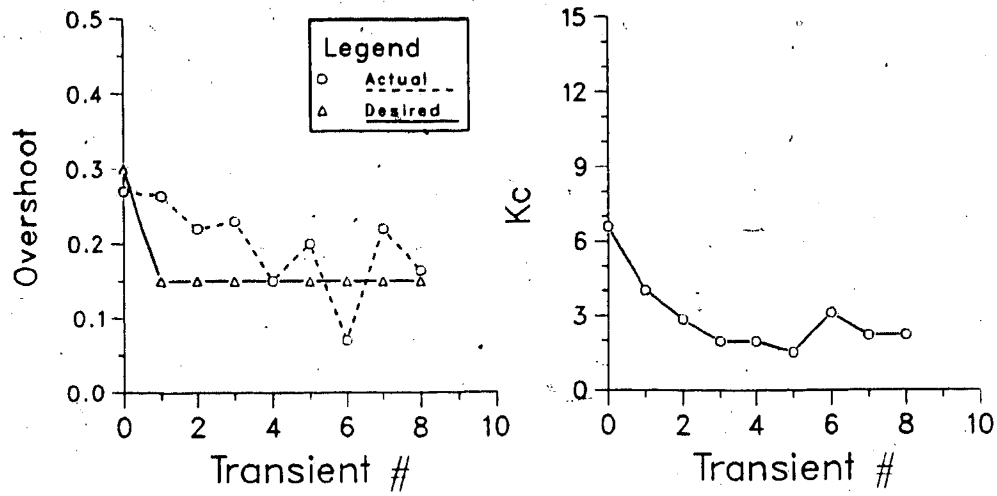


Fig. 8.23 Trajectories for Overshoot and Kc for Series of up/down 5% Set Point Changes.

overshoot on the next set point change. Control action eventually becomes almost deadbeat, oscillating between its upper and lower limits.

With the A^3 controller, this problem was solved by relaxing the overshoot specification so to OVR=0.20. The opeator can also specify overdamped closed loop responses through the choice of set point filter time constant as was shown in Figure 8.16. Figure 8.25 demonstrates how the 0.2 overshoot specification prevented the gain wind up problem. Figure 8.26 shows the trajectory of controller gain for both cases: OVR=0.5 and OVR=0.2.

8.7.2 Measurement Noise

The performance of GMV is dependent on the accuracy of predicted outputs which in turn can be adversely affected by

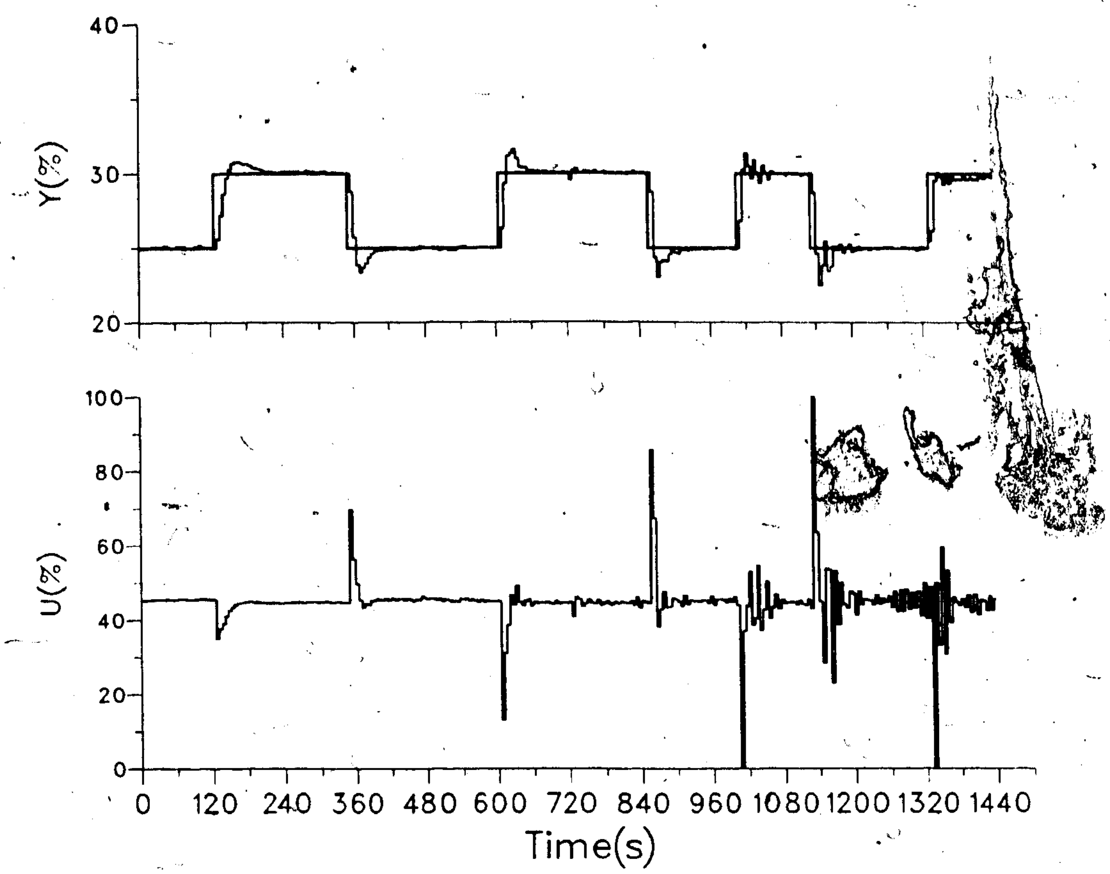


Fig. 8.24 Gain Wind-Up for GMV (OVR=0.5 DMP=0.3)

measurement noise (see Chapter 3). In this work the noise on process measurements was assumed to be random so that $C(z^{-1})=1.0$ for modelling purposes.

The effect of measurement noise is shown in Figure 8.27. Initially without measurement filtering, controller performance was slightly oscillatory due to feedback of noise (the noise generator was not used for this run). From Figure 8.28, parameter estimates were drifting a considerable amount up until t=330s when measurement filtering was enabled with a time constant of 6 seconds. The

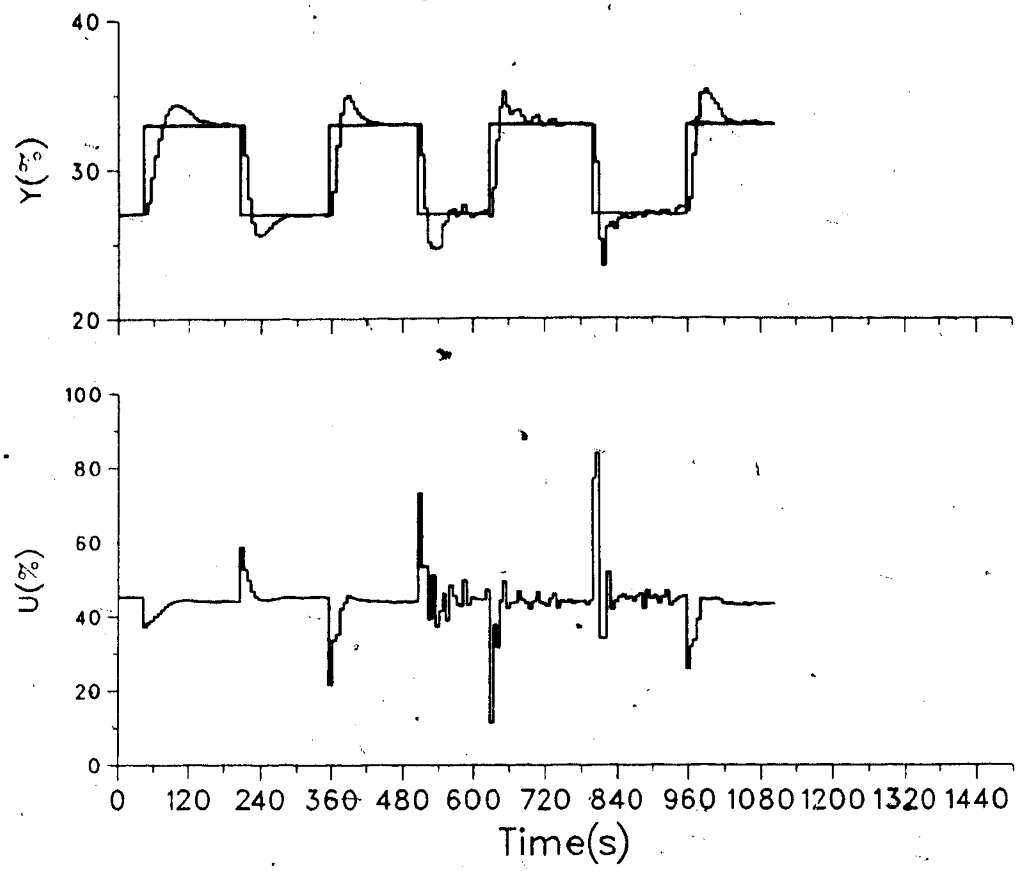


Fig. 8.25 Prevention of Gain Wind Up Through Specification of Conservative Overhoot (0.2)

filtering removed the high frequency content of the measurement signal and thus prevented the predicted outputs and $\theta(k)$ from varying unnecessarily. The result was an improvement in steady state performance for both the controller and the adaptive predictor. Parameter estimates stopped drifting and filtering enabled the On-Off criteria is ILS to take effect at steady state.

The second closed loop adaptative scheme is not sensitive to measurement noise when the user specified deadband exceeds the variance of $\psi(k)$.

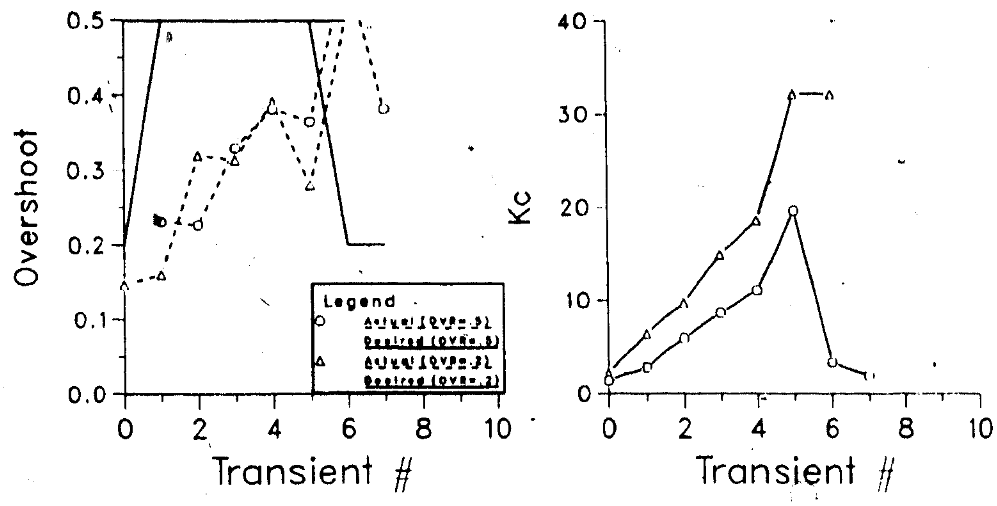


Fig. 8.26 Trajectories for Kc (OVR=0.50 and OVR=0.2)

8.7.3 Effects of Slowly Drifting Disturbances

When the A^3 controller's PI constants are well tuned, slowly drifting disturbances will not activate the closed loop adaptive scheme. These disturbances do adversely affect the quality of prediction estimates since $\theta(k)$ is adjusted based on I/O data that does not represent true u-y process dynamics. Figure 8.29 illustrates how a change in autotransformer output affects the adaptive controller's performance. At t=80s. the output was increased from 50 to 70% with the process measurement steady at 35% of span and PI constants Kc=3.0 and Ki=90.0s.

At t=345s. the controller gain was decreased to a value of 1.0 in order to remove the closed loop oscillations in u(k) and y(k) and the system stabilized.

The above results again demonstrate that the adaptive

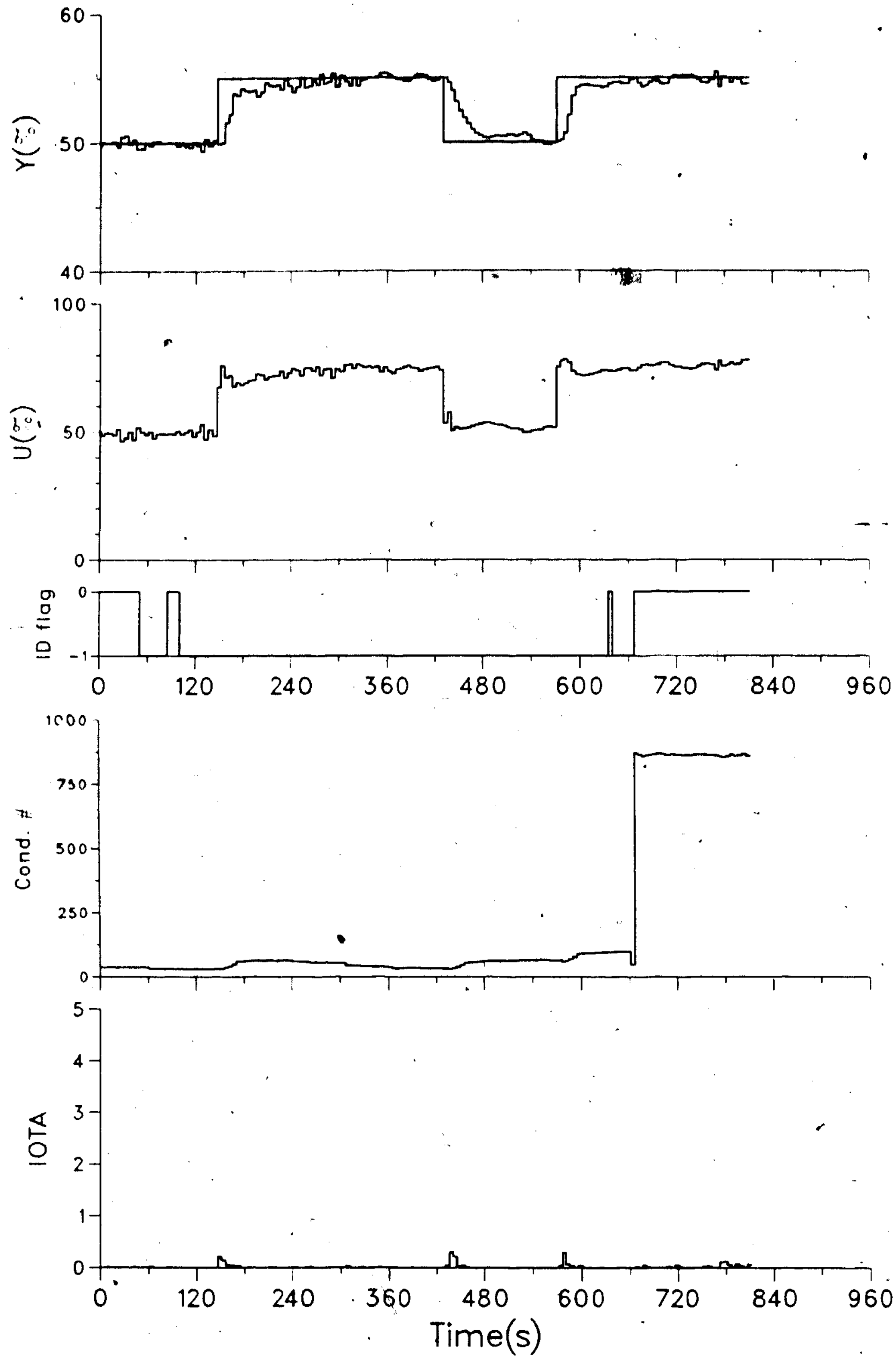


Fig. 8.27 Effect of Measurement Noise on Controller Performance

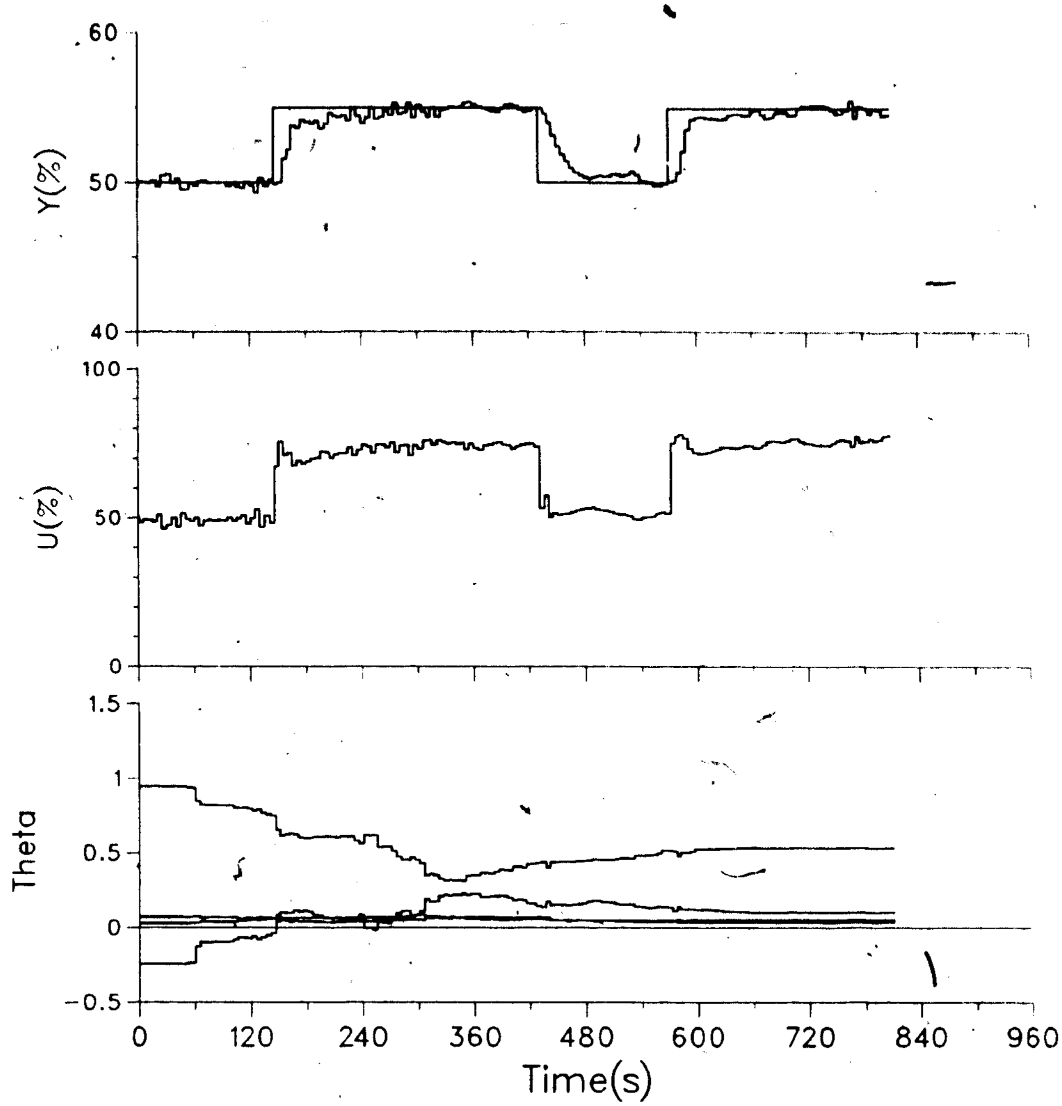


Fig. 8.28 Effect of Measurement Noise on Tuner Performance

prediction scheme alone is not sufficient to guarantee that closed loop performance will be "good". $Q(z^{-1})$ must be adjusted so that the appropriate amount of control weighting is used.

