Online Strategies for Adaptive Tuning of PI Controllers in HVAC Systems

Guang Qu

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montreal, Quebec, Canada

December 2002

ABSTRACT

Online Strategies for Adaptive Tuning of PI Controllers in HVAC Systems

Guang Qu

To improve the setpoint tracking and disturbance attenuation properties of the Discharge Air Temperature (DAT) control loop in HVAC systems, online adaptive PI tuning strategies are developed. The feasibility of adaptive control strategies for DAT systems is first studied by formulating and solving three adaptive control problems. These are: a model following adaptive (MFA) control problem, a control-model based adaptive (CMBA) control problem and an H_∞ adaptive PI (HIAPI) control problem. The DAT system is modelled as a first-order-plus-dead-time (FOPDT) model. Both MFA and HIAPI controls are model-based techniques whereas CMBA is a model-less control strategy. Simulation runs are made to compare the output responses of the above control strategies with constant gain PI control, LQR based adaptive control, and Ziegler-Nichols based adaptive PI control. Results show that MFA, CMBA and HIAPI control strategies perform well under a wide range of operating conditions and disturbances.

Motivated by these computer simulation results, two control strategies namely: CMBA and HIAPI are chosen as ideal candidates for implementation in a HVAC test facility. Both CMBA and HIAPI control algorithms are written in Powers Processor

Control Language for online implementation. Several experimental tests are conducted under variable setpoints, variable disturbances, and multiple loop interactions.

Experimental results show that CMBA control tuning method is simple to implement (does not require system identification) and robust to changes in setpoints and operating conditions. It is found to give stable and smooth response. The HIAPI tuning strategy requires system identification; as such it is sensitive to changes in operating conditions. Results show that HIAPI tuning control also gives good output responses under a wide range of operating conditions.

Noting that the current practice in HVAC control is either constant PI control or PI control with on-demand tuning, the online adaptive tuning strategies proposed in this thesis are of significant practical value in HVAC systems.

ACKNOWLEDGEMENTS

First of all, I would like to express my deep gratitude to my supervisors, Dr. M. Zaheer-uddin and Dr. R. V. Patel for their guidance, encouragement and financial support during this research work.

I would like to thank Mr. J. Hrib, Mr. L. Demers and Mr. J. Payer for their support in preparation of the experimental equipment.

Finally, I deeply appreciate the encouragement and moral support of my family and friends.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a	system pole of FOPDT model in discrete-time domain
$\hat{a}$, $\hat{b}$	elements of $\hat{\theta}$
aOfModel	$\hat{a}$ of the FOPDT model for experiment
A, B, C, D	matrixes
A_{cm}, B_{cm}	adjustable coefficients for tuning ue
A_m, B_m	adjustable coefficients for tuning um
b	system gain of FOPDT model in discrete-time domain
bOfModel	$\hat{b}$ of the FOPDT model for experiment
C	system parameter as matrix
CCITS	entering air temperature for experiment
CCOTS	discharge air temperature for experiment
CFM	cubic feet per minute
CMBA	control-model based adaptive
CV	constant volume
CWFRG	chilled water flow rate for experiment
CWRTS	chilled water return temperature
CWSTS	chilled water supply temperature for experiment
C_0	initial estimate of matrix C
$C_{1, 2, \ldots, 5}$	elements of matrix C
DAT	discharge air temperature

DDC	direct digital control
DEG.C	$^{\circ}C$
DPS	total duct supply air pressure
DPS001	duct supply air pressure to room 1
DPS002	duct supply air pressure to room 2
D(z)	characteristic polynomial
e	the difference between T_{set} and T_a that is $T_{set} - T_a$
em	the difference between T_{set} and T_{am}
$e(k;\theta)$	the difference between $y(k)$ and $\hat{y}(k)$
EMC	Energy Management Control
EMCS	Energy Management Control System
FF	feedforward
FFPI	feedforward proportional-integral
FR	airflow rate for entering air
FOPDT	first-order plus dead-time
gpm, GPM	gallons per minute
G(z)	transfer function
HIAPI	H_∞ adaptive PI
HVAC	heating, ventilating, and air conditioning
H_∞	Hankel norm used to measure control system properties
IdentFlag	status of identification updating
IMC	internal model control
ISE	integral square-error criterion

J	performance criterion
k	time in samples
k_s	the system gain of FOPDT system
k_u	an adaptive gain in CMBA PI control
$K_d(z)$	z transform of a controller in discrete system
K_{id}	integral gain of $K_d(z)$
K_m	an adaptive gain in MFA PI control
K_p	proportional gain of $K(s)$
K_{pd}	proportional gain of $K_d(z)$
K_{pdc}	mid-parameter for K_{pd} in CMBA PI control
$K_{pd\max}, K_{pd\min}$	bounded values of proportional gain for PI controller
K_{pdmc}	mid-parameter for K_{pd} in MFA PI control
$K(s)$	PID controller of continuous system
$KG(z)$	transfer function for applying the Nyquist criterion
l	dead-time in samples for FOPDT model
L	a 2×1 matrix
LofModel	dead-time in samples for FOPDT model for experiment
LQG	linear quadratic Gaussian
LQR	linear quadratic regulator
L_1, L_2	elements of L
MFA	model following adaptive
N	sampling size for minimization

NN	neural network
$p(s)$	transfer function of FOPDT model in continuous-time domain
P	a 2×2 matrix
PLY_1, PLY_2	scalar parameters for computing P_{21} and P_{22}
PI	proportional integral
PID	proportional integral derivative
PI-PD	a controller structure
PPCL	Powers Process Control Language a programming language used to write field panel control programs for building control and energy management functions
PRAC	pattern recognition adaptive controller
P_{11}, P_{12}, P_{21}, P_{22}	elements of P
Q	state weighting matrix
R	control weighting matrix
RLS	recursive least squares
t	time
T	sampling time
T_a	discharge air temperature as system output
T_{ae}	discharge air temperature experimental value from reference [13]
T_{am}	model output in MFA PI control
T_{a0}	entering air temperature
Tal	input air temperature
T_d	derivative time constant in $K(s)$

T_f	time constant for filter in $K(s)$
T_i	integral time constant in $K(s)$
T_s	time constant for FOPDT model in $p(s)$
T_{set}	desired discharge air temperature
Twl	input water temperature
u	controller output (as chilled water flow rate or in percentage)
ue	control model output in CMBA PI control
ue_c	mid-parameter for ue
um	control model output in MFA PI control
um_c	mid-parameter for um
$u_{\max}, u_{\min}$	bounded values of PI controller output
UD	user defined
UpdatingControl	a control signal for adaptive control taking effect
$U(z)$	the z transform of u
VALE	the difference between VALSP and CCOST
VALKD	derivative gain for experiment
VALKDEM	computed derivative gain for HIAPI control in experiment
VALKI	integral gain for experiment
VALKIE	computed integral gain for CMBA PI control in experiment
VALKIEM	computed integral gain for HIAPI control in experiment
VALKP	proportional gain of PID for experiment
VALKPE	computed proportional gain for CMBA PI control in experiment

VALKPEM	computed proportional gain for HIAPI control in experiment
VALOP	controller output in percentage for experiment
VALSP	desired discharge air temperature for experiment
VAV	variable air volume
VAVN	variable air volume variable temperature
$w(k)$	weighting function
W	weighting function matrix
y	the difference between T_{a0} and T_a
$\hat{y}$	estimated output of plant
$Y(N)$	a $(N-l)\times 1$ matrix
$Y(z)$	the z transform of y
Zeta	ς in experiment
α	$1-\gamma$ in weighted least squares or 1 in ordinary least squares
β	a constant number used as initial value for diagonal elements of P
γ	adjustable parameter for weighting function
δ	a constant error vector in tolerance change of $\hat{\theta}$ per sample
σ	a scalar parameter for computing $\hat{a}$ and $\hat{b}$
μ	a scalar parameter for computing L_1 and L_2
λ	a design parameter
τ	dead-time of FOPDT system
θ	predictive parameter value of plant
$\hat{\theta}$	estimated parameter value of plant

θ^0	true parameter value of plant
ε	a $(N-l)\times 1$ matrix
ς	adjustable coefficient for updating K_{pd} in CMBA PI control
ς_m	adjustable coefficient for updating K_{pd} in MFA PI control
ψ	a 2×1 measurand matrix
Ψ	a $2\times(N-l)$ matrix

1. Introduction

1.1 HVAC Systems

Heating, ventilating and air-conditioning (HVAC) systems play several roles to reduce the environmental impact on buildings. The primary function of HVAC systems is to provide healthy and comfortable interior conditions for occupants. The goal of HVAC control system design is to provide good control strategies to maintain comfort for the occupants of a building under variable load conditions with minimal use of energy. Reducing energy consumption becomes one of the most important aspects in HVAC control system design because of the fact that 50% of the world energy is consumed by HVAC equipment in industrial and commercial buildings [1].

Typical HVAC systems provide energy transfer via heating/cooling coils, airflow regulation to maintain air pressure or temperature via fan speed regulation or/and damper position changes, and central water supply servicing heating/cooling coils or multiple units. HVAC systems are generally operated in two modes: constant volume (CV) and variable air volume (VAV); VAV systems enjoy a significant advantage over CV systems in terms of the economy of operation [36], and result in the lower energy consumption [37].

1.2 VAV Systems

In the VAV mode, air-supply temperature is held at some constant setpoint while airflow rate is varied to satisfy zone load, in contrast to the CV mode of operation, where air-supply temperature is varied in response to zone loads while holding the airflow rate

constant. During low load periods, reduced airflow rates can be achieved in a VAV system, and this can lead to significant reduction in fan energy consumption because of the fact that fan energy consumption is proportional to the cubic power of airflow rate [2]. In [36], it is further shown that the variable air volume mode of operation in which both airflow rate and air-supply temperature are continuously modulated is a more energy efficient strategy.