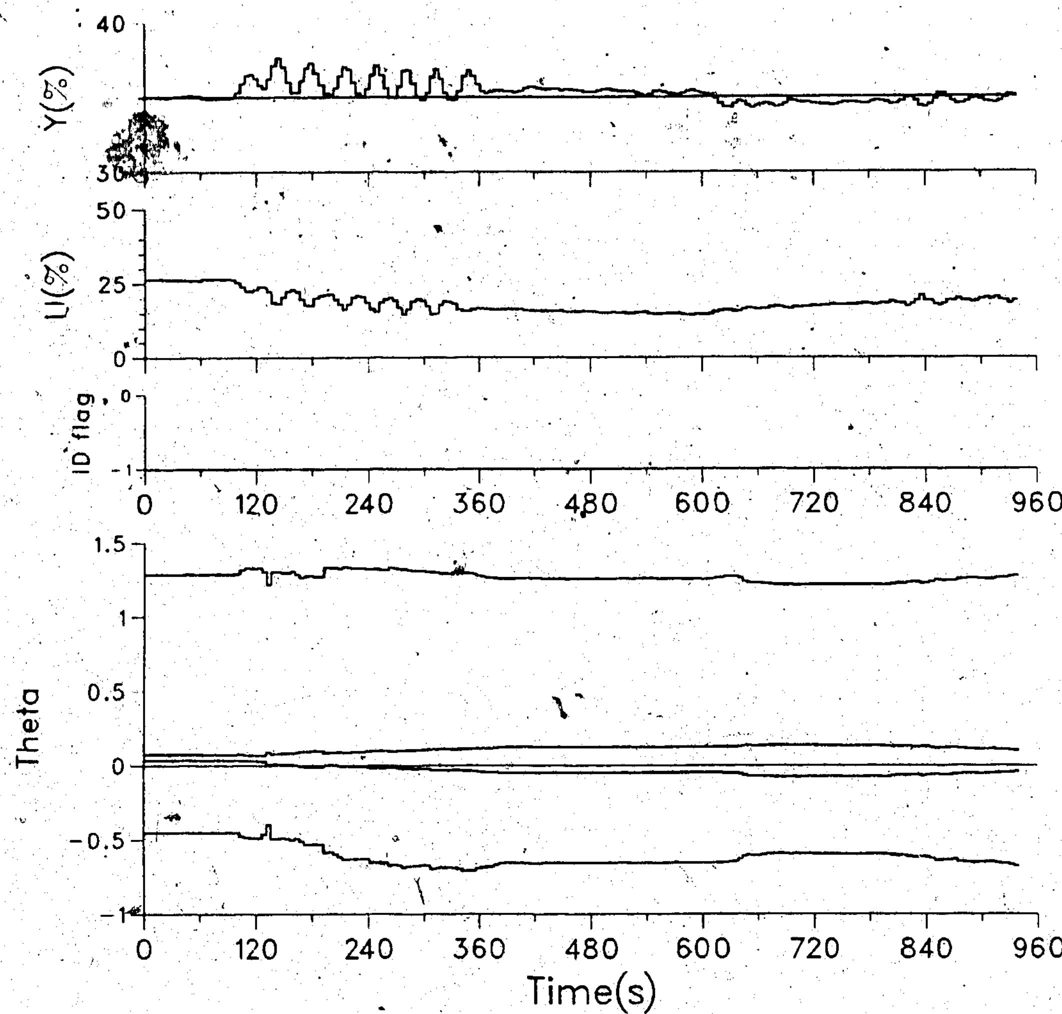


Fig. 8.29 Closed Loop Response to Change in Autotransformer Output (A^3 System)

8.7.4 Effects Of Oscillatory Disturbances

The closed loop tuner for the A^3 controller does not adjust Kc in response to load changes. If oscillatory disturbances are strong enough they may force the adaptive predictor to adjust $\theta(k)$ incorrectly, possibly resulting in unstable control.

When unmeasured disturbances enter the closed loop, the

STC effectively modifies the process model in an effort to correlate u(k) with y(k). In the open loop case with u(k) fixed, the process model could appear to be unstable if y(k) drifted away from its initial steady state value. The resulting $\theta(k)$ would be incorrect for closed loop control.

The STC controller must either recognize that a disturbance has entered the system and disable update of $\theta(k)$, or somehow estimate the disturbance term so that it will not adversely affect $\theta(k)$. If the effects of unmeasured disturbances are accounted for in the bias term, $\bar{d}$, one alternative would be to decrease the forgetting factor for mean estimation to 0.0 so that $\bar{d}=\bar{y}(k)=y(k)$ and, with mean deviation data, $\tilde{y}(k),\tilde{u}(k)=0.0$. This would suspend parameter identification until $\lambda_u(k)$, $\lambda_y(k)$ were again made non zero. From Chapter four $y(k+d)\cong\bar{y}(k)+\tilde{y}(k+d)$ where $\tilde{y}(k+d)=f(\tilde{y},\tilde{u})$. With $\lambda_u(k)$, $\lambda_y(k)=0.0$, $\hat{y}(k+d)=\bar{y}(k)$. This implies that in the presence of oscillatory, unmeasured disturbances STC should use direct feedback of y(k) rather than predicted estimates.

8.7.5 Nonlinear Processes

The demonstration of tracking ability in section 8.3 used a series of -5% set point changes in order to allow both the adaptive prediction and tuner time to adjust $\theta(k)$ and Kc to changing process dynamics. Due to an inadequate design, the A^3's tuner could not correctly adjust controller gain for the large -20% set point change. Figure 8.30 illustrates the effect of nonlinear process gain on closed loop performance.

The first three set point transients demonstrating stable control used a nonlinear process gain compensation option to adjust Kc as a function of operating set point. At t=420s, this feature was disabled with the set point at 50% of span. Without the compensation a set point change to 30% resulted in sustained oscillations in control and process output. The lower plot in Figure 8.30 shows the value of controller gains as a function of operating set point to indicate the degree of nonlinearity in the process.

8.8 Evaluation of Controller Features

8.8.1 Digital Filtering of Process Measurement

The sensitivity of the adaptive predictor to measurement noise was shown in section 8.7.3. The use of filtering to remove the higher frequency modes of the measured process output prevents noise from being fed back to either the controller or the adaptive predictor. This results in less drifting of estimated parameters used for calculating predicted outputs and less variance in controller outputs.

8.8.2 Clamping of Controller Parameters

As a safeguard the operator of an adaptive controller should be able to specify hard limits on adapted parameters so that failure of the adaptation mechanism will not result in unstable control. An adaptive tuner may give poor control parameters if operator specified performance can not be

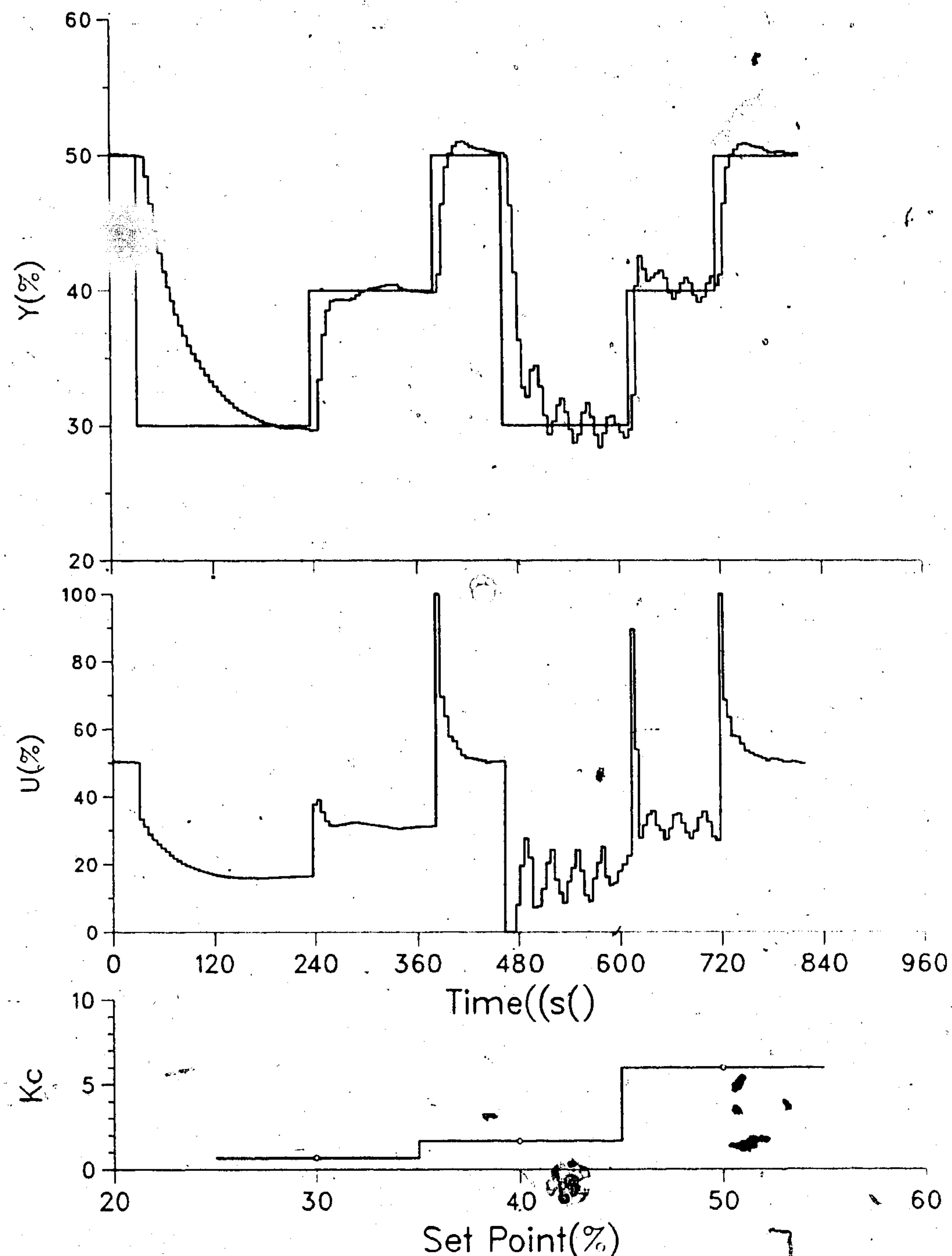


Fig. 8.30 Effect of Nonlinear Process Gain on A^3 Closed Loop Stability (Run AC39.5)

achieved. This was the case with overshoot specifications for a dominantly damped process, resulting in very high controller gain and excessive control action.

This particular feature was not implemented within the A^3 control system.

8.8.3 Compensation for Nonlinear Process Gain

A nonlinear process gain compensation scheme implemented within the GMV allows for automatic adjustment of controller output filtering ($Q(z^{-1}) \rightarrow Kc$) based on relative changes in estimated static process gain.

If, for a given operating point, a particular controller gain has been selected through tuning to give a desired performance and a set point change is then specified, the gain compensation will:

1. Estimate the process gain expected at the new set point.
2. Compare the new process gain with the current value.
3. Adjust the effective controller gain Kc and therefore $Q(z^{-1})$ so that $K_p(w_i(k)) \cdot Kc(w_i(k))$ is held constant.

When compensation is operating, the controller gain is not adjusted at every control interval due to computational load considerations. It is adjusted once per set point change based on the new set point and estimated process gain. This approach is adequate for maintaining stable performance at steady state, but results in degraded transient performance. For example, a set point change to a high process gain region will result in an immediate decrease in controller gain based on the relative change in process gains. This conservative approach will result in sluggish performance. The ideal approach would be to perform continual scheduling based on actual operating point (not set point) which would require estimation of process gains at every sampling interval.

Figure 8.31 demonstrates the effect of gain compensation on performance with the nonlinear plant. Figure 8.32 contains the corresponding trajectory of compensated controller gain for the same run. For a series of -5% set point changes from 55 to 30% of span, the controller gain ranged from 6.0 to 0.26. While the transient responses are somewhat sluggish, closed loop stability is maintained.

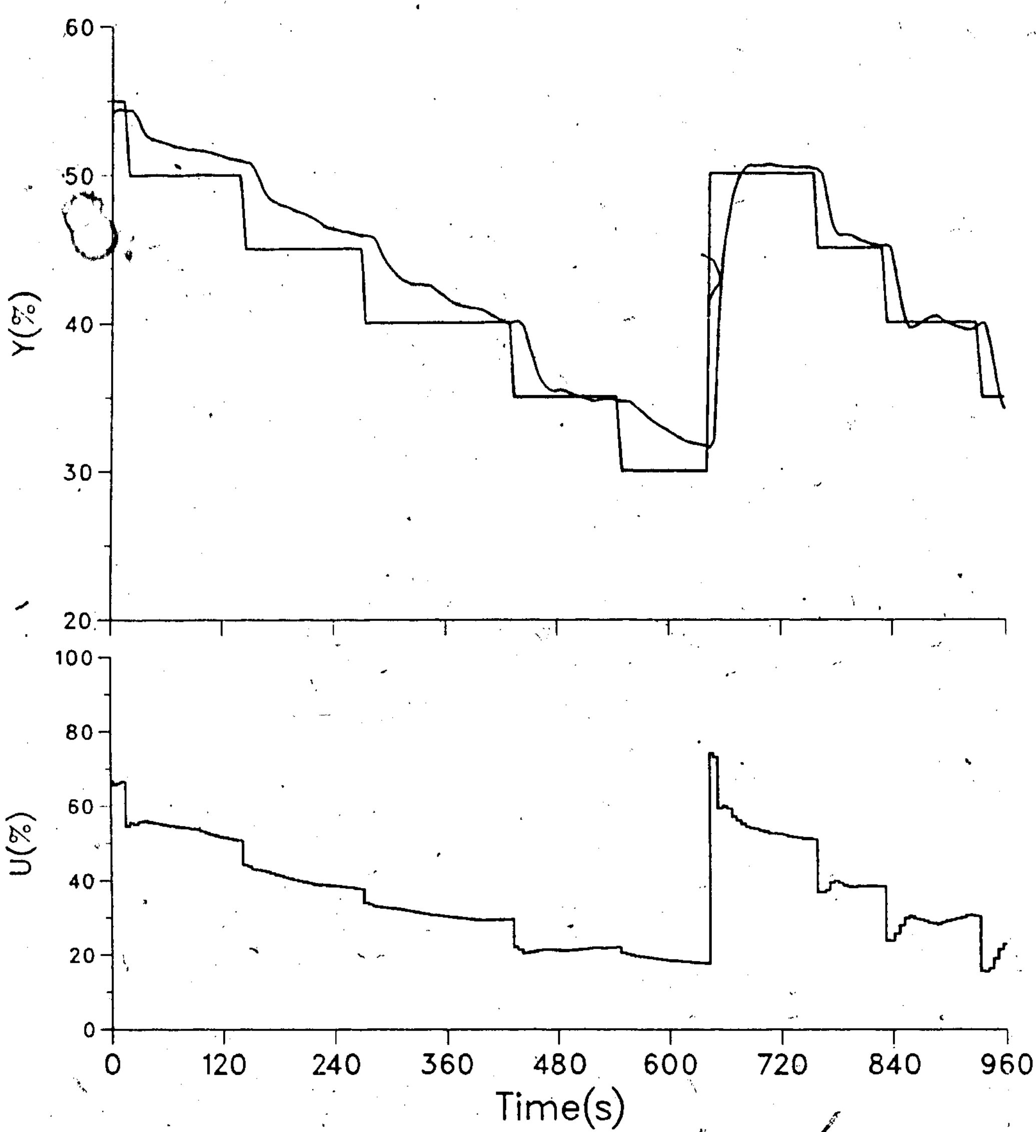


Fig. 8.31 Use of Gain Compensation in Controlling a Nonlinear Process (A^3 Run AC4.2)

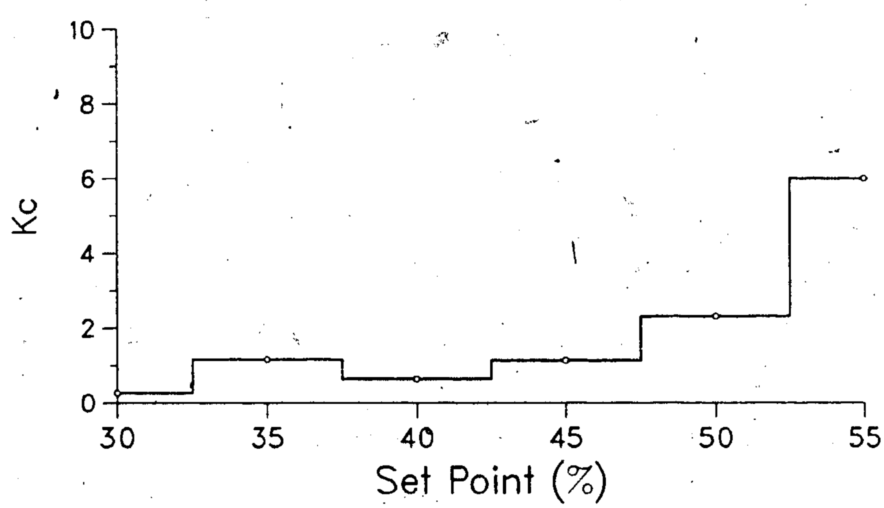


Fig. 8.32 Trajectory of Compensated Kc for Run AC4.2

8.9 Summary of Features and Performance - GMV Controller

1. The A^3 control system uses the Generalized Minimum Variance controller of Clarke and Gawthrop (1979). In this work, GMV is considered to be a 1/Q control law acting on filtered, predicted set point error.