1.3 DAT Systems

The discharge air temperature (DAT) system is one of the basic subsystems of HVAC systems. The discharge air temperature in a typical DAT system is maintained by controlling the chilled water flow rate through a cooling coil. The chilled water flow rate is changed by modulating a chilled water flow control valve. The mixed air which is warm enters the cooling coil and is cooled and dehumidified to desired conditions.

Because of the variability of the system conditions, such as, supply air temperature, airflow rate and chilled water supply temperature, a constant-gain PID controller cannot satisfy the requirement of tracking setpoint changes well under variable load conditions. In addition, intuitively, an adaptive regulator can change its behaviour in response to changes in the dynamics of the process and the disturbances [3]. Also, there are a number of apparent advantages in using adaptive controls in HVAC applications. Commissioning costs would be less since the commissioning period would be shorter and the commissioning procedure would demand the attention of less skilled personnel. Therefore, noting that the PID control structure for HVAC systems with industrial modular building controllers is familiar to field engineers and direct digital control

(DDC) systems with PID structure are available on the commercial market, it is of interest to develop online adaptive tuning methods to update PID parameters.

Motivated by these considerations, we are interested, in this thesis, in exploring the design of adaptive PID controllers including self-tuning control strategies for a DAT system.

This thesis is divided into six chapters. In Chapter 1, an introduction to adaptive PID control is given. The reasons for studying adaptive PID controllers including self-tuning control strategies for a DAT system are also discussed. In Chapter 2, reviews of the previous work on PID control, adaptive control and other control methods in HVAC, and real-time PID control strategies are given, and the objectives of this thesis are outlined. In Chapter 3, modelling and online identification of DAT system are developed. A second-order model and a first-order plus dead-time (FOPDT) model of DAT systems are studied. Simplified equations of online identification using the recursive least squares (RLS) method for FOPDT models are given, and simulation results for online identification of DAT systems are shown. Two heuristic adaptive control methods (model following adaptive (MFA) PI control and control-model based adaptive (CMBA) PI control) and an H_∞ adaptive PI control method for DAT systems in discrete-time domain are proposed, and simulation results for various operating conditions are given in Chapter 4. In Chapter 5, a two-zone Variable Air Volume (VAV) HVAC test facility and real-time Energy Management Control System (EMCS) is described. The developed adaptive control methods are implemented on a DAT system in the test facility. Experimental results under variable operating conditions are presented. Also, comparisons with

constant PI control are given. Conclusions and recommendations for future work are given in Chapter 6.

2. Literature Review

2.1 Introduction

As mentioned in the previous chapter, adaptive PID control strategies are a reasonable choice for DAT systems. During the last two decades, a few adaptive PID control strategies for HVAC systems have been proposed [25] – [27]. However, these methods still have some limitations. Therefore, developing some new adaptive PID control strategies is a useful area of research for the HVAC industry.

This chapter is divided into eight sections. After the introduction in section 2.1, the previous work on PID control in HVAC systems is reviewed in section 2.2, and that on adaptive control in HVAC systems is surveyed in section 2.3. Then, the work on other control methods and experimental research in HVAC systems is presented in sections 2.4 and 2.5 separately. Some literature related to real-time PID control strategies and other adaptive control is surveyed in sections 2.6 and 2.7. Finally, the objectives of the thesis are stated in section 2.8.

2.2 PID Control in HVAC Systems

In this section a survey of the published literature about the applications of PID techniques for HVAC systems is given.

In [21], Kasahara and others have presented a stability limit analysis and a new tuning method for PID controllers in bilinear systems with time-delayed feedback. The bilinear systems they considered are variable air volume (VAV) systems, which are defined as air-handling units that use variable airflow rate to satisfy the heating,

ventilating, and air-conditioning (HVAC) operation requirements – such as indoor temperature (thermal comfort.) In this reference, they have developed a normalized bilinear model of VAV systems by identifying the energy flow to and from the environmental space and presented a generalized parameter analysis of the stability limits by simulation. The results of the analysis reveal that controller parameters can be determined by assuming that the plant is linear. The proposed tuning method of PID controller for bilinear systems is that controller parameters selected by well-known methods must be modified slightly by finding an important gain reduction factor for practical applications. The main limitation for the online application of the tuning method is that it requires extensive computations to find an important gain reduction factor for practical applications.

In [22], Wang and others have proposed a PID controller design method that achieves high performance for a wide range of linear self-regulating processes. The PID tuning rules were developed using a second-order plus dead-time modeling technique and a closed loop pole allocation strategy. The technique was applied to processes with various dynamics, including those with low- and high-order, small and large dead-time, and monotonic and oscillatory responses. Simulation examples and comparisons with Ho's gain and phase margin method (GPM) [41] are presented and the results show that improved performance can be obtained. Also, real-time experiments were carried out both in the laboratory and in the industry. The test in the laboratory was conducted on a Dual Process Simulator KI 100 manufactured by KentRidge Instruments, and the results from using both the proposed tuning method and Ho's method for one oscillatory process were presented. The industrial test was performed on an HVAC digital control system

called *Enflex*, for a supply air pressure loop and a zone air temperature loop, in the Supersymmetry Services PTC LTD, Singapore. The process responses for both the proposed tuning method and Åström's modified Ziegler and Nichols method [40] were presented. The experimental results show that improved performance is achieved by using the PID tuning method developed by Wang et al.. The main limitation of the tuning method is that robustness to uncertainties in the models is not considered and uncertainties in the models certainly exist and have an effect on performance since an exact cancellation method is used.

In [23], Kasahara and others have developed a new design and tuning method in which the gains of a robust PID controller for HVAC systems are obtained by solving a two-disk type of mixed sensitivity problem. The PID gains obtained by the conventional Ziegler-Nichols rule were modified by applying this technique. To illustrate the method, the temperature control of a single-zone environmental space and the HVAC plant which was approximated by a first-order lag plus dead-time system was considered. The numerical simulation and the experiments on a commercial-size test plant for air conditioning were presented. The results showed that reduction of PID gains was required for robustness in contrast to those given by the Ziegler-Nichols rule [23]. The study showed that the robust PID gains could be expressed as simple linear functions of the ratio of the dead-time to the time constant [23]. However, for every plant, the use of this method requires the determination of six parameters off-line to compute the three gain reduction factors.

In [24], Krakow has proposed the relationship between the sampling interval and digital PI control system performance by using experimental and simulated response

characteristics for a PI-controlled mixing valve air heating system. The PI tuning rules were specified based on the analytical and experimental study by Krakow et al. [5], [6], Krakow and Lin [7], and Hussein [8], for the first-order system (without storage) and second-order system (within storage.) The conclusion the author made is that, long sampling intervals may yield more satisfactory response characteristics, than short sampling intervals if the system is tuned appropriately. Appropriate tuning implies using PI coefficients based on (non-conventional) theory developed specifically for long sampling intervals. A long sampling interval implies a complete response of the controlled variable to the change in the control signal. The limitation of the paper is that the results obtained are specific to a system and cannot be generalised.

In [25], Seem presented a method for implementing a new pattern recognition adaptive controller (PRAC) developed through optimization, for automatically adjusting the parameters of PI controllers while under closed loop control. Depending on patterns of the closed loop response, PRAC will determine the parameters of the digital PI controller used in a HVAC system. Simulation results subject to random noise and load disturbance modeled by either one or zero are presented. Some field test results are also given. The conclusion Seem made is that PRAC is robust, easy to use and has low computational and memory requirements. From the results presented, it can be noted that PRAC responses are either sluggish or oscillatory. It takes a long time to reach stable state. The limitation of this study is that PRAC was developed for systems that can be characterized by a first-order plus dead-time model with the ratio of dead time to time constant between 0.25 and 1, and the ratio of sampling time to time constant is between 0.1 and 1 for "good control."

In [26], Nesler has reported the implementation of three automated controller-tuning methods, which include a computer-assisted controller tuning program, an automatic tuning controller, and a self-tuning controller for HVAC processes. A computer-assisted controller-tuning program allows novice users to tune DDC controllers reliably and quickly. An automatic tuning controller fully automates the open-loop step-test tuning procedure and has self-monitoring capability that leads the automatic tuning routine, once initiated, can operate without supervision. Nesler concluded that automated controller tuning addresses two fundamental problems in HVAC control applications: the time-consuming initial tuning of the controller and the requirement for periodic controller retuning as system loads change. A self-tuning control system adjusts control parameters in real-time using closed loop control data. Experimental results from applying the self-tuning controller to a discharge air temperature control loop are given. Nesler also mentioned that unmodelled load disturbances, drifting parameters and actuator nonlinearity are limitations of self-tuning controllers.

In [27], Nesler has presented the implementation of a self-tuning controller to control typical HVAC processes (the model can be considered as a first-order plus dead-time). The self-tuning controller consists of five independent software blocks. The five blocks include an automatic tuning routine used to establish initial parameter estimates, a recursive least-squares estimator for making parameter estimates on-line, a controller design block, which computes the gains of PI controller depending on the new parameter estimates, a PI controller, and a performance monitor, which supervises the self-tuning controller operation. The open-loop step test method is used for the automatic tuning routine. The PI gains are computed by minimizing the integrated absolute error (IAE). In

addition, the performance monitor is introduced to determine when retuning is required. The use of the performance monitor can also increase system flexibility and robustness. The main limitation of RLS (Recursive Least Squares) estimation mentioned in [27] is that self-tuners can occasionally fail to produce useful estimates under certain conditions such as self-tuning control loops subjected to large and unmodeled load disturbances. Neseler also mentioned that these types of disturbances are common in HVAC processes.

2.3 Adaptive Control in HVAC Systems

A survey of the published literature about adaptive control in HVAC systems is presented in this section. References [25] – [29] are the relevant papers. The contributions of [25] – [27] are also related to PID control in HVAC systems and were discussed in the previous section.

In [28], Åström and others have presented a tuning method based on the relay feedback for a general digital controller. The method was developed for tuning digital control laws and the control design method used for the control law in the tuning method is based on pole placement. There is an interesting feature in the tuning method that the sampling period and the desired closed loop poles are determined from an experiment with relay feedback. A conclusion from extensive simulations is stated that the method works very well for low-order systems with time delay. Two test results applied to HVAC plants are presented. A limitation mentioned in [28] is that the direct approach does not work well for systems with an excess large pole in their continuous time expressions.

Chen et al. [29] have presented an adaptive robust control scheme applied to a single-zone HVAC system. A single-zone HVAC system with modelling uncertainty (which includes thermal storage effect, heat and moisture generation, and outside temperature and humidity variation) is established for a single zone HVAC system in a generic room. It is a nonlinear system, and the uncertainty is assumed bounded but the bound is unknown. Then, a class of adaptive robust controls which was originally designed by Corless and Leitmann [9] is used to achieve the control objective to drive the room's state (which is related to its temperature and humidity) into a comfort region. A comparison between the use of on-off control and the use of the adaptive robust control from simulation results is given. Chen et al. conclude that simulation results depict a satisfactory transient performance under a significant deviation of the initial state from the comfort region in the sense of maintaining small overshoot. A drawback mentioned in [29] is that the steady state performance has certain oscillations.

2.4 Other Control Methods in HVAC Systems

Several researchers have applied other control methods to HVAC systems, such as a combined feedforward and feedback control approach with a general regressive neural network (GRNN) method [30] – [32], a predicate neural network method [33], a predicate control method [34], optimal control [35] – [36], nonlinear control [37] and direct digital self-tuning control [38].

In [30], Ahmed et al. have proposed a combined feedforward and feedback control approach for a laboratory HVAC system. A general regression neural network (GRNN) is utilized in the feedforward component for HVAC system identification and

control, while the feedback component provides a control signal to offset any steady-state error. A typical variable-air-volume laboratory HVAC control system is considered by the authors. They studied pressure control in this paper. They also show the simulation results obtained by using a laboratory simulator. The simulation results indicate that the combined approach performs better than the feedback approach over widely varying operating conditions and different damper characteristics. The main limitation of this paper is that the simulation results were obtained for six cases of damper characteristics separately, that is, the results do not include the dynamic situation of the damper characteristics that exist in real systems.

In [31], Ahmed and his colleagues have presented the second application of the combined feedforward and feedback control approach. In the application, the internal heat generation in the laboratory space is considered as a disturbance. The implementation for temperature control is studied by using two closed loops: a supply air flow rate control loop and an exhaust air flow rate control loop so that the laboratory pressure constraint is met. In the control sequence the supply flow rate is increased by first opening the general exhaust damper to increase the total laboratory exhaust flow rate. They show that the FFPI control loop works well under a wide range of operating condition. But in the cooling temperature control system, the controlled variable is not the room temperature, so that some offset of the room temperature may appear in steady state if the model is not exactly correct.

In [32], Ahmed et al. have presented the third application of the combined feedforward and feedback control approach. In the application, two separate disturbance sequences are considered for heating, and then the implementation for temperature

control has been studied by using two closed loops, a temperature control loop and a supply airflow rate control loop. In addition, in reference [32], they also provided a summary and recommendations for the three systems, pressure control system, temperature control system for cooling and temperature control system for heating. They concluded that the FF part requires only a single smoothing parameter to be estimated, which can be held constant for most of the HVAC processes. The proposed FFPI controller may not need to be retuned. The main limitation in the three application papers [30, 31, 32] is that the identification results of GRNN were kept constant for the simulations.

In [33], Jeannette et al. have presented experimental results of a predictive neural network (PNN) controller applied to an unstable hot water system in an air-handling unit. The PNN controller works with a PID controller. The neural network learns the system while it is operating under the PID controller, and depending on the result of the predictions, the NN used is "good" or not, the NN will or will not take control from the PID algorithm and control the processes. The term "good" means that the average of the past ten COVs, where COV is the coefficient of variation defined by Kreider and Haberl [10] goes below 0.45. When the NN is active and the average COV rises above 0.55, then the NN model needs to be updated with new data and the controller reverts to PID control. Actual laboratory testing of the PNN and PID controllers shows favourable results for the PNN controller. The main limitation of the method is that the NN needs the PID controller's support for training and works only under specified range of operating conditions.

In [34], Dexter and Haves have developed a robust self-tuning predictive controller based on the Generalized Predictive Control algorithm [11] for HVAC applications. The controller uses default values for most of its parameters and requires selection of only one commissioning parameter: the control-sampling interval. In the controller, a parameter estimator for the HVAC plant, which is based on the UD filter form of the recursive least-squares algorithm, using a simple form of variable exponential forgetting [12], is used. To implement a set of expert rules, which supervise the operation of the on-line parameter estimator and the calculation of control action, Dexter and Haves have developed special jacketing software. Also, they have used a component-based computer simulation package (HVACSIM+) to examine the behaviour of the controller in both the zone and supply air temperature control loops. The robust behaviour of the self-tuning controller is demonstrated and its superior performance to that of a manually tuned PI controller is suggested by the results. Application of the self-tuning controller in a cascaded control configuration is also discussed and the conclusion for it is given that the use of two self-tuning controllers within a cascaded control scheme worked well when care was taken to deal with the interactions that occurred between the inner and outer loops during the tuning period. The controller is too complex for implementation on existing building control hardware. Also the accuracy of the estimator over extended period of operation is not proven in the simulation.

In [35], Kasahara and others have presented a multivariable autoregressive (AR) model which is a three input/two output model obtained by using observed data for a HVAC system. They used a combination of the model and the preview control that is a linear quadratic Gaussian (LQG) optimal control with feedforward compensation to

control process variables such as indoor temperature and indoor humidity. The comparison of measured data and simulation results shows that the plant model is adequately formulated. Also, experimental results on a commercial-sized test plant with LQG control system and preview control system are presented. The conclusion in [35] is that the optimal control system based on statistical modelling with the multivariable AR model has been found to be quite useful for the control of interactive HVAC systems, and the preview control system produces excellent control under normal operating conditions compared to LQG control system. The limitations of the optimal control system based on statistical modelling with the multivariable AR model are that the modelling and the control design are very difficult to implement online, and the model obtained off-line may not match the model dynamics undergoing variable operating conditions.

In [36], Zaheer-uddin and Zheng have explored the application of a multistage optimization technique to determine optimal operating strategies for HVAC systems. Simulation results are shown for a single-zone space heating system consisting of a heat pump, a storage tank, a heating coil, a fan, and ductwork. The optimization problem was solved using a typical building operation schedule, consisting of off-normal, start-up, and normal occupied periods, assuming time-of-day rates. A conclusion is presented, for the two most widely used operating strategies, namely, constant-volume (CV) and variable-air-volume (VAV) system. It is shown that the variable-air-volume mode of operation in which both air-supply temperature and flow rate are continuously modulated is the most energy efficient strategy. The limitations of the study are that the optimal result depends on exact dynamic models, and in real systems, to track the setpoints exactly may have some difficulties.

In [37], Argiiello-Serrano and Velez-Reyes have presented a non-linear disturbance rejection state feedback controller for a HVAC system. The controller was designed by using Lyapunov stability theory and consists of a regulator and a disturbance rejection component. To reduce the effect of thermal loads other than design loads on the system, they also proposed a thermal load estimator that allows the controller to obtain on-line estimates of the thermal loads affecting the thermal space. In addition, they have shown simulation results for a variable air volume (VAV) HVAC system. The results demonstrated the potential for the controller to keep comfort levels and save energy in a variable air volume HVAC system operating in the cooling mode. The main limitation of the design for the controller is its complexity for online implementation.

In [38], Wallonborg has proposed a control algorithm for a self-tuning controller. The control algorithm depends on discrete-time process transfer function parameters, and the parameters based on the wave form of a periodic oscillation obtained with a relay feedback tuning experiment. The self-tuning controller is a general linear discrete-time controller and is designed by using pole placement based on input-output models. In addition, the self-tuning controller also has a feature that the sampling period and the desired closed loop poles are determined automatically with respect to the process dynamics and the desired closed loop performance can be easily modified by the operator. The experimental results for applying the control algorithm to a supply air temperature control and an air duct pressure control are shown. Wallonborg concluded that the algorithm has worked well in many different HVAC applications, and a substantial reduction in commissioning can be achieved compared with manual tuning of

conventional controllers. The main drawback of the algorithm is that in some cases it may be difficult to obtain the necessary steady-state conditions for a tuning experiment.

2.5 Experimental Research in HVAC Controls

Experimental work on applying control methods to HVAC systems has been done by several researchers. In [22] and [23], experimental results for applying PID tuning methods to HVAC systems are presented. In [24], experimental responses for a PI-controlled mixing valve air heating system are used to show the relationship between sampling interval and digital PI control system performance. Experimental results for applying adaptive PID control methods to HVAC systems are given in [25] – [27]. In [28], experimental results for applying general digital control (which tuned by a tuning method based on relay feedback) to a HVAC system are presented. Experimental results for applying a self-tuning controller to control supply air temperature and air duct pressure are shown in [38]. These papers have been reviewed in sections 2.2 – 2.4.

2.6 Real-time PID Control Strategies

To develop new adaptive PID control strategies, it may be useful to consider some real-time PID control strategies. For this reason, a survey of real-time PID control strategies is given in this section.

In [39], Tan and others have developed PID tuning rules for three types of typical industrial processes: First-order plus dead-time (FOPDT), integrator plus FOPDT, and integrator plus dead-time, based on loop-shaping H_∞ optimal control. In PID tuning rules, the PID-type H_∞ controller parameters are expressed as direct functions of plant

constants. A design parameter is used to obtain the PID gains, which reflects the trade-off between stability, robustness and time domain performance of the closed loop system, thus making it convenient to choose PID parameters dependent on the uncertainties of the processes. However, the development of the tuning rules is for continuous-time systems.