2. The discrete filter Q was chosen to be an inverse PI control law acting on $(Rw(k)-\hat{y}(k+d))$ where R was a discrete exponential filter with user specified time constant and $\hat{y}(k+d)$ a prediction of process measurement.

3. The inverse PI control law formulation for Q allows for rejection of steady state controller offset and there is an explicit relationship between adjustable parameters in Q and PI controller parameters that allows for on line modification of Q based on changes in PI constants.

4. GMV control uses predicted error, $(Rw(k)-\hat{y}(k+d))$ in which an estimate for future process outputs is required. The A^3 control system uses either recursive least squares (UDU^T factorization and Fortescue's forgetting factor) or improved least squares for

implicit estimation of $\hat{y}(k+d)$.

The adaptive prediction mechanism recursively identifies parameters $\theta(k)$ in a linear model based on measured process I/O data and then uses those parameters to estimate $\hat{y}(k+d)$. For the purposes of practical control, RLS must at least give asymptotic convergence of $\hat{y}(k+d)$ to $y(k+d)$. In this work, both RLS and ILS met this objective using mean deviation data and load reconstruction techniques.

5. It was found that tracking performance degraded with increasing process nonlinearities. The problem was offset by using a covariance multiplier for ILS during set point transients which allowed for greater sensitivity to changes in process dynamics. The RUD algorithm with Fortescue's forgetting factor automatically inflates $P(k)$ when tracking prediction errors during transients.

6. The recursive identification algorithms require an assumed process model structure. In the experimental evaluations, a default second order model was used. Both the ILS and RUD algorithms gave satisfactory predictions.

7. During long periods of steady state operation ILS, through the use of on-off criteria, suspended adaptation thus preventing parameter drift and problems associated with poor numerical conditioning. Noise levels in measured I/O signals were not significant so that the RUD algorithm also gave good long term steady state performance.

8. The advantage of using predicted set point error over actual error when calculating controller output is clear. Good prediction introduces phase lead into the closed loop much like classical derivative action. With adaptive prediction, however, the degree of phase lead is dependent on actual process dynamics and changes with operating point for nonlinear processes. The A^3 system's PI control with predicted error offered two adavantages over conventional PI control. For a given set of PI controller constants the predictive control gave less oscillatory closed loop response to set point changes. With the nonlinear process less tuning of PI constants (and therefore $Q(z^{-1})$) was required for changes in operating point.

9. GMV control requires a separate self-tuning scheme based on closed loop performance. From initial experimental

work with the nonlinear process, it was found that $Q(z^{-1})$ had to be adjusted by the control engineer during closed loop operation. Changes in process dynamics with operating point dictated that varying amounts of control weighting be used. By selecting a PI control law structure for 1/Q, closed loop performance was modified using heuristic tuning rules to adjust the effective PI parameters. These PI constants were then used to explicitly determine appropriate filtering, $Q(z^{-1})$, of u(k).

10. An ad hoc self-tuner tuner was designed to automatically adjust controller gain based on the difference between user specified and actual overshoot to set point changes. The tuner demonstrated that the use of closed loop tuning is a necessary feature for an adaptive controller. With a robust set of tuning rules relating PI constants to closed loop measures of performance, GMV could be made a truly practical adaptive controller. The performance of the tuner used in this work was satisfactory demonstrating convergence of closed loop overshoot for set point transients. A more complete set of tuning rules would increase the A^3 system's adaptive performance and practical range of application.

11. The A^3 control system's initial open loop identification option enables an operator to startup specifying only two parameters: the magnitude of a step change in u(k) and the allowable deviation of the process output from its initial steady state value. Using this information, the A^3 system automatically perturbs the open loop process and obtains initial estimates for required inputs. For open loop stable processes, the initial test gave good starting values for sample time, process delay time, process time constant and static gain. This information was then used to determine estimates for PI control controller parameters and set point filter time constant which were then used to specify $Q(z^{-1})$ and $R(z^{-1})$. If reliable initial estimates for the above inputs are available the system can be easily started up without using the above option. PI parameters from the initial open loop test were obtained using the Cohen and Coon reaction curve method with 25% overshoot and $\frac{1}{4}$ decay specifications. For the nonlinear plant, more conservative specifications are required to avoid oscillatory control upon transfer to automatic adaptive mode. PI control constants can be later modifified by the user or adapted based on user specified overshoot.

12. A nonlinear static process gain compensation feature allows the operator to make use of reliable steady state

I/O data. Using this information controller gain is adjusted following set point changes as a function of relative changes in estimated process gains. Experimental results demonstrate that these adjustments help to maintain stable closed loop control for highly nonlinear processes. With good compensation there is less "load" on the closed loop self-tuner which ideally will only have to adjust controller gain for changes in closed loop performance specifications.

13. The A^3 controller's closed loop performance can be modified for servo changes using $R(z^{-1})$ for set point filtering. In this work, an exponential filter with user specified closed loop time constant was evaluated. Set point filtering improved overall performance by smoothing controller action and transitions to new operating points. This smoothing of process I/O data resulted in smoother changes in $\theta(k)$ and allowed the adaptive predictor more time to track changes in process dynamics during set point changes.

14. A discrete approximation to a second order Butterworth filter was used for filtering of process measurements. This discrete approximation uses a bilinear transformation with a prewarped frequency scale. The

digital filter was used as a low pass filter, removing high frequency content (noise) from measured process outputs. Filtering resulted in better performance of the adaptive predictor with less variance in $\theta(k)$ and $\hat{y}(k+d)$ and therefore controller outputs were smoother.

15. The provision for backup conventional PID control allowed the operator to quickly stabilize the closed loop in the event of problems with the A^3 controller.

16. The A^3 control system is very easy to use. The operator uses a menu driven task for configuration, displaying information and modifying adjustable parameters. Options are clearly defined so that the operator can quickly have the system running in closed loop adaptive mode. Once the adaptive controller is operating, a single "MAIN" table is used to display and modify parameters. Lower level parameters such as those related to recursive least squares identification or the closed loop tuner are accessed using other menus.

17. The A^3 control system give good closed loop performance for nonlinear processes using the default configuration and reasonable estimates for required inputs. Once

running, A^3 is as easy to operate as a fixed gain PI controller. The added features are easily used and disabled if desired. Design features such as automatic adjustment of $Q(z^{-1})$ and $R(z^{-1})$ based on desired PI control constants and closed loop time constant indicate how GMV can be made more "user friendly".

9. Comparative Study of the Three Adaptive Controllers

9.1 Introduction

Historically, the most common form of single loop, feedback control has been the three term PID control law, in its various forms. The selection of the adjustable controller parameters is typically performed using conventional tuning methods and heuristic rules. However, in recent years both the academic and industrial communities have moved towards self-tuning controllers capable of automatically tuning themselves with little or no operator supervision. Justification for such controllers has stemmed from the need for more consistent and better control performance for a wider range of process conditions. Nonlinear and slowly time varying processes require periodic retuning to maintain desired performance levels as well as the initial tuning at startup. This can be a time consuming task on large applications unless some "automatic" tuning mechanism is available.

In the selection or design of a self-tuning controller, the end user must ensure that his requirements for control are satisfied. With the wide range of options currently available, it is important to consider: the controller mechanism and adaptive design, practical performance features, amount of information required for initialization, documented performance and overall ease of use. Each of these topics is discussed further in the following sections.

The format of the chapter is intended to summarize and document the key points that should be considered by anyone designing or implementing an adaptive or self-tuning controller for industrial applications.

9.1.1 Controller Structure and Adaptive Design

Currently, the terms "adaptive" and "self-tuning" are used to describe a wide range of both industrial and academic controllers. In the following discussion, self-tuning describes those controllers that adjust controller parameters based on some closed loop measure of performance. Adaptive will be reserved for those controllers that incorporate model based estimation schemes into their algorithms for the purpose of prediction and identify, or adapt the model parameters on-line.

9.1.2 Model Based Self-Tuning PID - Advisor

This type of controller uses a model based estimation scheme to predict future process outputs. Calculated model parameters are then incorporated into a self-tuning design procedure to determine PID controller parameters. The modelling scheme requires that assumptions be made regarding the process being controlled. A model structure must be selected so that the number of estimated model parameters is fixed. Figure (9.1) shows a block diagram structure for this type of controller. Turnbull Control Systems' model 6355 controller features continual calculation of recommended PID

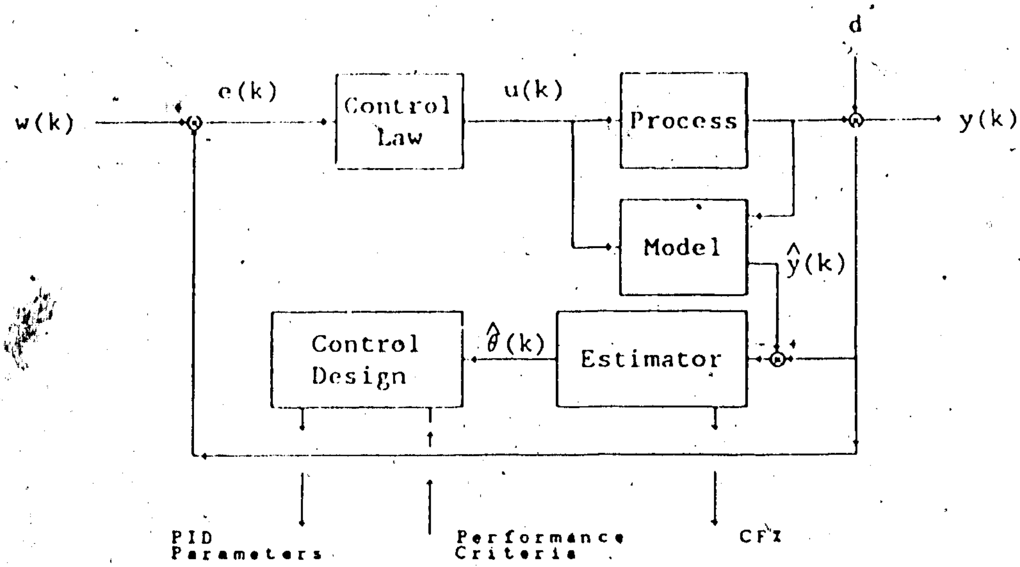


Fig. 9.1 Structure of Model Based Self-Tuning PID Controller

parameters based on parameter estimates, θ obtained from a statistically based process model estimation scheme. The model is used only to determine θ values. The predicted output terms are not fed back to the controller for predictive control. The control design block for this type of controller typically assumes that the given θ from the model are exactly correct. The calculation of the three PID controller parameters usually involves solving for the roots of a characteristic equation. The desired closed loop performance criteria or index is prespecified (unknown in the 6355's case) so that the controller parameters are the only unknowns to be solved. The most important limitation of the above type of self-tuning scheme is that PID constants are adjusted on the basis of changes in estimated model parameters instead of on the basis of actual closed loop performance. The accuracy of θ is dependent on the statistical model's performance which, in turn, is degraded by errors in assumed model structure (likely second order) and current process conditions. Nonlinear, high order processes will be much harder to accurately model than low order, linear plants as shown in Chapter 7. Hence, the recommended PID constants may not give good control. It is important that the reader understands that the recommended PID constants are not automatically transferred to the PID control block. Based on model performance, a "confidence factor", CF(%), is generated to assist the operator in his decision to explicitly transfer these constants to the control law. In controlling the nonlinear process, confidence factors rarely exceeded 60%. During experimental runs, PID parameters were updated whenever the displayed confidence factor exceeded a value of approximately 50%.

To highlight the above points, closed loop performance of the model 6355 is shown in Figure (9.2) for the first order, linear plant and (9.3) for the high order, nonlinear temperature process. With the easier to model linear plant, the controller gave smooth, critically damped responses to step changes in set point (for the large set point changes, the Error Limit parameter, EL, took effect causing slight overshoot- see section 7.3.4), i.e. "optimal" control. However, with the nonlinear process, control was sluggish and slightly oscillatory at steady state. One important advantage of this type of self-tuning approach is is that PID parameters can be calculated as often as estimated model parameters, $\underline{\theta}$. With perfect or near perfect modelling the 6355 controller would always have "optimum" closed loop performance, unlike the "once per transient" approach used by the Exact controller. However, current technology can not give perfect modelling of real processes. Therefore, self-tuning schemes that are sensitive to model parameter estimates or modelling performance are not practical for complex processes. There is no feedback of closed loop performance to the self-tuning block so that PID parameters are a function of process model parameters (based on process I/O data) and desired performance but not actual performance.

9.1.3 Performance Based Self-Tuning PID

The most important feature of this type of self-tuning controller is that PID parameters are adjusted based on actual measured closed loop performance. Estimated PID parameters are not a function of model estimates and therefore do not suffer from the negative effects of poor modelling discussed in section 9.2.1. Foxboro's Exact controller falls into this category of self-tuning, since it used measured overshoot and damping of closed loop responses to disturbance or set point changes, as performance criteria. Figure (9.4) illustrates the Exact controller's block diagram structure with closed loop self-tuning.