Åström et al. [40] have proposed two methods for automatic tuning of PID regulators with specifications on phase and amplitude margins. They have also described the procedures which need relay controls and will automatically generate appropriate test signals. They presented the responses of experiments conducted on a process with a tank. They concluded that the methods are robust and easy to use. The algorithms may be incorporated in single-loop controllers to provide an option for automatic tuning, and may also be used to initialize more sophisticated adaptive algorithms. The disadvantage of the procedures is that the estimation procedure requires the interruption of the PID closed loop.

In [41], Ho and others have presented a tuning method for PID controllers based on gain and phase margin specifications. Simple formulae for tuning both PI and PID controllers are derived by using an approximation for the arctan function they proposed. The method is particularly useful in the context of adaptive control and auto-tuning, where the controller parameters have to be calculated on-line. The rise time of the closed loop system can also be predicted using the results given in the reference, which is useful for self-diagnosis. They have also presented simulation examples to substantiate the results obtained, and made comparisons with the internal model control design for the PID controller (IMC-PID). The GPM formulae obtained in this reference can give more flexibility in addressing the robustness problem, since the IMC-PID design has only one

tuning parameter. The main limitation of the reference is that the results are valid for continuous systems.

In [42], a method for automatic tuning of PID process control parameters has been proposed. The method is established on both the parameter estimation and the weighted ISE optimal tuning method. The parameter estimation method relies on the characteristic areas of the step response. They also mentioned that the relationship between the characteristic areas and the optimal settings is approximated by a set of simple polynomials. In addition, the results of an application to a cement plant, to temperature control of a heat exchanger and to control of a tank level are shown. They concluded that by applying the proposed method to real processes, sufficiently good settings of the PID parameters without knowing the exact form of the process transfer function can be obtained. The main limitation of the proposed method is that the proposed procedure is a kind of man-machine interactive one in the sense that revision of the process parameters is initiated under the direction of the operator.

In [43], Majhi and Atherton have presented a set of general expressions derived from a single asymmetrical relay feedback test for on-line plant identification. The expressions remain valid for an odd symmetrical limit cycle test method. They have also presented exact analytical expressions for the parameters of open-loop stable and unstable first-order plus dead-time (FOPDT) and second-order plus time delay plant transfer functions and integrator plus time delay plant transfer function in terms of limit cycle measurements. In addition, Majhi and Atherton have proposed tuning methods based on standard forms of a PI-PD controller controlling time delay processes which are particularly effective for integrating unstable plants. They have shown simulation

examples to illustrate the value of the general identification method and the improved performance provided by the proposed controllers. The main limitations of the methods are that the identification needs to interrupt the closed loop for the estimation procedure and the method for the identification and tuning are valid only for small delay processes.

2.7 Other Adaptive Control Literature

To develop new adaptive PID control strategies, it is useful to review some of the other adaptive control.

In [3], Åström has presented a survey of adaptive control theory and its applications, and has discussed different approaches with particular emphasis on model reference adaptive systems (MRAS) and self-tuning regulators (STR). In addition, he has discussed stability and convergence analysis techniques for analysing adaptive systems, and has shown that adaptive control laws can also be obtained from stochastic control theory. Åström has also presented three different uses (auto-tuning, automatic construction of gain schedules, and adaptive regulators) of adaptive control, and issues of importance for applications covering parameterization, tuning, and tracking. He concluded that when designed with adequate practical precautions adaptive control may be used successfully in a variety of applications even though many theoretical problems remain to be solved.

In [4], Brandt has stated several adaptive control implementation issues, such as "boundary conditions" and "performance issues." Boundary conditions include those conditions where the controller cannot control the system at the desired value. Performance issues, cover long-time stability, steady-state offset, response time of the

controller, and related topics. Brandt has examined some of these issues and demonstrated their effect on the performance of actual systems, by a presentation of data collected from the application of a self-tuning controller in mixed air and cooling coil control loops. The experimental results show that "boundary conditions" cause algorithm instability, and there exists long-time stability difficulties in the basic self-tuning algorithm. In addition, the experimental results demonstrate that self-tuning control augmented with bracketing software to the self-tuning controller is a viable control approach in HVAC systems. The main limitation in the reference is that to guarantee long-time loop stability, the setpoint changes which generated new process information, were considered, but this situation might not be the only factor causing changes in system dynamics.

In [44], Peterka has presented a time-domain method of quadratic-optimum control synthesis for a system described by a linear finite-memory output predictor updated in real-time, as an internal representation of the uncertain process to be controlled. The synthesis covers both the servo problem and the regulation problem, including program control (with pre-programmed command signal) and feedforward from measurable external disturbances. Peterka concluded that unlike the standard Riccati equation, the method leads to algorithms (or explicit formulae in low-order cases) which are numerically robust and therefore suitable for real-time computation using microprocessors with reduced word-length. But, from the OCS (optimum control synthesis) algorithm, it appears extremely difficult to compute all parameters in real-time.

2.8 Objectives of the Thesis

From the literature review in sections 2.2 – 2.7, we note that several adaptive control techniques have been proposed for HVAC systems. However, as indicated in the review, except for PRAC [25], most methods are analytical techniques and have not been implemented on HVAC systems. Therefore, the design of simple online adaptive tuning strategies for PID controllers in HVAC systems remains a priority research area. To this end, we propose three new adaptive PID control techniques. One of them is developed by using the PID tuning rules for the first-order plus dead-time (FOPDT) based on the loop-shaping H_∞ optimal control proposed in [39]. The other two methods are based on the idea of tracking control augmented with adaptive characteristics of DAT systems.

The specific objectives of the thesis are:

1. To develop an FOPDT model of a DAT system using experimental data set. Furthermore, to simplify the existing RLS algorithm [14] for online identification of DAT system on a microcomputer based modular building controller.
2. To develop three adaptive PID control strategies for a DAT system. These are:
   (i) a Model Following Adaptive (MFA) PI control strategy,
   (ii) a Control-Model Based Adaptive (CMBA) PI control strategy,
   (iii) an H_∞ Based Adaptive PI (HIAPI) control strategy.

   The above control strategies are designed to have the following properties:
   - Ease of implementation on existing industrial controllers;
   - Fast regulation against static and dynamic disturbances;
   - Continuous adaptation to system operating conditions;
   - Fast tracking of desired discharge air temperature changes;

- Robust to changes in operating conditions, multiple disturbances and control-loop interactions.

3. To present H_∞ PI tuning rules for first-order plus dead-time (FOPDT) models for discrete-time systems based on the PID tuning rules for continuous-time systems proposed in [39].
4. To simulate the closed loop responses of the three adaptive control strategies and compare them with a constant PI, a LQR optimal adaptive PI control and the Ziegler-Nichols based adaptive PI control responses using Simulink.
5. To carry out experiments in an HVAC laboratory test facility to implement and test (i) the CMBA control strategy and (ii) the HIAPI control under realistic operations. Also, to compare the output responses with those from existing constant PI control.

In the following chapters, the research results, which satisfy the above-mentioned objectives, are presented.

3. Modeling and Online Identification of DAT System

3.1 Introduction

The Discharge Air Temperature (DAT) loop must be designed and operated to maintain dynamic equilibrium between the zone thermal loads and plant heat extraction rates. Because of its impact on occupants' comfort, equipment performance and operating costs, the DAT control problem is a challenging research problem with the potential for significant economic benefits.

Several authors have studied the modelling aspects of DAT systems. For example, the most detailed models based on the principle of energy and mass balances are reported in [15]. Other simplified models such as first-order models have been reported in [16]. These latter models have been used in the design of PID controllers for DAT systems. The controller design methods include both classical design methods [17] and more recent advanced design methods [18].

From a practical standpoint, the DAT controller should be

(i) simple for implementation using available hardware,

(ii) adaptive to load changes and

(iii) stable,

and give

(iv) near optimal temperature control.

In this chapter we propose to develop both second-order and first-order models of DAT systems. The major focus of this development will be that the models should be suitable for online implementation and control.

3.2 Physical Model

Figure 3.1 shows a schematic diagram of a DAT system. Mixed air enters the cooling coil at temperature $Ta0$ and flow rate FR. It is cooled and dehumidified in the cooling coil by using chilled water. The temperature of the air leaving the cooling coil Ta is controlled by modulating the chilled water flow rate in the coil as shown by the feedback control loop.

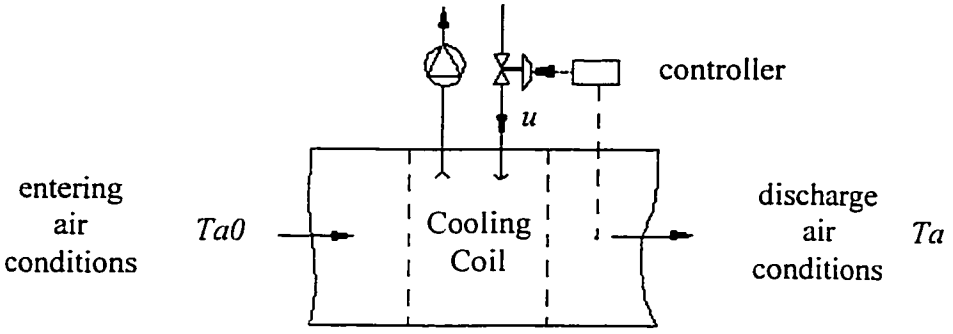

Figure 3.1 Schematic diagram of DAT system

If we assume humidity to remain constant, the input-output model can be represented by

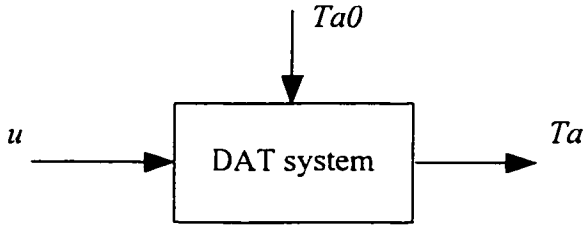

Figure 3.2 Block diagram of DAT system

where u, the chilled water flow rate, is the input and Ta, the DAT, is the output. The entering air temperature $Ta0$ is considered as disturbance on the system.