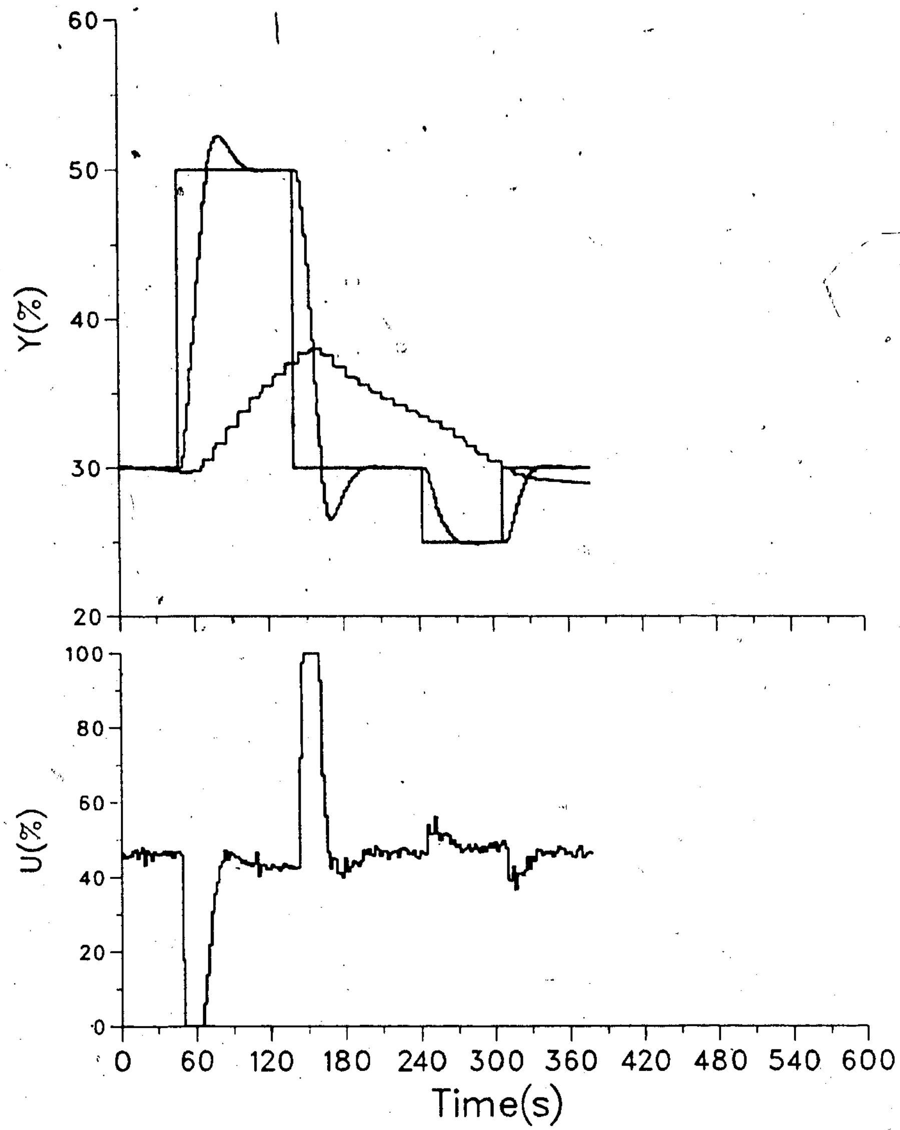


Fig. 9.2 Performance for the Model 6355 - Linear, First Order Process - Tc2 5.2 EL=5%

Unlike model based self-tuners, PID parameters are adjusted only after "closed loop performance" has been measured. In the Exact's case PID constants are tuned <u>once</u> per underdamped transient response, since overshoot and damping are "once-per-transient" measures of performance. Figure (9.5) illustrates a transient response for a +5% step in set point for the first order liquid level process. Prior

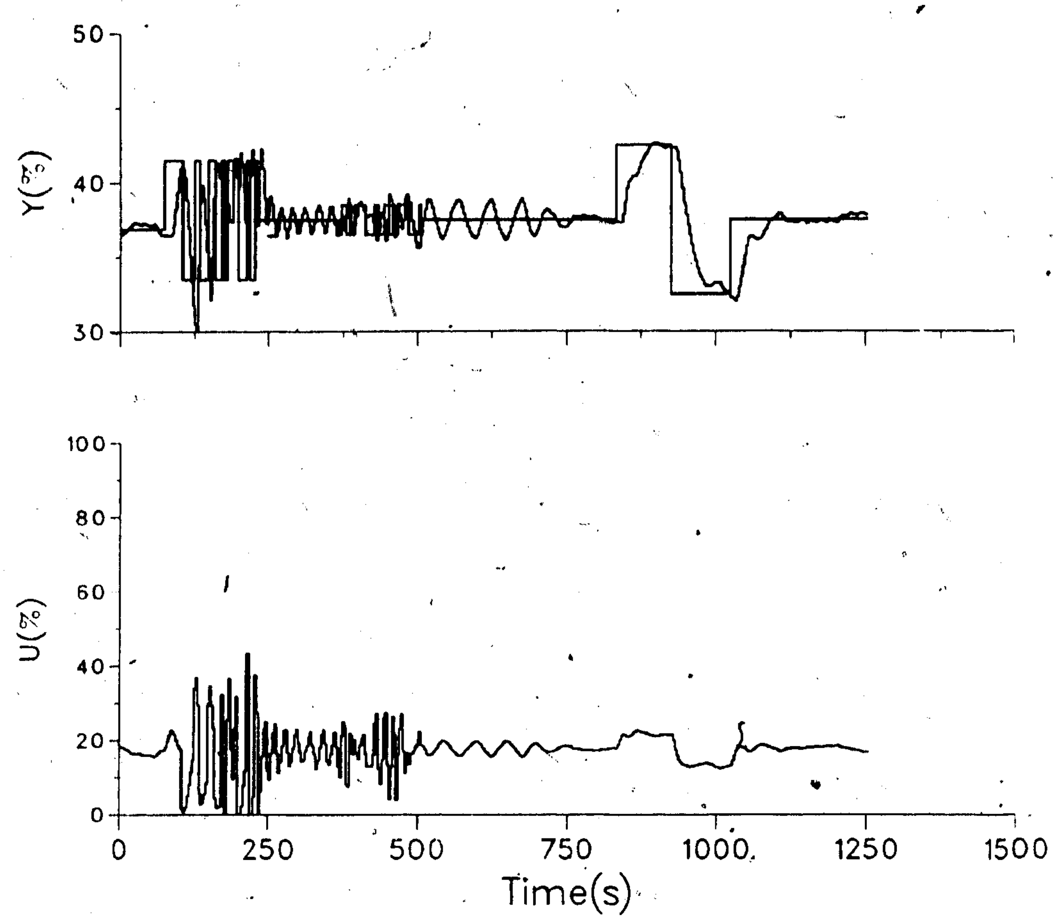


Fig. 9.3 Performance for the Model 6355 - Nonlinear, High Order Process - Tc3_3.5, Transformer at 70%

to the transient shown, converged PID values (Band=17%, Reset time=0.41 minutes, Derivative time=0.07 minutes) were obtained for specified overshoot/damping of 0.5 and 0.3, respectively. The desired closed loop performance was then changed to OVR=0.0 and DMP=0.0, and the set point change implemented. With measured overshoot and damping values of 0.46 and 0.28, the self-tuner adjusted PID constants to give: band=27%, reset time=0.44 minutes and a derivative

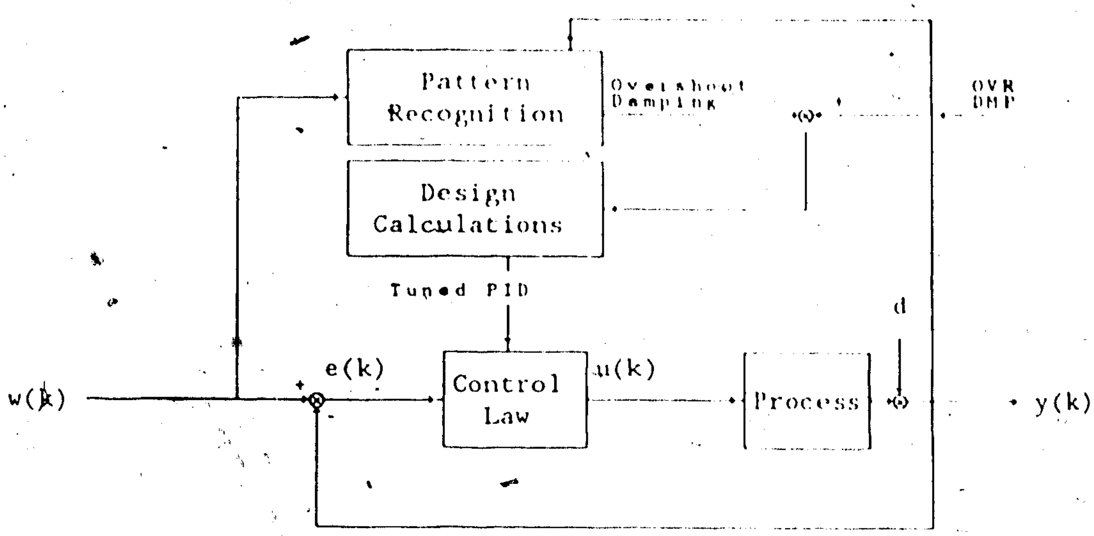


Fig. 9.4 Block Diagram Structure for the Exact Controller

time of 0.05 minutes. Subsequent set point transients then approached the desired closed loop specification.

The once per transient approach to tuning has its limitations: Figure (9.6) illustrates closed loop performance for a large set point change into a high process gain region (nonlinear process). Control about the new operating point was initially very oscillatory until a series of three peak responses were recognized and the controller detuned. This Figure simply illustrates that a user must carefully consider the implications a particular self-tuning design will have for his application. Another important aspect of this self-tuning design is the implicit assumption that a process can be forced to overshoot and

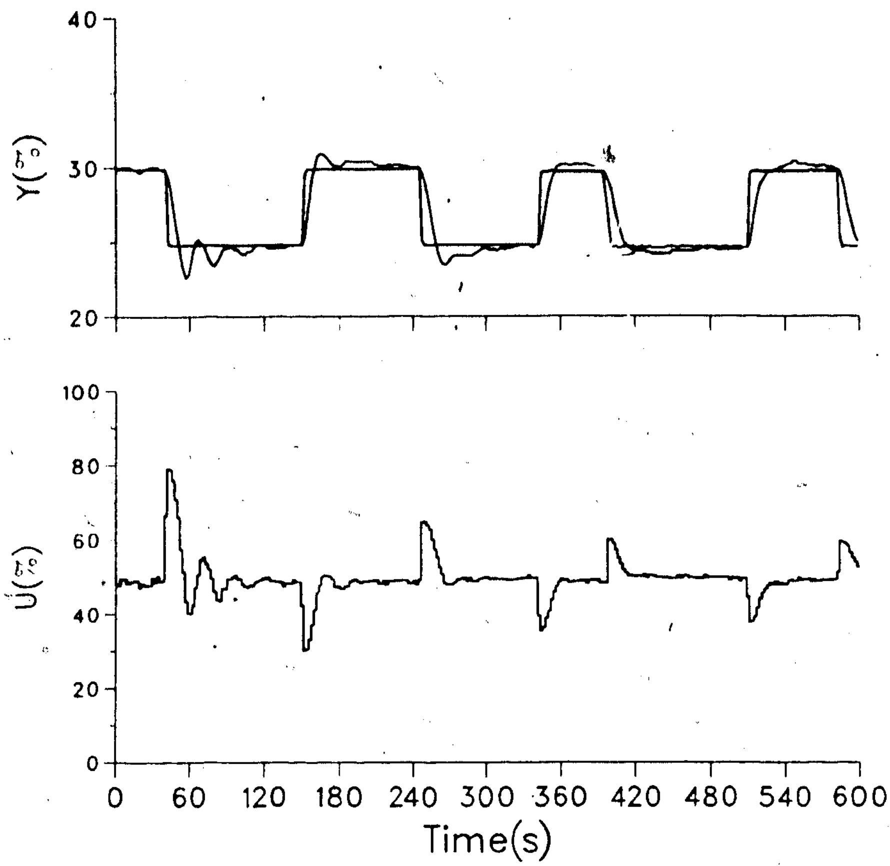


Fig. 9.5 Tailoring of Closed Loop Response Characteristics Run C_2_1.2

oscillate in a feedback control scheme, given an appropriate set of PID parameters. Because controller output, u(k), is not considered in the performance index, overshoot specifications may not be an appropriate performance index for highly damped processes. Controller gains are tuned to large values, resulting in large, high frequency oscillations in u(k). This phenomena, described as gain wind-up, is shown in Figure (9.7). In selecting a self-tuning controller, it is very important to consider what criteria are used by the tuner and their

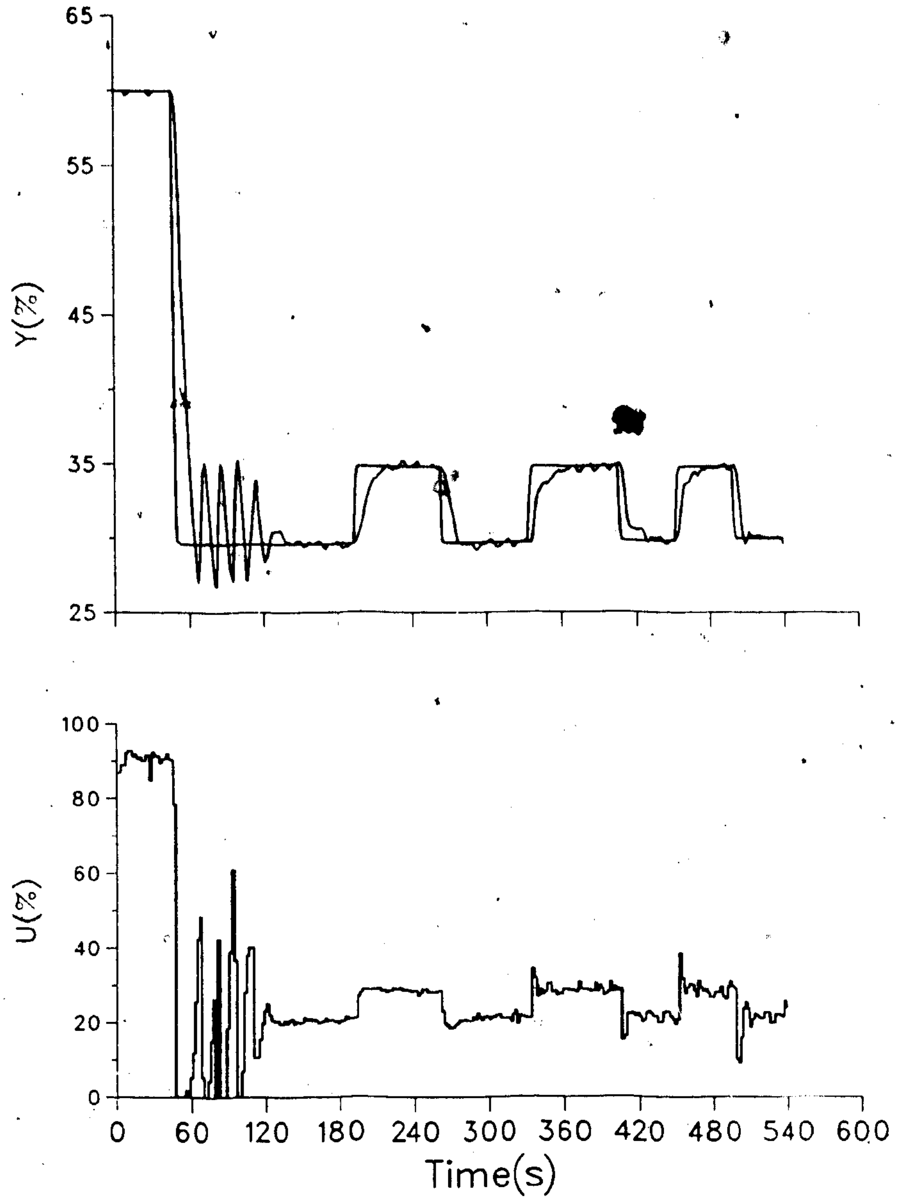


Fig. 9.6 Effect of Nonlinear Process Gain on Closed Loop Stability (CODGF.6 OVR=0.25 DMP=0.20 LAG=0.0 min.)

appropriateness for a given application.

9.1.4 Adaptive Predictive Control

The major drawback of the PID controller is its inability to compensate for processes with time delay, plus the fact

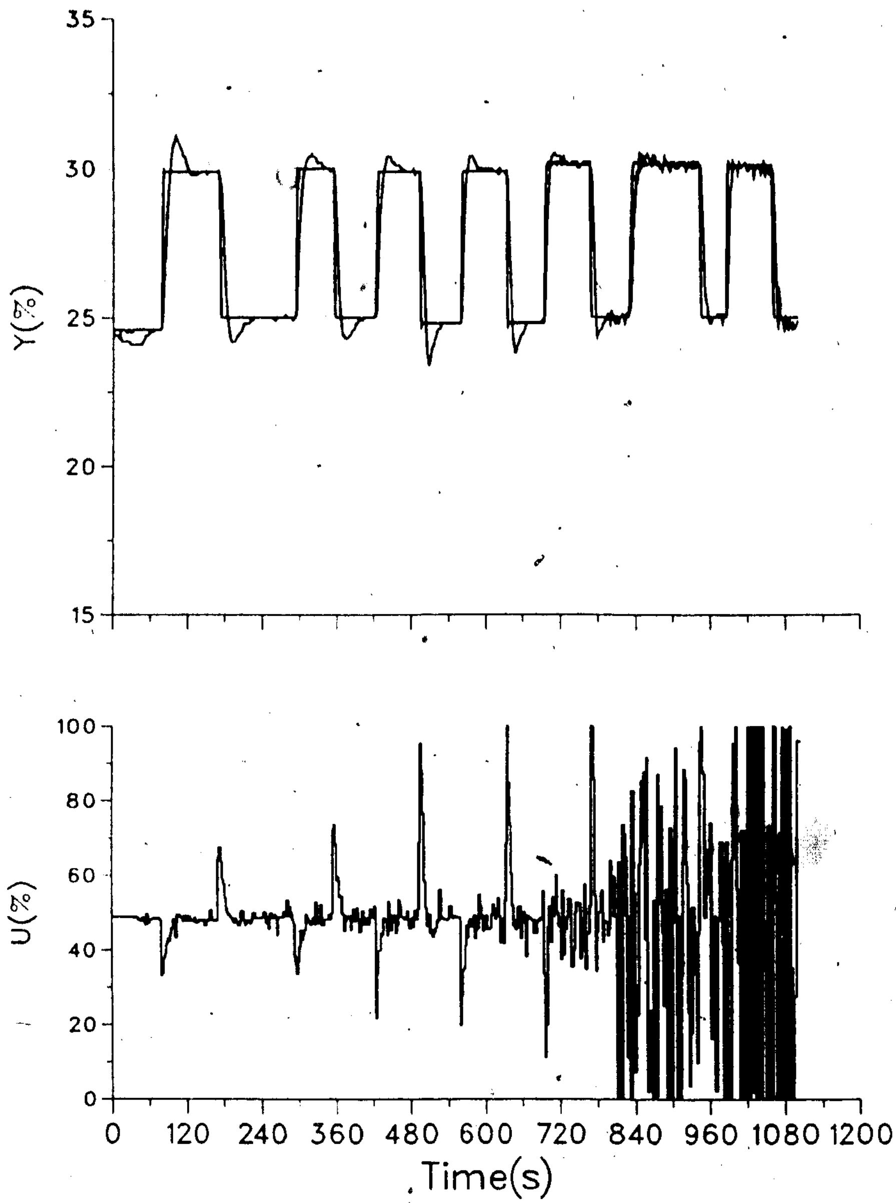


Fig. 9.7 Gain Wind-Up Due to Unreasonable Specification for OVR=0.5 and DMP=0.3 (C_1_1.2)

that it waits until there is an error in the actual process output before taking corrective action. Without time delay compensation, controller gains must be decreased to maintain closed loop stability at the expense of sluggish control performance. The Smith predictor is an example of non-adaptive predictive control which gives improved performance for plants with known process models. For nonlinear, time-varying plants, a fixed parameter process model is not adequate. The Generalized Minimum Variance (GMV) controller used in this work, and shown in Figure (9.8), is one example of an adaptive predictive controller.