3.3 Second-Order Model

If we consider that the DAT system dynamics can be represented by a second-order system, then the model equation can be expressed as

$$T_a(k) = C_1 T_a(k-1) + C_2 T_a(k-2) + C_3 u(k) + C_4 u(k-1) + C_5 CWRTS \qquad (3.1)$$

where *CWTRS* is the chilled water return temperature. By using an available experimental data set [13] (Table 3.1) and employing the least-squares method [14] we minimized the error between the estimated temperature (T_a) and the measured temperature (T_{ae}):

$$\min_{C} \sum_{k=0}^{N} \left(T_{ae}(k) - T_a(k)\right)^2 \qquad (3.2)$$

to compute the system parameters $C = [C_1 \quad C_2 \quad C_3 \quad C_4 \quad C_5]^T$. With a sample size ($N$) of 150, and initial estimate $C0 = [0 \quad 0 \quad 0 \quad 0 \quad 0]^T$ the resulting parameters were

$$C = [1.318 \quad -0.3432 \quad -1.0715 \quad 0.2036 \quad 0.1133]^T.$$

We noted that different initial estimates converged to slightly different output values. For example, with another estimate of $C0 = [1.3 \quad -0.3 \quad -1.1 \quad 0.2 \quad 0.1]^T$ the new set of system parameters were

$$C = [1.7246 \quad -0.7333 \quad -1.125 \quad 0.1639 \quad 0.1037]^T.$$

Figures 3.3 and 3.4 show the open-loop responses for T_a of the second-order model and the experimental data T_{ae} at the following operating conditions (condition1): $T_{a0} = 63.77 \ ^\circ F$, $u = 4.5 \ gpm$ and $CWRTS = 46.04 \ ^\circ F$. Note that in Figure 3.3, the parameter set (set1)

$$C = [1.318 \quad -0.3432 \quad -1.0715 \quad 0.2036 \quad 0.1133]^T$$

was used, whereas Figure 3.4 corresponds to the second set of parameters (set2)

$$C = [1.7246 \quad -0.7333 \quad -1.125 \quad 0.1639 \quad 0.1037]^T .$$

t	0	4	8	12	16	20	24	28	32
Tae	17.65	17.58	17.05	16.6	16.1	16	15.63	15.4	15.1
t	36	40	44	48	52	56	60	64	68
Tae	14.9	14.65	14.5	14.4	14.37	14.3	14.2	14.1	14.07
t	72	76	80	84	88	92	96	100	104
Tae	14	13.95	13.9	13.8	13.68	13.6	13.55	13.4	13.3
t	108	112	116	120	124	128	132	136	140
Tae	13.2	13.15	13.1	13.02	12.98	12.9	12.8	12.68	12.65
t	144	148	152	156	160	164	168	172	176
Tae	12.57	12.5	12.45	12.4	12.35	12.32	12.22	12.21	12.2
t	180	184	188	192	196	200	204	208	212
Tae	12.18	12.15	12.1	12	11.98	11.95	11.93	11.9	11.85
t	216	220	224	228	232	236	240	244	248
Tae	11.83	11.81	11.79	11.75	11.68	11.67	11.65	11.61	11.6
t	252	256	260	264	268	272	276	280	284
Tae	11.595	11.59	11.56	11.5	11.48	11.46	11.44	11.42	11.41
t	288	292	296	300	304	308	312	316	320
Tae	11.4	11.4	11.4	11.4	11.398	11.39	11.38	11.37	11.37
t	324	328	332	336	340	344	348	352	356
Tae	11.365	11.365	11.364	11.364	11.363	11.363	11.362	11.362	13.32
t	360	364	368	372	376	380	384	388	392
Tae	11.3	11.29	11.28	11.28	11.28	11.27	11.27	11.27	11.26
t	396	400	404	408	412	416	420	424	428
Tae	11.26	11.25	11.24	11.24	11.23	11.23	11.229	11.226	11.223
t	432	436	440	444	448	452	456	460	464
Tae	11.22	11.218	11.216	11.214	11.212	11.208	11.206	11.204	11.202
t	468	472	...	600					
Tae	11.201	11.2	11.2	11.2					

Table 3.1 Experimental data at $u = 4.5$ *gpm*, $T_{a0} = 63.77$ and $CWRTS = 46.04\ ^\circ F$

In both cases there is a good match between the response of the model and the experimental data. This indicates that the second-order model is likely adequate to capture the dynamics of the DAT system.

To check the variability in operating conditions and their effect, the system parameters were identified using a new set of experimental data [13] (Table3.2).

t	0	4	8	12	16	20	24	28	32
Tae	18.9	18.9	18.8	18.63	18.55	18.44	18.4	18.28	18.24
t	36	40	44	48	52	56	60	64	68
6..2.1.1 Tae	18.2	18.18	18.14	18.1	18.05	18	17.95	17.9	17.8
t	72	76	80	84	88	92	96	100	104
Tae	17.6	17.5	17.35	17.05	16.7	16.5	16.3	16.1	15.8
t	108	112	116	120	124	128	132	136	140
Tae	15.55	15.2	14.9	14.7	14.58	14.38	14.08	13.98	13.79
t	144	148	152	156	160	164	168	172	176
Tae	13.6	13.43	13.26	13.05	12.93	12.8	12.7	12.6	12.4
t	180	184	188	192	196	200	204	208	212
Tae	12.22	12.18	12.1	12	11.91	11.8	11.73	11.7	11.65
t	216	220	224	228	232	236	240	244	248
Tae	11.6	11.45	11.3	11.25	11.2	11.15	11.1	11.06	11.03
t	252	256	260	264	268	272	276	280	284
Tae	11	10.9	10.8	10.78	10.77	10.76	10.74	10.72	10.7
t	288	292	296	300	304	308	312	316	320
6..2.1.2 Tae	10.65	10.6	10.55	10.54	10.53	10.5	10.4	10.37	10.3
t	324	328	332	336	340	344	348	352	356
Tae	10.25	10.24	10.23	10.22	10.215	10.21	10.205	10.2	10.19
t	360	364	368	372	376	380	384	388	392
Tae	10.18	10.17	10.15	10.13	10.1	10.09	10.08	10.07	10.06
t	396	400	404	408	412	416	420	424	428
Tae	10.05	10.04	10.03	10.025	10.02	10.015	10.01	10.005	10
t	432	...	600						
Tae	10	10	10						

Table 3.2 Experimental data at $u = 5.4$ *gpm*, $T_{a0} = 66.02$ and $CWRTS = 44.06\ ^\circ F$

The new set of experimental data was obtained at a new operating condition (condition2): $T_{a0} = 66.02\ ^\circ F$, $u = 5.4$ *gpm* and $CWRTS = 44.06\ ^\circ F$. Figure 3.5 and Figure 3.6 show the open-loop responses T_a of the DAT system corresponding to the new operating conditions using two sets of parameters separately. The set of parameters (set3) for Figure 3.5

$$C = [1.91 \quad -0.9128 \quad -2.0838 \quad -0.031 \quad 0.2623]^T$$

and the set of parameters (set4) for Figure 3.6

$$C = [1.9128 \quad -0.9155 \quad -2.8504 \quad -0.0283 \quad 0.3559]^T$$

were used. Again the second-order model is able to yield the output temperature responses that are very close to the measured temperatures.

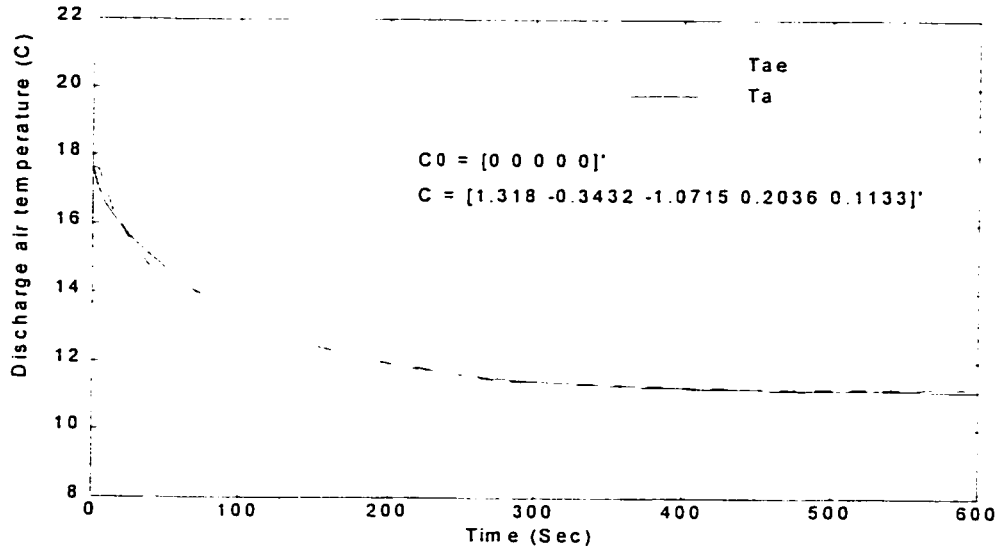

Figure 3.3 **Open-loop responses for the second-order model using set1 in condition1**

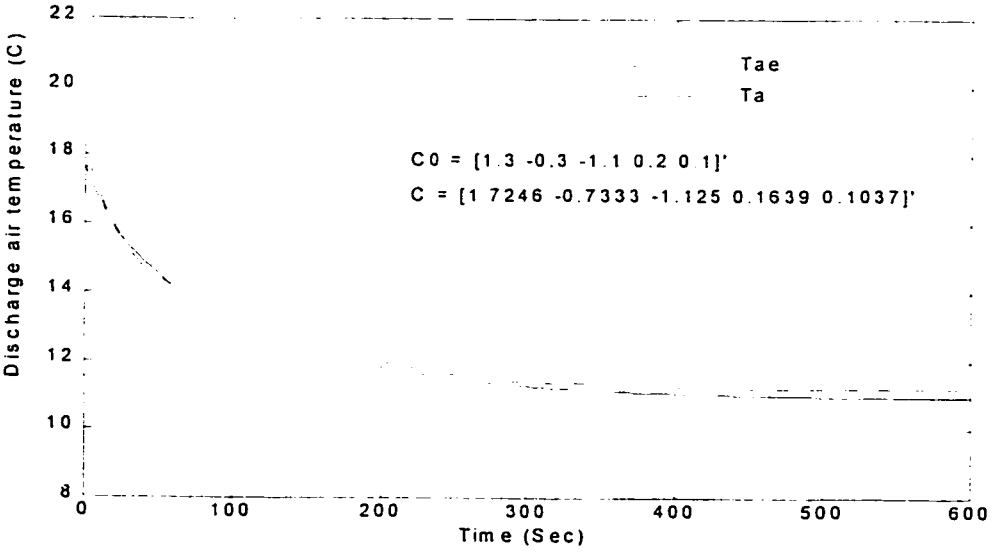

Figure 3.4 **Open-loop responses for the second-order model using set2 in condition1**

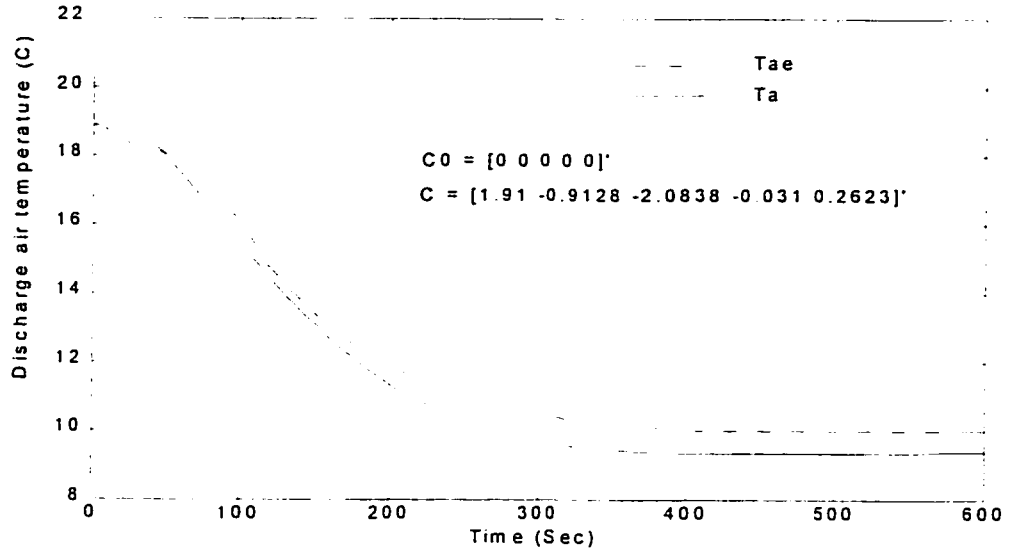

Figure 3.5 **Open-loop responses for the second-order model using set3 in condition2**

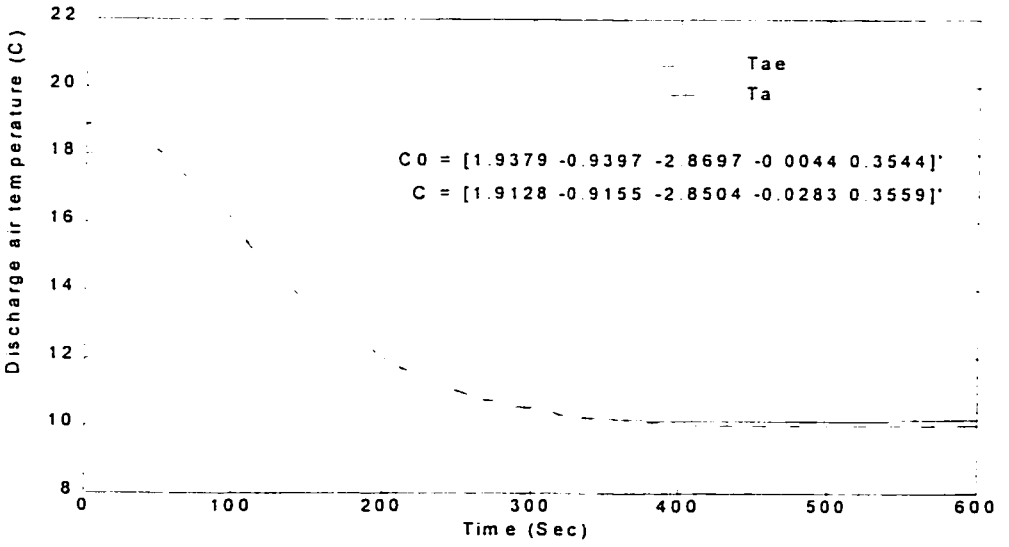

Figure 3.6 **Open-loop responses for the second-order model using set4 in condition2**

From the open-loop responses, it can be noted that there is some difficulty in determining the true least-squares result for the second-order system because different initial conditions leads to different final values. However, these differences are not very significant.

3.4 First-Order Model

From the point of view of implementation, it is necessary to reduce the computational effort required in the identification and control of the DAT system. To this end, we model the DAT system as a first-order plus dead-time (FOPDT) system. The model can be represented as follows.

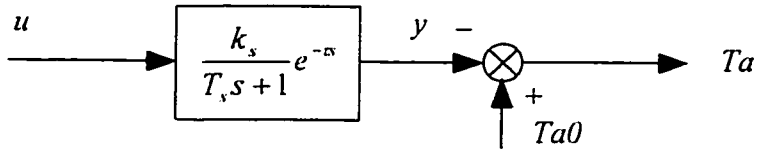

Figure 3.7 First-order plus dead-time model

Where $Ta0$ is the output of the system when $u = 0$, Ta is the output of the system and u is the input of the system. Assuming T is the sampling time in the discrete system and

$$l = \frac{\tau}{T} \qquad (3.3)$$

is an integer, then the discrete model can be expressed as

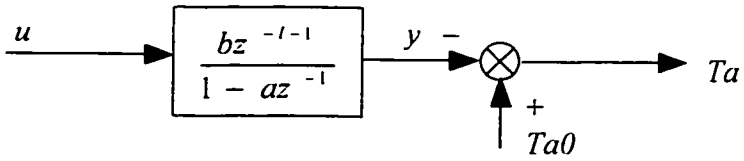

Figure 3.8 First-order plus dead-time model in discrete system

where, $a = e^{-\frac{T}{T_s}}$ (3.4)

and

$$b = k_s(1-a). \qquad (3.5)$$

By using the experimental data given in Table 3.1, and minimizing the function

$$\min_{C} \sum_{k=0}^{N} \left(T_{ae}(k) - T_a(k)\right)^2 \qquad (3.6)$$

the system parameters $C = [0.9547 \quad 0.1164]^T$ (set5) were obtained, where $C = [a \quad b]^T$, $T_a(k) = T_{a0} - y(k)$ and $N = 150$.

Figure 3.9 shows the open-loop response T_a of the FOPDT model using the system parameters (set5), and the experimental data T_{ae} of the DAT at the operating condition corresponding to $T_{a0} = 63.77\ ^\circ F$, $u = 4.5\ gpm$ and $CWRTS = 46.04\ ^\circ F$.

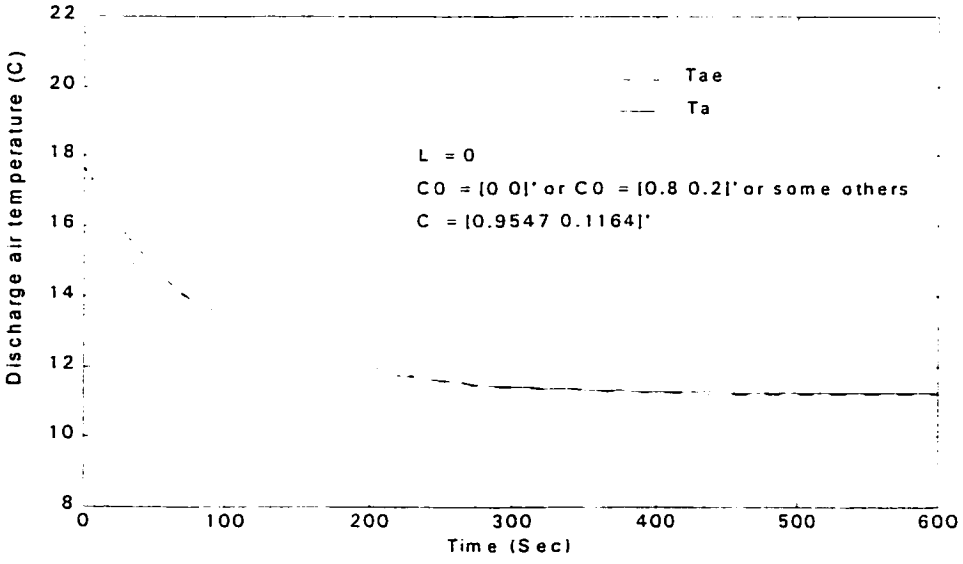

Figure 3.9 Open-loop responses for FOPDT model using set5 in condition1

Similarly Figure 3.10 shows the open-loop response T_a of the FOPDT model using the system parameters $C = [0.9558 \quad 0.1326]^T$ (set6), corresponding to the second

experimental data set with the operating point given by $T_{a0} = 66.02\ ^{\circ}F$, $u = 5.4\ gpm$ and $CWRTS = 44.06\ ^{\circ}F$.