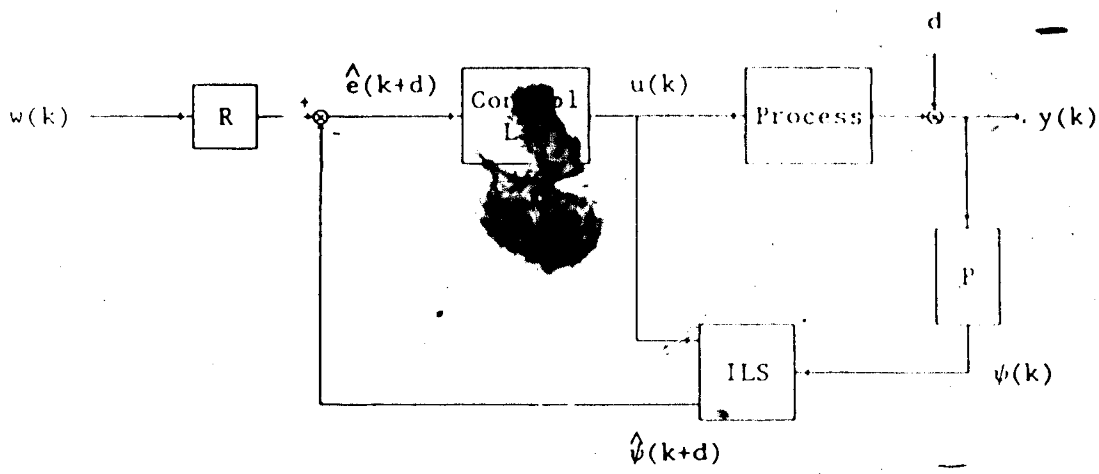


Fig. 9.8 Block Diagram Structure for Clarke and Gawthrops GMV Adaptive Predictive Controller

Theoretical considerations in Chapter three and experimental results in Chapter eight support the argument that GMV is a non self-tuning, 1/Q controller acting on predicted control error. Adaptive prediction is model based, using a recursive least squares algorithm to identify the process model parameters needed to calculate predicted process outputs. Figure (9.9) illustrates that the chosen PI formulation for 1/Q allows for Q to be adjusted using conventional tuning techniques for a PI controller. With the highly nonlinear plant used for experimental study, there was a demonstrated need for self-tuning of Q. Figure (9.10)

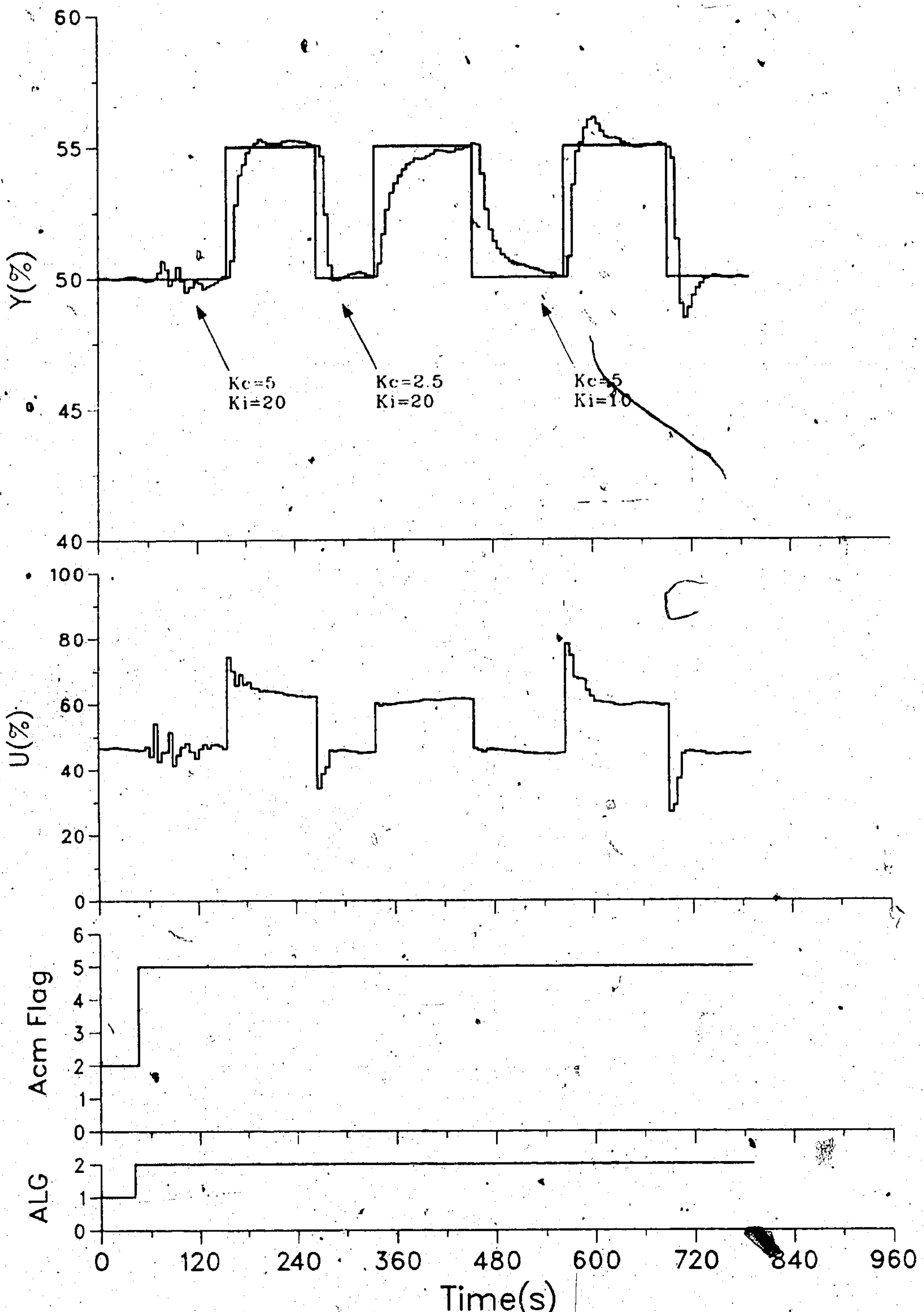


Fig. 9.9 Predictive PI Control - Effect of Tuning Kc and Ki

shows how process nonlinearities affected closed loop stability. The effective PI parameters in 1/Q were adjusted on-line by the author in an effort to offset the effects of higher process gains at lower operating points. Figure (9.10) also demonstrates the advantage of adaptive predictive over conventional error driven PI control. With initial PI constants of Kc=7.5 and Ki=10 seconds, the conventional PI controller required detuning of the reset time in order to maintain stable control (first three transients up to t=480 seconds). Using the same initial PI constants, the adaptive predictive PI controller (GMV) required less detuning as the set point was moved towards the 45% operating point. The introduction of phase lead due to prediction allows for larger controller gains and more stable control through a wider operating range than ordinary PI control.

Like the TCS model 6355, GMV performs modelling of the controlled plant, but for different purposes. In the GMV controller, estimated process model parameters are of secondary importance to predicted process outputs, $\hat{y}(k+d)$. The recursive estimation algorithm must remain stable so that $\hat{y}(k+d)$ at least asymptotically approaches $y(k+d)$, but the actual values of $\theta(k)$ are generally not critical, unless they are also used to adjust 1/Q on-line.

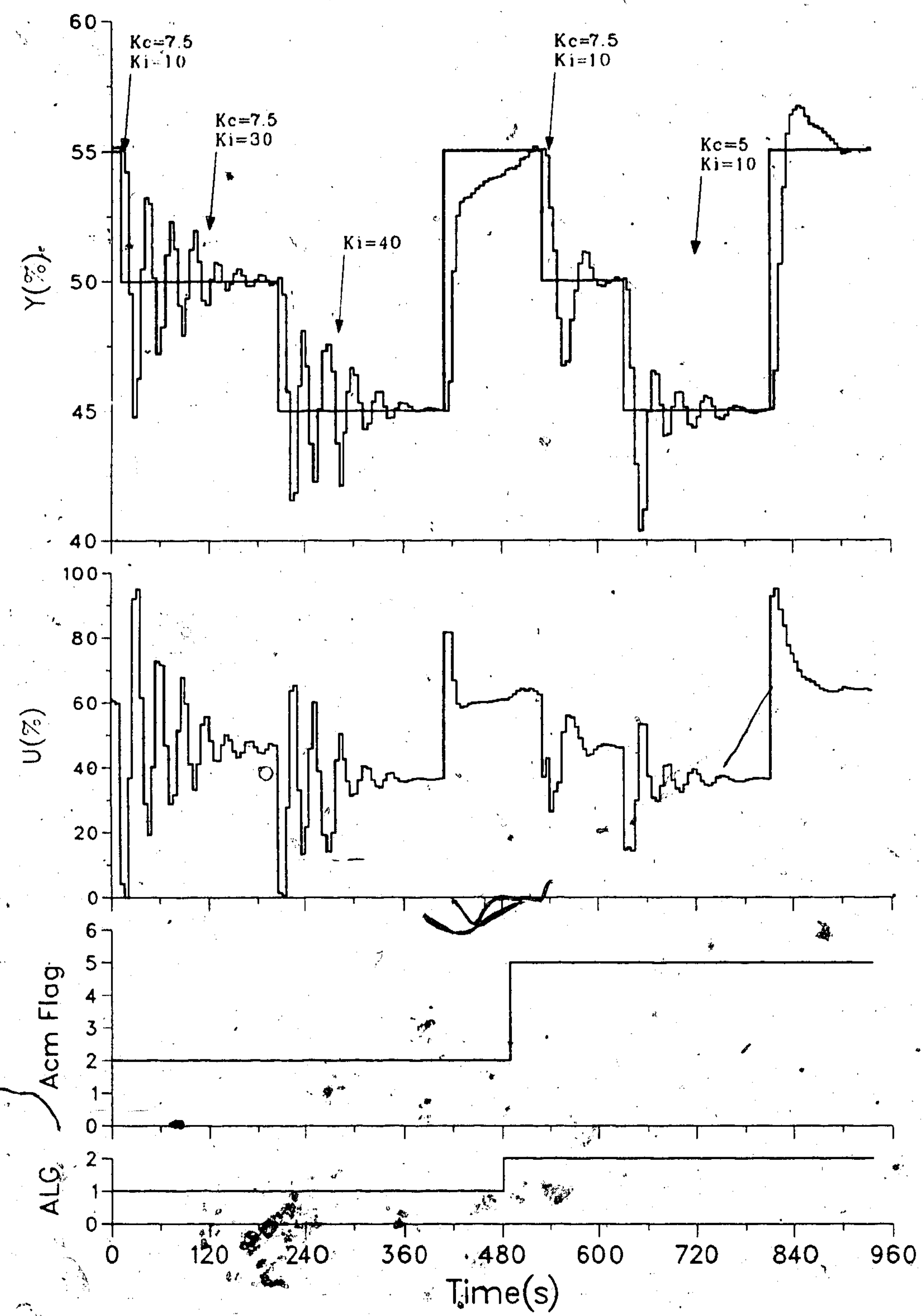


Fig. 9.10 Effect of Process Nonlinearities on Closed Loop Performance - PI vs. Predictive PI Control

9.1.5 Self-Tuning Adaptive Predictive Control

The previous discussion has demonstrated that self-tuning controllers and adaptive predictive controllers use two different approaches, each with its own advantages and problems. Evaluation work with the Exact has shown that a tuner that adjusts controller parameters based on measured closed loop performance is necessary for a practical self-tuning control system. At the same time, results with the GMV controller have demonstrated the advantage of dead time compensation and adaptive prediction over the error driven feedback control schemes. The block diagram in Figure (9.11) is for a self-tuning adaptive predictive controller which combines a closed loop self-tuner (cf. Foxboro Exact) and an adaptive predictive feedback loop (cf. GMV). The benefits derived from each type of adaptation are combined to give a better controller, overall. In this work, a self-tuning loop was added to the adaptive predictive GMV controller to adjust controller gain within the 1/Q=PI control law so that a user specified overshoot for set point transients was maintained. For a series of +5% set point changes, the tuner demonstrated its ability to maintain a user specified overshoot of 0.30. Figures (9.12) and (9.13) show closed loop performance and the trajectory of controller gains.

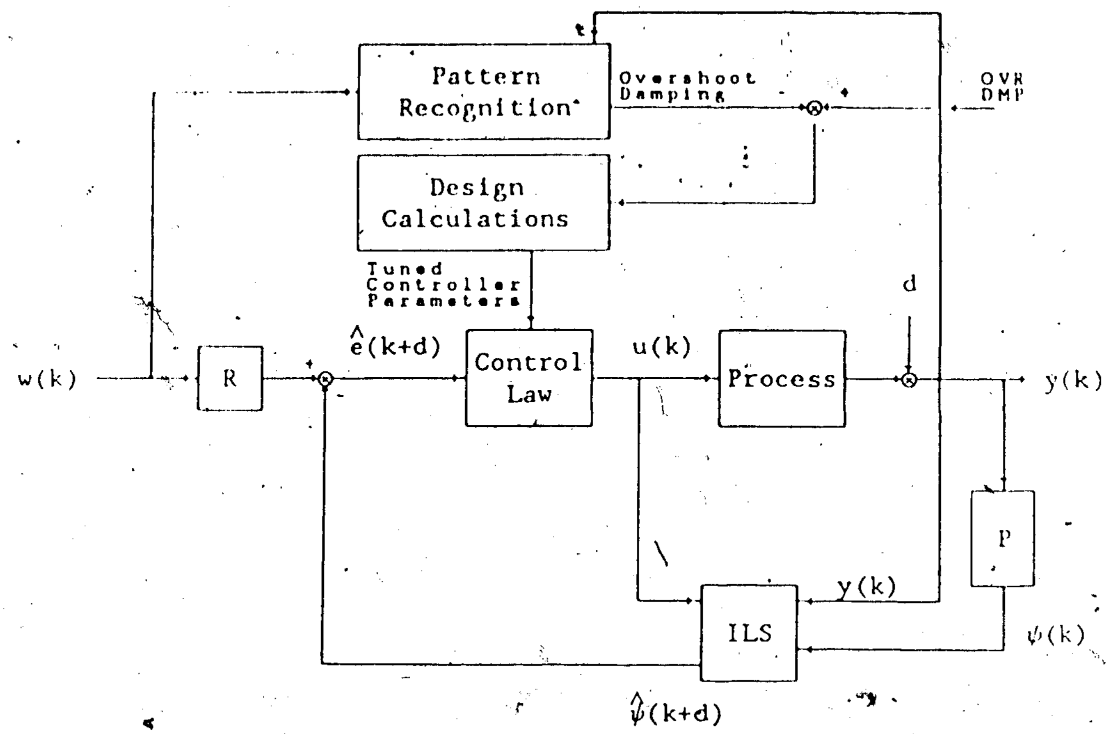


Fig. 9.11 Block Diagram Structure for an Adaptive Predictive (GMV), Self-Tuning Controller (Exact)

9.2 Practical Performance Features

The performance of any self-tuning or adaptive predictive controller is degraded by the presence of "non-ideal" process conditions. Most of these controllers' designs are based on some assumptions about the plant being controlled. Because these assumptions are virtually always violated, practical features and safeguards should, and in some cases, must be incorporated into the overall design. These features can guarantee controller integrity for long term operation and minimize operator supervision requirements, thus increasing a controller's useful range of application.

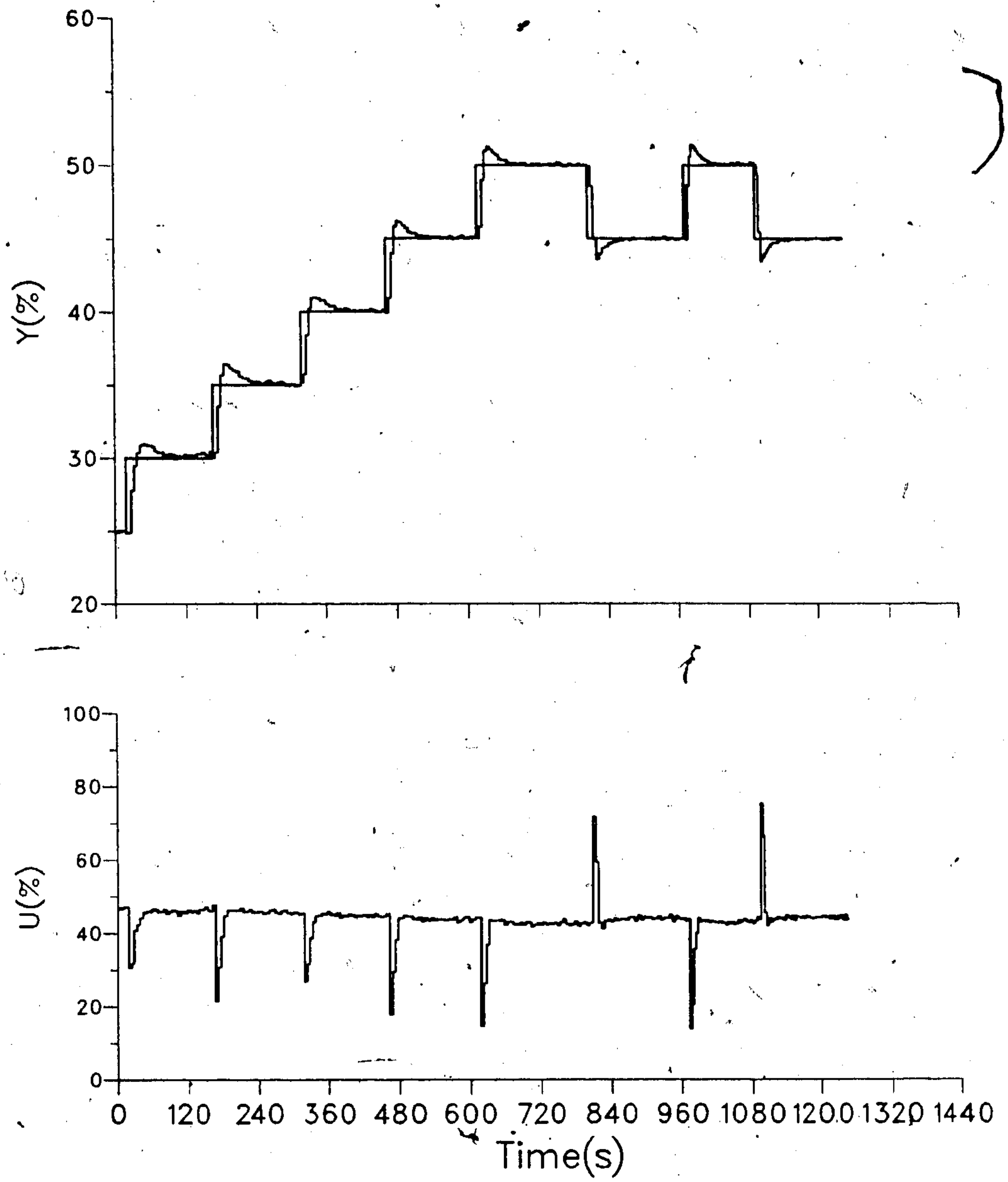


Fig. 9.12 Tracking Ability of the A^3 Controller (OVR=0.30)

9.2.1 Highly Nonlinear Plants

Plant nonlinearities are one of the main justifications for self-tuning and adaptive predictive controllers. Most model based adaptive controllers assume that a locally linearized model will accurately characterize even a

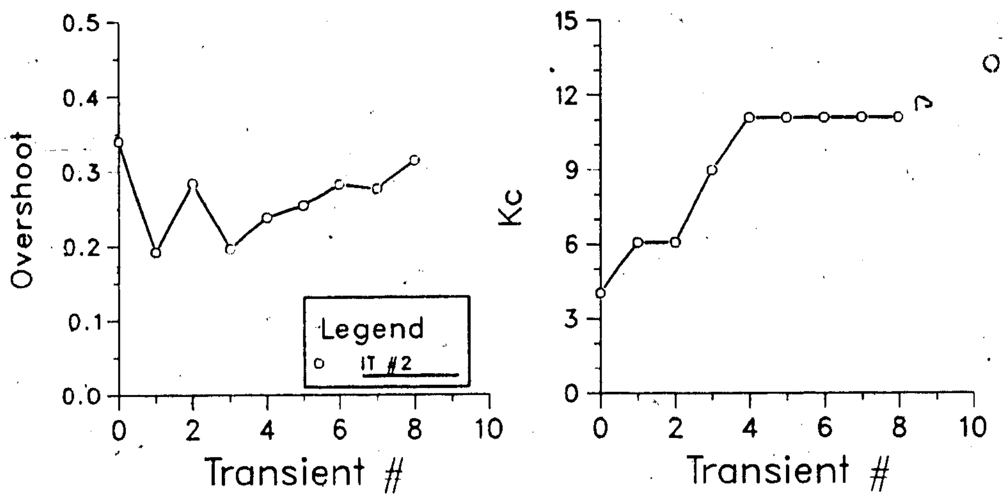


Fig. 9.13 Trajectories for Overshoot and Kc for a Series of +5% Set Point Changes to 52.5.% (OVR=0.30 DMP=0.10)

nonlinear plant. As the degree of actual nonlinearity increases, however, this assumption becomes valid only for a narrow range of operating conditions. Large changes in operating points can not be perfectly modelled and closed loop performance suffers.

One solution to the problem is the use of nonlinear process gain compensation. With known process information, such as steady state I/O data, estimated static process gains, $\hat{K}_p$, can be determined as a function of operating point. Controller gain, K_c can then be automatically adjusted so that the forward loop gain, $(K_c\hat{K}_p)$, is maintained at a constant value, based on relative changes in $\hat{K}_p$. This procedure was used in the A^3 and Foxboro Exact controllers. Figure (9.14) demonstrates how gain compensation in the A^3 controller maintains stable closed loop operation for the highly nonlinear plant. At a time of approximately 450 seconds, gain compensation was disabled and the same series of set point changes implemented. At both the 40% and 30% operating points, overall loop gain was too high, resulting in oscillatory control. The reader should note that this feature can be used in conjunction with a normal self-tuning scheme. The self-tuner will, however, only have to adjust K_c for differences between an assumed nonlinear process model and actual nonlinearities, thus minimizing the amount of tuning required.

For model based adaptive prediction, nonlinear process gain compensation may also be used to improve dynamic predictive performance. Measured I/O data can be conditioned by removal of $\hat{K}_p$ from process outputs prior to recursive estimation. This will result in less change in parameters for $\theta(k)$ for a process' given range of operation. This technique was not used in this study but is suggested for future work.

9.2.2 Process Measurement Noise

The presence of noise in measured process output signals is a common problem. Feedback of noisy measurements results in unwanted variance in controller output, u(k) (i.e. derivative action). Model based adaptive predictive schemes are particularly sensitive to noise because I/O data no longer represents the true process model. Estimated process model parameters can drift sometimes resulting in unstable predictions. In addition, calculated predictions can have

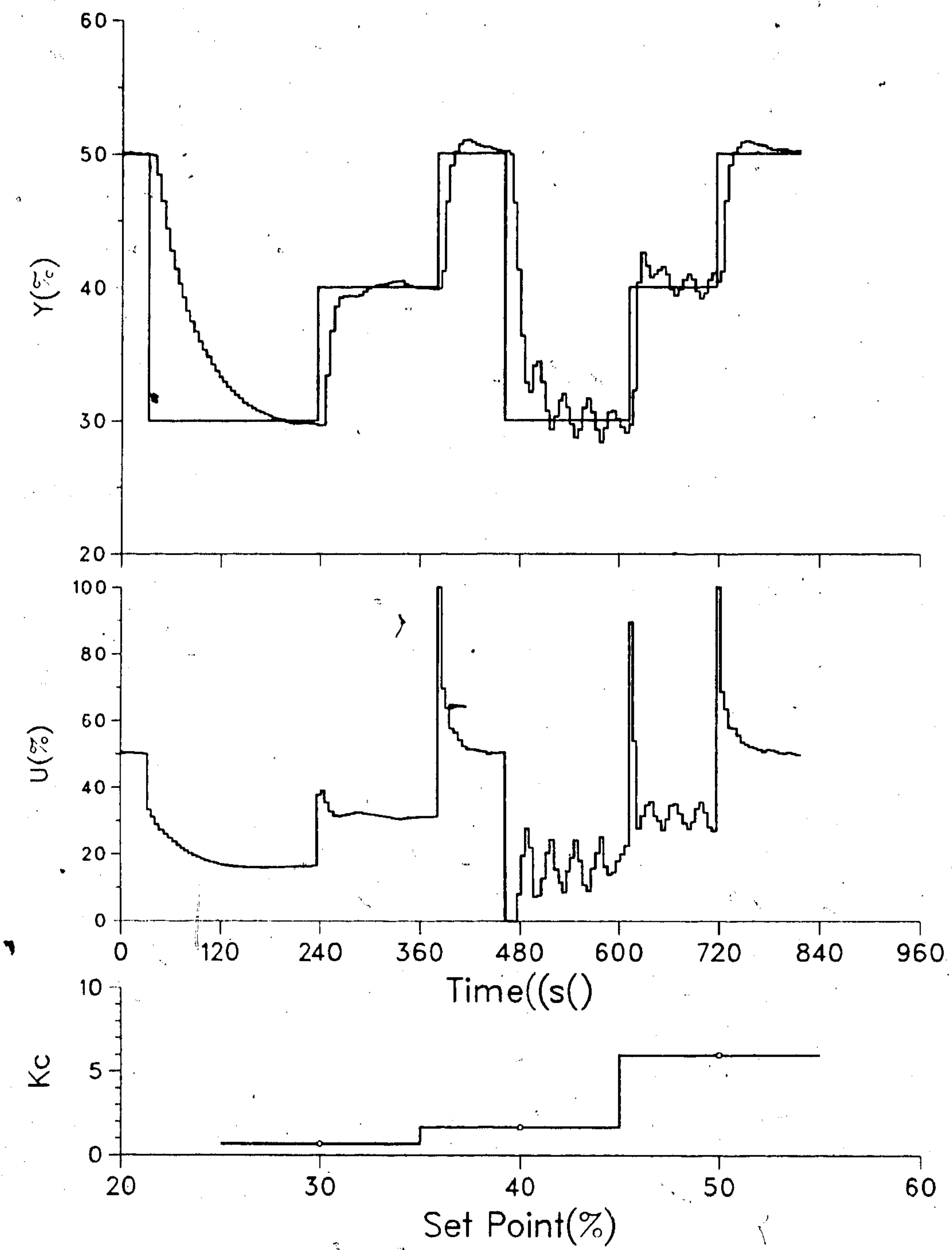


Fig. 9.14 Effect of Nonlinear Process Gain on A^3 Closed Loop Stability (Run AC39.5)

sustained offset, which will result in steady state control errors.

There are two approaches to solving the problem and both should be used. Analog and digital filtering can be used as low pass filters, to prevent feedback of noise. Filtered I/O data will more accurately represent true process dynamics, resulting in better predictor performance. The second approach is to disable identification when measured I/O signals do not contain useful dynamic information. The use and effect on performance of these two techniques is shown in Figures (9.15) and (9.16) for the A^3 control system. Prior to t=390 seconds, measurement filtering was not used. Both y(k) and u(k) were oscillatory due to noise feedback. ILS identification was used to calculate predicted outputs and disable identification due to low information content ($\|P(k-1)x(k)\|$ less than 0.001) or poor conditioning (Cond[P(k)] greater than 500). From Figure (9.15), ILS did not disable ID (ID flag=0) adequately, resulting in parameter drift (see Figure (9.16)) particularly between t=180 seconds and t=390 seconds (This could have been corrected by changing the user specified ON-OFF criteria limits). The effects of filtering are clear. The variance in both y(k) and u(k) decreased to very small levels. From Figure (9.16, parameter estimates quickly stabilized. During and following the +5% set point transient at t=570 seconds, parametes estimates had converged. Identification was subsequently suspended due to poor conditioning.

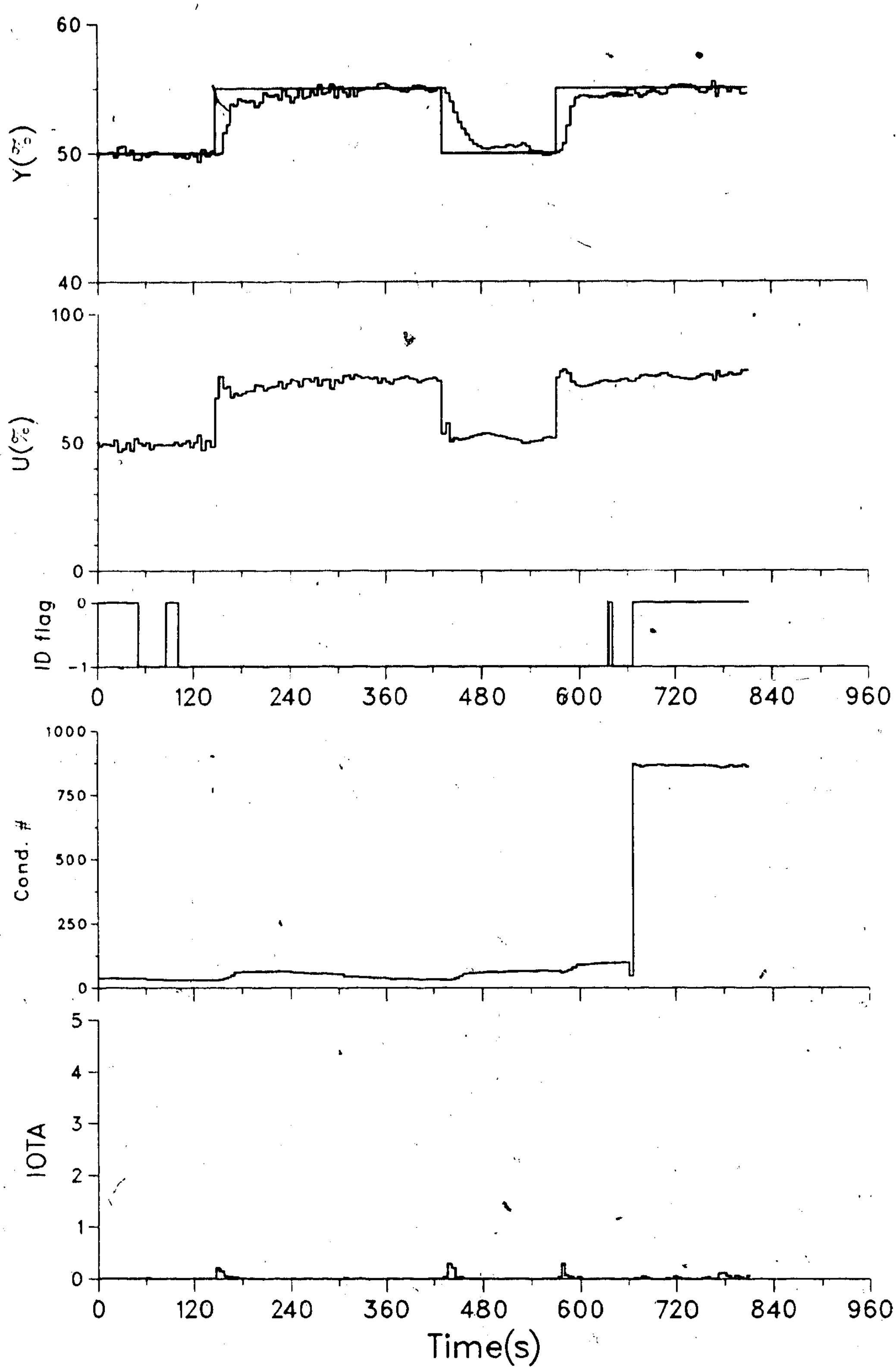


Fig. 9.15 Effect of Measurement Noise on Controller Performance

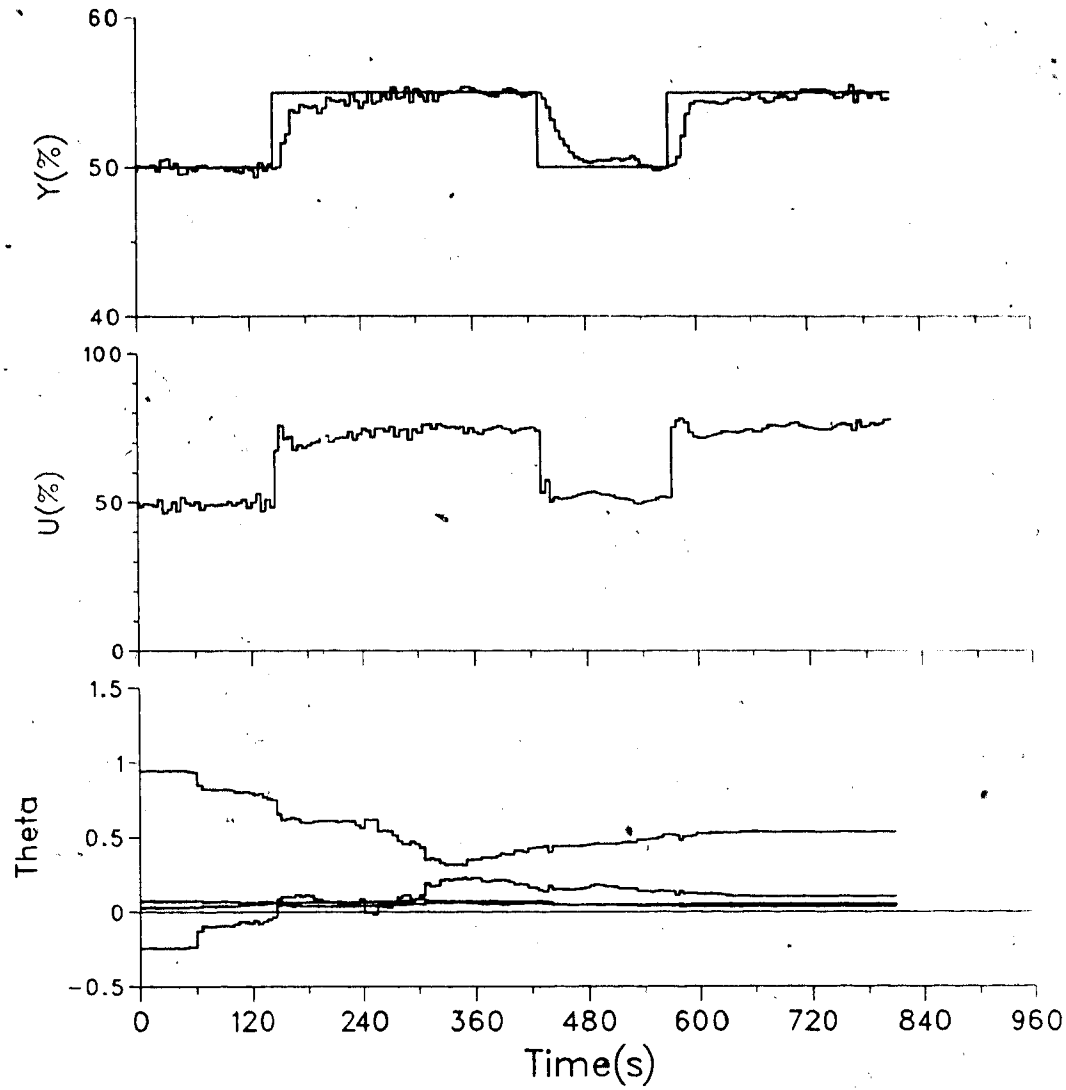


Fig. 9.16 Effect of Measurement Noise on Tuner Performance

9.2.3 Variable Transport Delay

The use of predictive control has been shown to improve control performance for processes with time delay. In turn, adaptive predictive control improves performance with nonlinear or time varying processes. Although not shown in this work, incorrect assumptions about model transport delay can have disastrous effects on predictor performance and closed loop stability. Additional design effort is required to implement an adaptive dead time compensation scheme (ADTC). Both the model 6355 and the GMV controller assume a fixed process delay time with the process model. A robust ADTC scheme would increase any model based controller's useful range of application. Because the Foxboro Exact does not explicitly consider time delays, it would have to detune if the time delay increased.