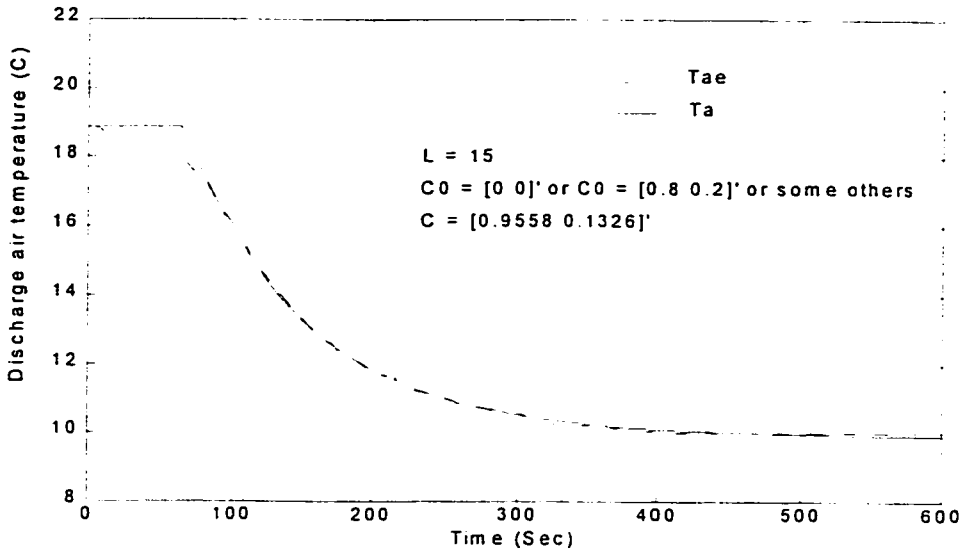

Figure 3.10 Open-loop responses for FOPDT model using set6 in condition2

Unlike the second-order model, from the open-loop responses of FOPDT model, we note that the least square results are much closer. Because of simplicity, we consider the first-order model for the control design in this thesis.

3.5 Online Identification of the DAT System

The recursive least squares (RLS) method is an effective approach in online identification because a new estimate can be obtained easily. From a physical understanding of the dynamics of the system, it is possible to reduce the memory requirements of RLS method in updating the parameters of the first-order plus dead-time model. Also, the RLS equations will be simplified for direct implementation on an existing Energy Management Control System (EMCS) [13].

3.5.1 RLS Method for the FOPDT Model

In order to implement the RLS algorithm for online identification of the DAT system, we consider the follow diagram as in [14].

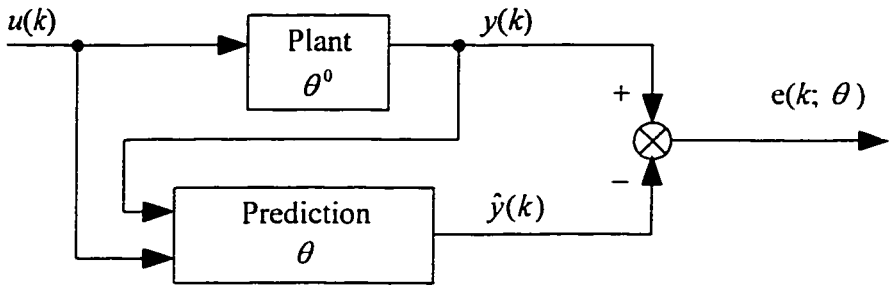

Figure 3.11 Block diagram showing the generation of output prediction error

Considering the system of Fig.3.11 as a model, we note that, if l is fixed, the system identification reduces to the estimation of parameters a and b of the plant

$$G(z) = \frac{Y(z)}{U(z)} = \frac{bz^{-l-1}}{1-az^{-1}}. \qquad (3.7)$$

Using measured values for $Ta0$, $Ta(k)$ and $u(k)$ for some k, we have

$$y(k) = Ta0 - Ta(k) \qquad (3.8)$$

and $$y(k) - ay(k-1) - bu(k-l-1) = e(k;\theta). \qquad (3.9)$$

Let $$\theta = [a \quad b]^T \qquad (3.10)$$

and $$\psi(k) = [y(k-1) \quad u(k-l-1)]^T \qquad (3.11)$$

then, through analysis, we can write the error equation as

$$Y(N) = \Psi(N)\theta + \varepsilon(N;\theta) \qquad (3.12)$$

where $\Psi(N) = [\psi(l+1) \quad \psi(l+2) \quad \cdots \quad \psi(N)]^T$ (3.13)

$$\varepsilon(N;\theta) = [e(l+1;\theta) \quad \cdots \quad e(N;\theta)]^T \qquad (3.14)$$

$$Y(N) = [y(l+1) \quad \cdots \quad y(N)]^T . \qquad (3.15)$$

Assume that we observe the set of inputs and outputs

$$\{u(0), u(1), \cdots, u(N), y(0), y(1), \cdots, y(N)\},$$

and that we need to estimate the parameter θ^0 of the plant and the prediction is θ. The least-squares method can be expressed as

$$\min_{\theta} J(\theta) \qquad (3.16)$$

where $J(\theta) = \sum_{k=n}^{N} w(k) e^2(k;\theta) = \varepsilon^T W \varepsilon$ (3.17)

and the weighting function $w(k)$ is positive.

If we now substitute the expression for $\varepsilon(N;\theta)$ from (3.12) into (3.17), we have

$$\begin{aligned} J(\theta) &= (Y(N) - \Psi(N)\theta)^T W(N)(Y(N) - \Psi(N)\theta) \\ &= Y^T(N)W(N)Y(N) - \theta^T \Psi^T(N)W(N)Y(N) \\ &\quad - Y^T(N)W(N)\Psi(N)\theta + \theta^T \Psi^T(N)W(N)\Psi(N)\theta \end{aligned} \qquad (3.18)$$

which is a quadratic function of the two parameters in θ. To find the minimum $\hat{\theta}$ of $J(\theta)$, we consider that

$$\frac{\partial J}{\partial \theta} = -2\Psi^T(N)W(N)Y(N) + 2\theta^T \Psi^T(N)W(N)\Psi(N) \qquad (3.19)$$

and $$\frac{\partial^2 J}{\partial \theta^2} = 2\Psi^T(N)W(N)\Psi(N) \qquad (3.20)$$

is positive definite (because $w(k)$ is positive for any k,) then

$$\hat{\theta} = \left(\Psi^T(N)W(N)\Psi(N)\right)^{-1}\Psi^T(N)W(N)Y(N) \qquad (3.21)$$

(when $\Psi^T(N)W(N)\Psi(N)$ is non-singular,) and by the first-order necessary conditions for a minimum.

When we choose $W = I$, the identity matrix, the weighted least squares cost $J(\theta)$ reduces to ordinary least squares. Another common choice for $w(k)$ is

$w(k) = (1-\gamma)\gamma^{N-k}$ [14].

From equation (3.13) we have, for data up to time $N+1$,

$$\Psi(N+1) = \left[\psi(l+1) \quad \psi(l+2) \quad \cdots \quad \psi(N) \quad \psi(N+1)\right]^T. \qquad (3.22)$$

Assume $w(k) = \alpha\gamma^{N-k}$, then

$$\Psi^T(N+1)W(N+1)\Psi(N+1) = \sum_{k=n}^{N+1}\psi(k)w(k)\psi^T(k) = \sum_{k=n}^{N+1}\psi(k)\alpha\gamma^{N+1-k}\psi^T(k) \qquad (3.23)$$

which can be written in two terms as

$$\Psi^T(N+1)W(N+1)\Psi(N+1) = \sum_{k=n}^{N}\psi(k)\alpha\gamma\gamma^{N-k}\psi^T(k) + \psi(N+1)\alpha\psi^T(N+1)$$

$$= \gamma\Psi^T(N)W(N)\Psi(N) + \psi(N+1)\alpha\psi^T(N+1). \qquad (3.24)$$

It can also be written as

$$P(N+1) = \left[\gamma P^{-1}(N) + \psi(N+1)\alpha\psi^T(N+1)\right]^{-1} \qquad (3.25)$$

when we define the 2×2 matrix P as

$$P(N+1)=\left[\Psi^T(N+1)W(N+1)\Psi(N+1)\right]^{-1}. \qquad (3.26)$$

Using the matrix inversion lemma

$$(A+BCD)^{-1}=A^{-1}-A^{-1}B(C^{-1}+DA^{-1}B)^{-1}DA^{-1} \qquad (3.27)$$

and considering

$$A=\gamma P^{-1}(N),\ \ B=\psi(N+1),\ C=\alpha,\ D=\psi^T(N+1) \qquad (3.28)$$

we have

$$P(N+1)=\frac{P(N)}{\gamma}-\frac{P(N)}{\gamma}\psi(N+1)\left(\frac{1}{\alpha}+\psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\right)^{-1}\psi^T(N+1)\frac{P(N)}{\gamma} \qquad (3.29)$$

and $\frac{1}{\alpha}+\psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)$ is a scalar, so in the above equation, no matrix inversion is required.