9.2.4 Nonminimum Phase Processes

Nonminimum phase or wrong way response processes can not be handled using conventional PID control without detuning controller gains so that inverse responses are essentially ignored. GMV control suffers from the same limitation when 1/Q is formulated as a PI control law. The use of lower controller gain implies heavier weighting on u(k) in the performance index, which results in a shift in closed loop poles to more stable regions of the unit circle. The other common approach is to select controller sample times greater than the duration of the inverse response. This allows for higher controller gains (less weighting on u(k)) provided that sample times do not approach the dominant time constant of the open loop process.

With the GMV controller, the design of the P and Q filters can be tailored for specific non-minimum phase applications at the expense of an increase in complexity.

9.2.5 Limits on Variance of Controller Output Oscillations

One criteria often omitted from the closed loop performance indices is the dynamic behavior of u(k). Foxboro's Exact is such an example. For highly damped processes, overshoot specifications can result in large control gains and oscillations in u(k), as was shown in Figure (9.8) (gain wind-up). As a safeguard, Foxboro has provided an option to automatically decrease controller gain when sustained high frequency oscillations in u(k) exceed a user specified limit for three minutes. Figure (9.17) demonstrates the use of this feature in controlling the damped first order process. The controller was detuned at times of 90, 260 and 440 seconds due to a specified 3% limit on oscillations in controller output.

9.2.6 High and Low Limits on Tuned Controller Parameters

If a user has available, known limits for controller parameters based on past experience or closed loop design, they can be used to ensure that tuned values never exceed these specified limits. Such a feature is easily incorporated into a controller's design. The performance of a clamping option in the Exact controller is shown in Figure (9.18). This run used the same initial conditions as that used to demonstrate the gain wind-up problem in Figure (9.8), but the initial band value of 30% was limited to a range of 7.5% to 120%. Clamping prevented the tuner from implementing large controller gains due to inappropriate

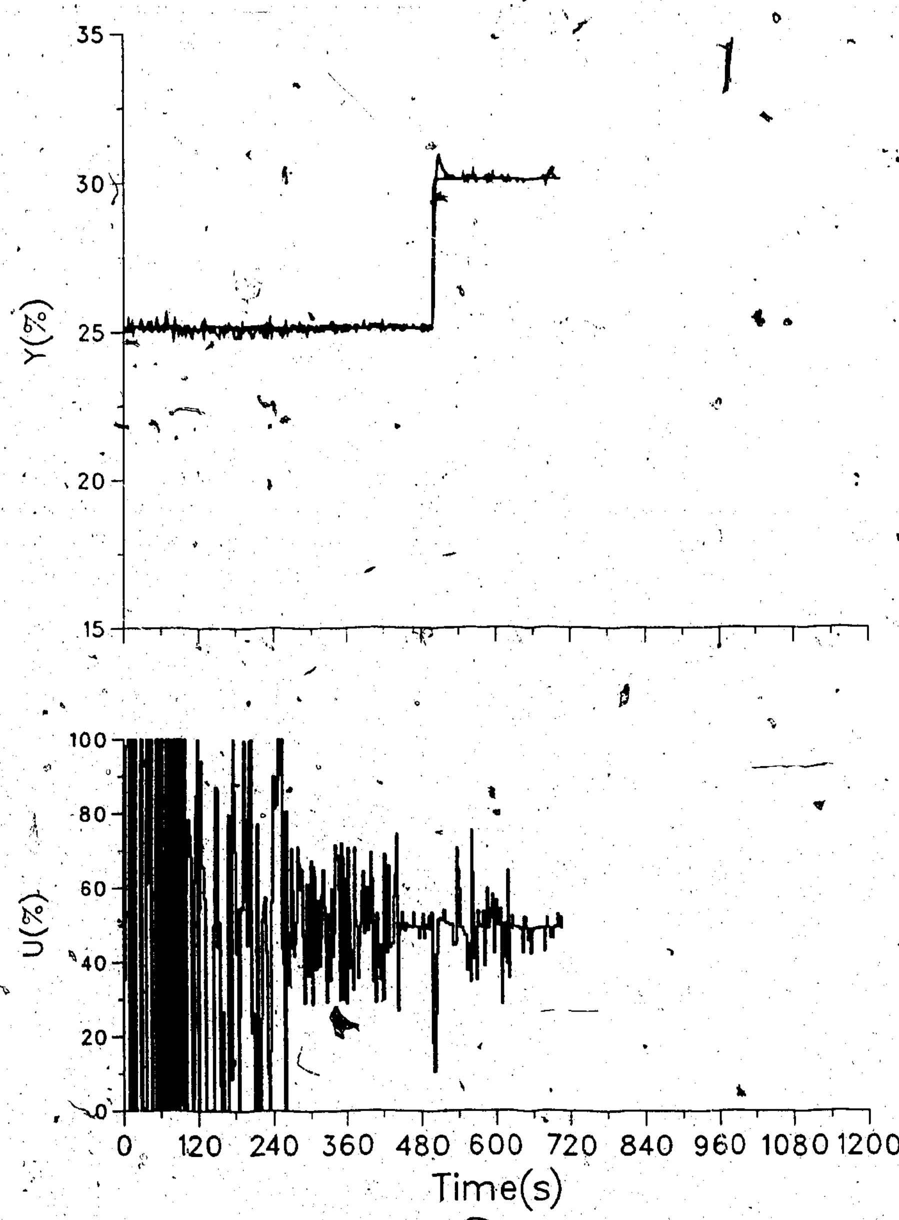


Fig. 9.17 Use of Output Cycling Limits to Prevent Large Oscillations in Controller Output (C_1_5.2)

specifications for desired overshoots.

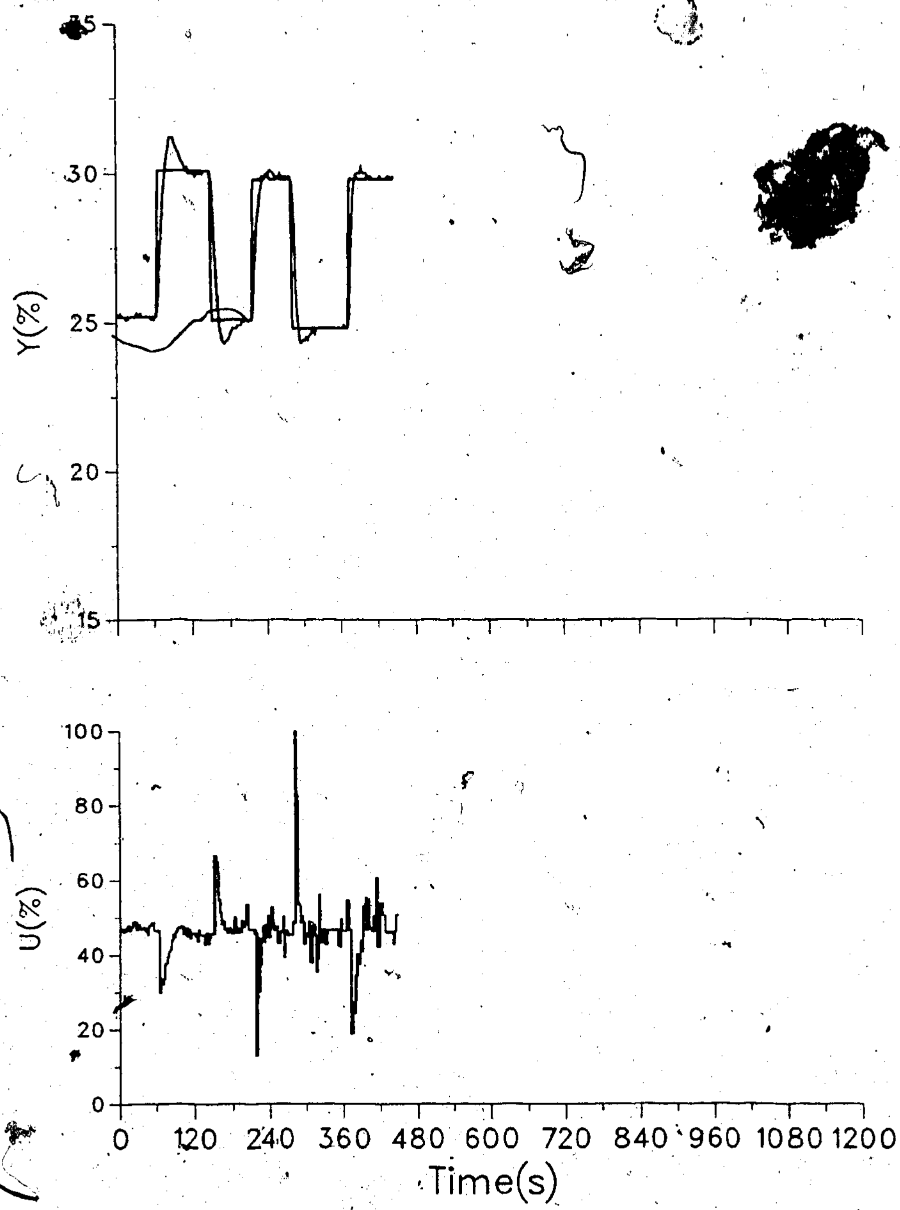


Fig. 9.18 Clamping of PID Parameters to Prevent the Gain Wind-Up Problem

9.2.7 I/O Data Validation

Several techniques are available. For example, as a safeguard against loss of measurements due to mechanical or electronic failure, maximum allowable or expected limits of change in u(k) or y(k) per sampling period should be incorporated into the control design. Changes in controller output can be limited to prevent upsets in downstream processes. If process measurements periodically fail, previous values should be used to temporarily provide an ad hoc measurement to maintain stable operation and avoid unnecessary shutdowns.

9.2.8 Back up Controller Parameters and Mode of Operation

In the event of tuner or predictor failure, stable back up controller parameters should be made available for transfer to the control block or predictor equation. Experimental evaluation of the model 6355 controller with the nonlinear plant demonstrated this need. Sometimes, when recommended PID parameters were transferred to the control block, control became unstable. With no back up PID parameters available, the controller had to be switched to manual mode and stable PID values entered into the controller's database manually. By contrast, with the other controllers, it was only necessary to switch to the back up mode (Note that "expert systems" could be used to make this decision on-line).

9.3 Initialization

All of the controllers evaluated in this study required initial estimates for parameters related to self-tuning or adaptive predictive operation. Starting values for controller parameters, closed loop time scales, sampling rates for control and adaptive prediction and initial values for adapted predictor parameters were the most important.

To minimize the amount of knowledge and effort needed for commissioning, all three controllers provided initial open loop identification options to determine the various required initial parameters. The common factor in using these options was the specification of limits for the change in u(k) used to perturb the process and maximum allowable limits on subsequent deviations in process output. The second limit is important because it prevents process outputs from changing by unexpectedly large amounts (i.e. for integrating type and/or high gain processes).

The initial test option for the GMV controller, shown in Figure (9.19) used two open loop perturbation sequences. A single user specified bump in controller output (% span of u(k)) was used to perform a Cohen and Coon step test from which estimated τ_d, τ_p and K_p were used to determine T_s, τ_{sp}, K_c, τ_i and d. A second square wave perturbation signal in u(k) was then used to provide excitation to determine $\theta(k)$ for adaptive prediction. From the test; $R(z^{-1})$ (based on τ_{sp} and T_s); $Q(z^{-1})$ (based on T_s, K_c and τ_i) and $\theta(k)$ were obtained with the specification of only two parameters: deviation in u(k) and maximum allowed deviation in y(k).

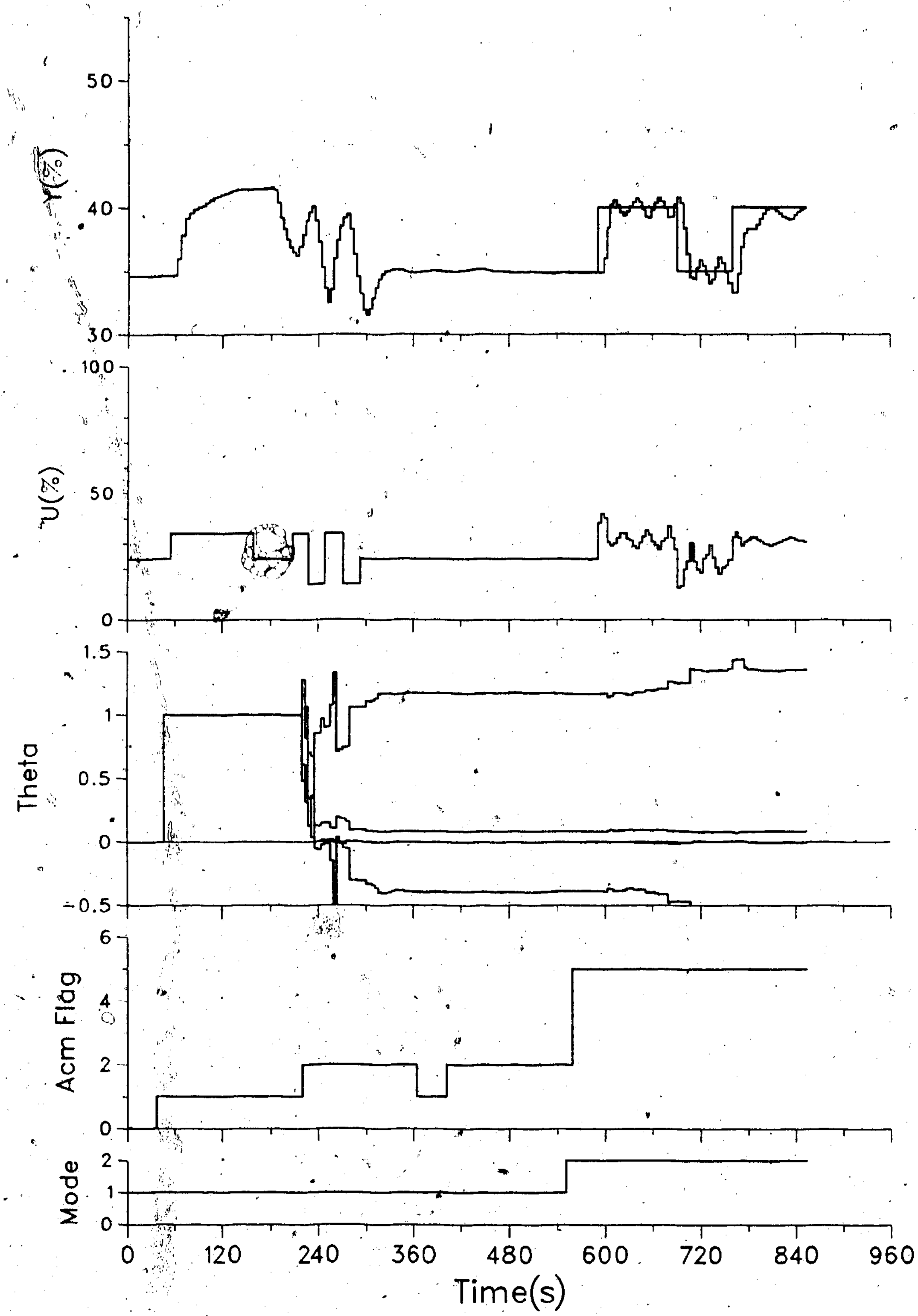


Fig. 9.19 Open Loop Identification Followed by Closed Loop Control

9.4 Ease of Use

The practical success of <u>any</u> controller is heavily dependent on how easy it is to use. The basis for closed loop self-tuning (performance index) should be measurable and easily understood. The Exact controller satisfies this requirement using limits on overshoot and damping. Because these two quantities are easily measured, an operator has immediate feedback of tuner performance. Turnbull's model 6355 controller, however, does not specify what closed loop criteria are used for the adjustment of PID values. It is difficult to accept the controller on "blind faith" alone particularly when the tuner periodically recommends unstable PID parameters. The GMV controller currently lacks a robust self-tuning loop to adjust parameters in P, Q and/or R. Before such a loop can be designed, parameters within these discrete transfer functions must be related to measurable closed loop performance criteria (directly or indirectly). An attempt to indirectly relate 1/Q to closed loop overshoot (through adjustment of K_c in 1/Q) demonstrated adequate tuning performance and the need for further work.

While addition of practical features can significantly increase a controller's useful range of application, they must be designed to require a minimum amount of knowledge and effort for use. For example, the Exact controller had several useful features that were easily implemented. One feature's effect on closed loop performance, directionally varying derivative action, however, was not easily understood and therefore not very useful. Subsequent Foxboro products with Exact tuning, such as the 760 series controllers have eliminated this option.

Another important factor is the process operator interface design. Both industrial controllers evaluated in this study used hand held "configurators" or "programmers" to display/modify parameters (one at a time) within the controllers' databases. Foxboro's Exact was superior in most aspects. Information was functionally organized into four tables and accessed using direct retrieval or scrolling. In contrast, the TCS 6355 used two character mnemonics (not necessarily easily understood). Direct retrieval of parameters was only allowed from within the current "page" or table of parameters. To access information in other pages, the operator had to first switch pages. This limitation was critical when control became unstable due to poor PID values.

For the experimental work with both industrial self-tuning PID controllers, supervisory software was designed/implemented by the author to provide a more efficient process-operator interface. Using these programs allowed for faster retrieval and modification of database information, particularly for the model 6355 controller. Turnbull's 6355 controller is provided with an excellent serial communications protocol, specifically intended for supervisory communications. The supervisory programs were also used to configure the controllers for specific types of experimental runs and archiving for subsequent documentation of performance.

The multi-tasked design of the A^3 control system allowed for the implementation of a lower priority process operator interface task. All features and options were accessed using menus and advanced terminal I/O features available with the QNX operating system. The current interface design allows for considerable flexibility in configuring the A^3 control system. A commercial implementation would restrict access to users (operators versus design engineer) so that only the necessary information for operation would be available.

9.5 Summary

The most important differences between the three evaluated controllers were due to design. The Foxboro Exact controller performed well over a wide range of process conditions, but tuned only once per cycle. Also, the inherent PID algorithm does not compensate for time delays nor does it offer the advantage of predictive control. The TCS 6355 acted as an adaptive advisor rather than a self-tuning controller. Thus, the operator had to decide when to implement the recommended PID parameters, and the confidence factor provided by the controller was not always a good indicator of subsequent process performance. The GMV controller as originally defined by Clarke and Gawthrop (Clarke and Gawthrop 1979) showed the advantages of an adaptive predictive control system, but did not have a direct self-tuning mechanism. There are a considerable number of parameters to specify for the A^3 controller, but experience with the Exact and Auto-Tuning controllers has shown that the number of operator initialized values must be minimized. The A^3 system was designed so that an operator must specify only two (2) parameters for startup of the GMV controller (see section 8.1.4).

The recommended approach is to combine an adaptive predictive control strategy of say the GMV, with a self-tuning performance mechanism such as used by the Foxboro Exact plus the best of the "practical features" included in the individual controllers, eg. nonlinear gain compensation, initialization, etc. The potential performance of such a "combined controller" was illustrated by the extended, self-tuning GMV implemented as part of this work.

10. Conclusions and Recommendations

The primary objective of this thesis was to functionally and experimentally evaluate the performances of two industrial, adaptive PID controllers and, based on those results and a requirements definition, design and implement a practical adaptive control system using Clarke and Gawthrop's (1979) Generalized Minimum Variance controller. By experimentally comparing this academic controller's performance with the two industrial models, areas for future improvement were identified.

10.1 Conclusions

The results of theoretical analysis and experimental evaluations for the three controllers used in this study are summarized below:

Foxboro Company's Exact Controller

1. The Foxboro Company's Exact controller is an example of a self-tuning PID controller that provides good adaptive control for a wide range of applications. The controller automatically adjusts PID parameters, once-per-transient, based on user specified overshoot and damping following set point changes or load disturbances. Given reasonable initial PID values, the Exact's self-tuner requires five to ten tuning transients before giving user specified performance.

When initial PID values are not available, an open loop PRETUNE option can be used that gives good initial estimates. Results demonstrate that the Exact controller delivered good performance in controlling both a simple first order process and a highly nonlinear plant with directionally varying dynamics.

2. Foxboro has also implemented a number of practical features designed to handle specific process conditions. Nonlinear process gain compensation minimizes the number of tuning transients required for convergence and reduces the effects of process nonlinearities on dynamic closed loop control. A low pass digital filtering option is also provided to remove process measurement noise. Clamping of PID parameters, automatic detuning in the event of high frequency oscillations in u(k) and backup PID parameters were shown to improve the Exact's overall performance.

3. The operator can not specify overdamping for set point or load disturbance transients. As a result, the tuning approach used is not suitable for processes not easily forced to oscillate. Overshoot specifications on such processes result in large controller gains and unacceptably large variations in controller output.

4. The controller does not use dead time compensation. For processes with large time delays, the controller will detune automatically and this will result in slowly damped oscillatory control. If changes in time delay are significant relative to closed loop period, the operator will have to reuse the initial PRETUNE option to estimate WMAX, a parameter required for the pattern recognition algorithm.

Turnbull Control System's Auto-Tuning Controller

1. Turnbull Control System's Auto-Tuning PID controller acts as an advisor which recommends PID parameters which can be transferred by the operator to the PID control law. Based, on the tracking ability of a statistically determined model, a confidence factor (0-100%) is generated to assist the operator in making the transfer decision. To obtain high (≃50-70%) confidence levels, an automatic set point perturbation option is used to generate the necessary excitation for modelling. The use of this option always results in unacceptably large variance in controller outputs (set point perturbations of 1-2%) and the "confidence factor" was not always a good indicator of how well the new parameters would perform.

2. The performance criteria on which the recommended PID constants are based is not specified, but the controller typically gave critically damped responses to step changes in set point when controlling the simple first order process.

3. The Auto-Tuning controller did not give good tuning performance for the nonlinear plant used in this work. The statistical model apparently could not track process nonlinearities resulting in low confidence factors and operator uncertainty when transferring recommended PID values. Resulting control performance was sometimes unstable following PID updates.

4. Besides the digital filtering option for process measurements, the Auto-Tuning controller lacked practical features such as backup PID parameters and nonlinear process gain compensation. Based on experimental results, the Auto-Tuning controller does not satisfy the requirements for a practical adaptive controller.

Academic GMV Controller

1. Clarke and Gawthrop's 1979 controller can be interpreted as a non-adaptive, fixed gain, 1/Q predictive control law acting on "set point minus predicted process outputs", or predicted error. Predicted outputs are calculated using parameters estimated by an implicit recursive scheme.

2. Experimental results demonstrate the advantage of using predicted set point error over measured set point error. The use of prediction introduces phase lead into the closed loop much like the derivative action of a PID controller. A fixed parameter PI control law acting on predicted set point error required less tuning in controlling a nonlinear plant than a conventional PI controller.

3. Experimental evaluations demonstrated robust long term operation of an "Improved Least Squares" algorithm for recursive estimation of predicted outputs. Useful ON-OFF criteria effectively suspended estimation during periods of low excitation, preventing problems associated with parameter drift and measurement noise.

4. To overcome problems associated with incorrect assumptions about process model structure, the use of Q or P filtering is required. In this work, an inverse PI controller structure was chosen for Q to guarantee controller offset rejection and allow for the use of conventional tuning rules to adjust Q. Control performance on a nonlinear process demonstrated that the inverse PI law for Q required a self-tuning mechanism to automatically adjust the controller gain and reset time. An ad-hoc scheme was implemented to tune controller gain based on user specified desired overshoot for set point transients. This tuner improved the GMV's overall performance by maintaining stable closed loop control throughout the operating range of the nonlinear plant. Additional design effort is required to improve the robustness of this second closed loop self-tuning mechanism for a wider range of applications. Another alternative is to use the Foxboro Exact controller as a "piggyback" tuner in conjunction with the GMV adaptive predictive controller. The Exact could be used to tune effective PI constants within the 1/Q discrete filter of the GMV controller.

5. Several features were implemented that improved the academic controller's overall performance. Nonlinear process gain compensation was used to automatically

adjust the gain of the 1/Q filter based on changes in estimated static process gains. Use of this option decreased the amount of closed loop tuning required to maintain user specified overshoots and closed loop stability. Exponential filtering of set point changes was implemented via the R transfer function to allow the operator to specify response times for set point transients. This filtering also minimized upsets in the estimated parameters used to calculate predictions.

6. Although the GMV controller required the specification of a large amount of information prior to startup, an initial open loop identification option was implemented that reduced the number of user specified parameters to two values. A Cohen and Coon step response procedure was used to obtain initial PID parameters based on 25% overshoot and one quarter decay ratio for set point transients. Results show that for general applicability, the initial PID values should be based on more conservative criteria. Following the step test, estimated transport delay, calculated sample time and the default second order process model structure were used to automatically determine the structure of the predictor equation. Recursive identification of predicted outputs was automatically started and a user specified perturbation used to provide the excitation.

10.2 Recommendations

To improve the GMV controller's general range of applications, the following areas for future work are recommended:

1. Dynamic performance of the adaptive predictive scheme must be improved. Steps should be taken to minimize the effects of unmeasured disturbances on the estimated parameters used to calculate predictions, i.e. suspension of process model identification when load disturbances enter the closed loop, constraints on estimated parameters. Known process nonlinearities should be used to modify measured I/O data prior to recursive estimation so that the recursive identification algorithm does not have to track large changes in estimated parameters.

2. Introduce dynamic dead-time compensation for processes with variable transport delays.

3. Relate the specification of P,Q and R to measurable, time domain based performance characteristics. The performance characteristics must be suitable for a

general class of applications. For example, overshoot and damping, used by the Exact controller's self-tuner are not suitable for all processes. By incorporating provisions for the specification of overdamped responses, the Exact controller would increase its useful range of applicability.

4. Develop a robust self-tuning mechanism to adjust parameters within P, Q and/or R based on the specified performance criteria and measured process values. This work is needed to make GMV a practical self-tuning adaptive predictive controller.

5. Consider an implementation of the integrating form of GMV based on the use of the CARIMA process model formulation. The integrating GMV controller should involve considerably less design and implementation effort, particularly in terms of rejection of prediction offset and elimination of controller offset.

11. References

Anderson, B.D.O., "Adaptive System, Lack of Persistency of Excitation and Bursting Phenomena", Automatica, Vol. 21, No. 3, pp. 247-258, 1985.

Astrom, K.J., "Computer Control of a Paper Machine - An Application of Linear Stochastic Control Theory", IBM J. Res. and Design, Vol. 11, pp. 389-400, 1967.

Astrom, K.J., "Introduction to Stochastic Control Theory", Academic Press, New York, 1970.

Astrom, K.J.; Wittenmark, B., "On Self-Tuning Regulators", Automatica, Vol. 9, pp. 185-199, 1973.

Astrom, K.J.; Hagander, P.; Sternby, J., "Zeros of Sampled Data Systems", Automatica, Vol. 20, No. 1, pp. 31-38, 1984.

Bierman, G.J., "Measurement Updating Using the U-D Factorization", Automatica, Vol. 12, pp. 375-382, 1976.

Bristol, E.H., "The Design of Industrially Useful Adaptive Controllers", ISA Transactions, Vol. 22, No. 3, pp. 17-25, 1983a.

Bristol, E.H., Experimental Analysis for Engineering of

Adaptive Designs", Chemical Engineering Progress, pp. 69-79, June 1983b.

Carlson, N.A., "Fast Triangular Formulation of the Square Root Filter", AIAA J., Vol. 11, No. 9, pp. 1259-1264, 1973.

Clarke, D.W., "Generalized-Least-Squares Estimation of the Parameters of a Dynamic Model", Proc. IF[illegible] Symposium on Identification in A[illegible] Control Systems", Prague, paper 3.17, 19[illegible]

Clarke, D.W., "Self-Tuning C[illegible] of Non-Minimum Phase Systems", Autom[illegible]ol, 1984. 20, No. 5, pp. 501-517.

[illegible]ke, D.W.; Gawthrop, P.J., "Self-Tuning Controller", Proc. IEE, Vol. 122, No. 9, pp. 929-934, Sept. 1975.

Clarke, D.W.; Gawthrop, P.J., "Self-Tuning Control", Proc. IEE, Vol. 126, No. 6, June 1979.

Clarke, D.W.; Hodgson, A.F.J.; Tuffs, P.S., "Offset Problems and K-Incremental Predictors in Self-Tuning Control", Proc. IEE, Vol. 130, Pt D, No. 5, Sept. 1983.

Dumont, G.A., "Self-Tuning Control of a Chip Refiner Motor Load", Automatica, Vol. 18, No. 3, pp. 307-314, 1982.

Fortescue, T.R.; Kershenbaum, L.S.; Ydstie, B.E., "Implementation of Self-Tuning Regulators with Variable Forgetting Factor", Automatica, Vol. 17, No. 6, pp. 831-835, 1981.

Gawthrop, P.J., "Some Interpretations of the Self-Tuning Controller", Int. J. Control, Vol. 31, No. 5, Sept. 1977.

Goodwin, G.C.; Leal, R.L.; Maynes, D.Q.; Middleton, R.H., "Rapprochement Between Continuous and Discrete Model Reference Adaptive Control", Automatica, Vol. 22, No. 2, pp. 199-207, 1986.

Goodwin, G.C.; Sin, K.S., "Adaptive Filtering Prediction and Control", Prentice-Hall Inc., New Jersey, 1984.

Guttman, I.; Wilks, S.S.; Hunter, J.S., "Introductory Engineering Statistics", 3rd. Ed., John Wiley and Sons, Toronto, 1982, pp. 93-94.

Hasting-James, R.; Sage, M.W., "Recursive Generalized Least Squares Procedure for On-Line Identification of Process Parameters", Proc. IEE, Vol. 116, No. 12, pp. 2057-2062, Dec. 1969.

Higham, E.H., "A Self-Tuning Controller Based on Expert Systems and Artificial Intelligence", Internation Conf. "IEE Control '85", University of Cambridge, 1985.

Isermann, R., "Parameter Adaptive Control Algorithms- A Tutorial", Automatica, Vol. 18, No. 5, pp. 513-528, 1982.

Kalman, R.E., "Design of a Self-Optimising Control System", Trans. ASME, pp. 468-478, 1958.

Kraus, T., "Self-Tuning Control: An Expert System Approach", Houston ISA Show, 1984.

Kraus, T.W.; Myron, T.J., "Evaluation and Performance of a Self-Tuning PID Controller", Paper Presented at the 1983 ISA Conf., 1983.

Kraus, T.W.; Myron, T.J., "Self-Tuning PID Controller Uses Pattern Recognition Approach", Control Engineering, 1984.

MacGregor, J.F., Correspondence: "Discrete Stochastic Control with Input Constraints", Proc. IEE, Vol. 124, No. 8, August 1977.

Minter, B.J., "On-Line Estimation Techniques - A Review of Practical Problems and Solutions for Recursive Least Squares", Internal Department Report, Department of Chemical Engineering, University of Alberta, 1987.

Morris, A.J.; Nazer, Y.; Wood, R.K.; Lieuson, H. "Evaluation of Self-Tuning Controllers for Distillation

Column Control", Proc. 6th. IFAC/IFIP Conf. on Digital Computer Applications to Process Control, Dusseldorf, Oct., 1980.

Nachtigal, C.L., "Adaptive Controller Performance Evaluation: Foxboro Exact and ASEA Novatune", ACC Conf., Seattle, pp. 1428-1433, 1986a.

Nachtigal, C.L., "Adaptive Controller Simulated Process Results: Foxboro Exact and ASEA Novatune", Acc Conf., Seattle, pp. 1434-1439, 1986b.

Panuska, V., "An Adaptive Recursive Least Squares Identification Algorithm", Proc. 8th. IEEE Symp. on Adaptive Processes, Decision and Control, Pennsylvania State University, Nov., 1969.

Phillips, C.L.; Nagle, Jr., H.T., "Digital Control Systems Analysis and Design", Prentice-Hall Inc., New Jersey, pp. 344-353, 1984.

Potter, J.E., "New Statistical Formulas", Space Guidance Analysis Memo 40, Instrumentation Lab, MIT, Cambridge, Mass., April, 1963.

Proudfoot, C.G.; Gawthrop, P.J.; Jacobs, O.L.R., "Self-Tuning PI Control of a pH Neutralization Process", IEE Proc. D, Control Theory and Appl., Vol. 130, No. 5, pp. 267-272, 1983.

Sripada N.R.; Fisher, D.G., "Improved Least Squares

Identification for Adaptive Controllers", Proc. ACC Conf., pp. 2027-2037, Minneapolis, June, 1987a.

Sripada N.R.; Fisher, D.G., "Improved Least Squares Identification", Int. J. Control, Accepted for Publ., 1987b.

Stephanopoulos, G., "Chemical Process Control: An Introduction to Theory and Practice", Prentice-Hall Inc., New Jersey, p.596, 1984.

Tham, M.T., "Some Aspects of Multivariable Self-Tuning Control", Dept. of Chem. Eng., University of Newcastle-Upon-Tyne, May, 1985.

Tuffs, P.S.; Clarke, D.W., "Self-Tuning Control of Offsets: A Unified Approach", Proc. IEE, Vol. 132, Pt D, No. 3, May, 1985.

Vogel, E.F.; Edgar, T.F., "Application of an Adaptive Pole-Zero Placement Controller to Chemical Process with Variable Dead Time", Proc. ACC, Virginia, June, 1982.