To obtain the final recursive least-squares equations, we also need to consider $\Psi^T(N+1)W(N+1)\Psi(N+1)$ which can be written as

$$\Psi^T(N+1)W(N+1)\Psi(N+1)=$$

$$\left[\psi(l+1)\ \ \cdots\ \ \psi(N)\ \ \psi(N+1)\right]\begin{bmatrix}\alpha\gamma^{N-l} & & & O\\ & \ddots & & \\ & & \alpha\gamma & \\ O & & & \alpha\end{bmatrix}\begin{bmatrix}y(l+1)\\ \vdots\\ y(N)\\ y(N+1)\end{bmatrix}$$

$$=\gamma\Psi^T(N)W(N)Y(N)+\psi(N+1)\alpha y(N+1). \qquad (3.30)$$

Therefore,

$$\hat{\theta}(N+1) = \left(\Psi^T(N+1)W(N+1)\Psi(N+1)\right)^{-1}\Psi^T(N+1)W(N+1)Y(N+1)$$

$$= \left[\frac{P(N)}{\gamma} - \frac{P(N)}{\gamma}\psi(N+1)\left(\frac{1}{\alpha} + \psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\right)^{-1}\psi^T(N+1)\frac{P(N)}{\gamma}\right]\times$$

$$[\gamma\Psi^T(N)W(N)Y(N) + \psi(N+1)\alpha y(N+1)] \quad (3.31)$$

with $\hat{\theta}(N) = P(N)\Psi(N)W(N)Y(N)$, then

$$\hat{\theta}(N+1) = \hat{\theta}(N) + \frac{P(N)}{\gamma}\psi(N+1)\alpha y(N+1)$$

$$-\frac{P(N)}{\gamma}\psi(N+1)\left(\frac{1}{\alpha} + \psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\right)^{-1}\psi^T(N+1)\hat{\theta}(N)$$

$$-\frac{P(N)}{\gamma}\psi(N+1)\left(\frac{1}{\alpha} + \psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\right)^{-1}\psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\alpha y(N+1).$$

(3.32)

Inserting the expression

$$\left(\frac{1}{\alpha} + \psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\right)^{-1}\left(\frac{1}{\alpha} + \psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\right)$$

between the $\psi(N+1)$ and the α in the second term on the right of the above equation, we can obtain the recursive equation

$$\hat{\theta}(N+1) = \hat{\theta}(N) + L(N+1)\left(y(N+1) - \psi^T(N+1)\hat{\theta}(N)\right) \quad (3.33)$$

where

$$L(N+1) = \frac{P(N)}{\gamma}\psi(N+1)\left(\frac{1}{\alpha} + \psi^T(N+1)\frac{P(N)}{\gamma}\psi(N+1)\right)^{-1}. \quad (3.34)$$

If we now consider the expression for $L(N+1)$ from the above equation, we have the expression for $P(N+1)$ from (3.29).

$$P(N+1)=\left(I-L(N+1)\psi^T(N+1)\right)\frac{P(N)}{\gamma} \qquad (3.35)$$

The RLS algorithm for FOPDT model can be summarized as follows [14]:

Step 1: Select $N\,(>2\times l)$

Step 2: Select α and γ

Comment: $\alpha=\gamma=1$ is ordinary least squares; $\alpha=1-\gamma$ and $0<\gamma<1$ is exponentially weighted least squares

Step 3: Select initial values for $P(N)$ and $\hat{\theta}(N)$. One possibility is to set $\hat{\theta}(N)=\begin{bmatrix}0\\0\end{bmatrix}$, $P(N)=\begin{bmatrix}\beta & 0\\0 & \beta\end{bmatrix}$, where β is a large scalar. This choice has the property that we do not need much memory and matrix inverse, and the computations can be reduced in order to calculate $P(N)$ and $\hat{\theta}(N)$

Step 4: Collect $y(N)$ and $u(N-l)$ and form $\psi^T(N+1)$

Step 5: Let $k \leftarrow N$

Step 6: $L(k+1)\leftarrow\frac{P(k)}{\gamma}\psi(k+1)\left(\frac{1}{\alpha}+\psi^T(k+1)\frac{P(k)}{\gamma}\psi(k+1)\right)^{-1}$

Step 7: Collect $y(k+1)$ and $u(k-l+1)$

Step 8: $\hat{\theta}(k+1)\leftarrow\hat{\theta}(k)+L(k+1)\left(y(k+1)-\psi^T(k+1)\hat{\theta}(k)\right)$

Step 9: $P(k+1)\leftarrow\frac{1}{\gamma}\left[I-L(k+1)\psi^T(k+1)\right]P(k)$

Step 10: Form $\psi(k+2)$

Step 11: $k \leftarrow k+1$

Step 12: Go to step 6.

3.5.2 Implementation

The RLS equations can be put in a more explicit form for the FOPDT model as shown below.

Assume that

$P(k)=\begin{bmatrix} P_{11}(k) & P_{12}(k) \\ P_{21}(k) & P_{22}(k) \end{bmatrix}$, $\hat{\theta}(k)=\begin{bmatrix} \hat{a}(k) \\ \hat{b}(k) \end{bmatrix}$ and $L(k)=\begin{bmatrix} L_1(k) \\ L_2(k) \end{bmatrix}$, then considering (3.11) we have

$$\hat{a}(N+1)=\hat{a}(N)+L_1(N+1)\sigma(N+1) \tag{3.36}$$

$$\hat{b}(N+1)=\hat{b}(N)+L_2(N+1)\sigma(N+1) \tag{3.37}$$

from (3.33), where

$$\sigma(N+1)=y(N+1)-\left(\hat{a}(N)y(N)+\hat{b}(N)u(N-l)\right) \tag{3.38}$$

and

$$L_1(N+1)=\frac{\mu(N+1)}{\gamma}\left(P_{11}(N)y(N)+P_{12}(N)u(N-l)\right) \tag{3.39}$$

$$L_2(N+1)=\frac{\mu(N+1)}{\gamma}\left(P_{21}(N)y(N)+P_{22}(N)u(N-l)\right) \tag{3.40}$$

from (3.34), where

$$\mu(N+1)=\frac{1}{\frac{1}{\alpha}+P_{11}(N)y^2(N)+\left(P_{12}(N)+P_{21}(N)\right)y(N)u(N-l)+P_{22}(N)u^2(N-l)} \tag{3.41}$$

and $$P_{11}(N+1)=\frac{1}{\gamma}\left[P_{11}(N)\left(1-L_1(N+1)y(N)\right)-P_{21}(N)L_1(N+1)u(N-l)\right] \quad (3.42)$$

$$P_{12}(N+1)=\frac{1}{\gamma}\left[P_{12}(N)\left(1-L_1(N+1)y(N)\right)-P_{22}(N)L_1(N+1)u(N-l)\right] \quad (3.43)$$

$$P_{21}(N+1)=\frac{1}{\gamma}\left[-PLY_1(N+1)+P_{21}(N)\left(1-L_2(N+1)u(N-l)\right)\right] \quad (3.44)$$

$$P_{22}(N+1)=\frac{1}{\gamma}\left[-PLY_2(N+1)+P_{22}(N)\left(1-L_2(N+1)u(N-l)\right)\right] \quad (3.45)$$

from (3.35), where

$$PLY_1(N+1)=P_{11}(N)L_2(N+1)y(N) \quad (3.46)$$

$$PLY_2(N+1)=P_{12}(N)L_2(N+1)y(N). \quad (3.47)$$

Equations (3.36) to (3.47) can be directly implemented on the existing EMCS [13] because they are expressed in scalar form.

Considering (3.39) to (3.41), step 6 in the RLS algorithm can be expressed as three sub-steps such as

$$\text{step6-1: } \mu(k+1)\leftarrow\frac{1}{\frac{1}{\alpha}+P_{11}(k)y^2(k)+\left(P_{12}(k)+P_{21}(k)\right)y(k)u(k-l)+P_{22}(k)u^2(k-l)}$$

$$\text{step6-2: } L_1(k+1)\leftarrow\frac{\mu(k+1)}{\gamma}\left(P_{11}(k)y(k)+P_{12}(k)u(k-l)\right)$$

and

$$\text{step6-3: } L_2(k+1)\leftarrow\frac{\mu(k+1)}{\gamma}\left(P_{21}(k)y(k)+P_{22}(k)u(k-l)\right).$$

These steps show that we need to add a variable to store μ and two variables to store L in order to reduce the computing time in practical systems. Now considering (3.36) to (3.38), step 8 of the RLS algorithm can be rewritten as another three sub-steps:

step8-1: $\sigma(k+1) \leftarrow y(k+1) - \left(\hat{a}(k)y(k) + \hat{b}(k)u(k-l)\right)$

step8-2: $\hat{a}(k+1) \leftarrow \hat{a}(k) + L_1(k+1)\sigma(k+1)$

and

step8-3: $\hat{b}(k+1) \leftarrow \hat{b}(k) + L_2(k+1)\sigma(k+1)$.

These steps also show that we need to add a variable to store σ and two more variables to store θ. Lastly considering (3.42) to (3.47), step 9 can be simplified as a sequence of six sub-steps:

step9-1: $PLY_1(k+1) \leftarrow P_{11}(k)L_2(k+1)y(k)$

step9-2: $PLY_2(k+1) \leftarrow P_{12}(k)L_2(k+1)y(k)$

step9-3: $P_{11}(k+1) \leftarrow \frac{1}{\gamma}\left[P_{11}(k)\left(1 - L_1(k+1)y(k)\right) - P_{21}(k)L_1(k+1)u(k-l)\right]$

step9-4: $P_{12}(k+1) \leftarrow \frac{1}{\gamma}\left[P_{12}(k)\left(1 - L_1(k+1)y(k)\right) - P_{22}(k)L_1(k+1)u(k-l)\right]$

step9-5: $P_{21}(k+1) \leftarrow \frac{1}{\gamma}\left[-PLY_1(k+1) + P_{21}(k)\left(1 - L_2(k+1)u(k-l)\right)\right]$

and

step9-6: $P_{22}(k+1) = \frac{1}{\gamma}\left[-PLY_2(k+1) + P_{22}(k)\left(1 - L_2(k+1)u(k-l)\right)\right]$.

These steps show that we need to add two variables to store PLY_1 and PLY_2 and four variables to store P. Altogether, we need four variables for the 2×2 matrix P, two variables for the values of PLY_1 and PLY_2, two variables for the 2×1 vector L, one

variable for μ, two variables for vector θ and one variable for σ as memory to compute the next estimate of $\hat{\theta}$. Also, we need two variables to store y and its previous value, and $l+1$ variables to store u and its medium values.

The RLS equations were first implemented using Simulink as depicted in Figure 3.12.

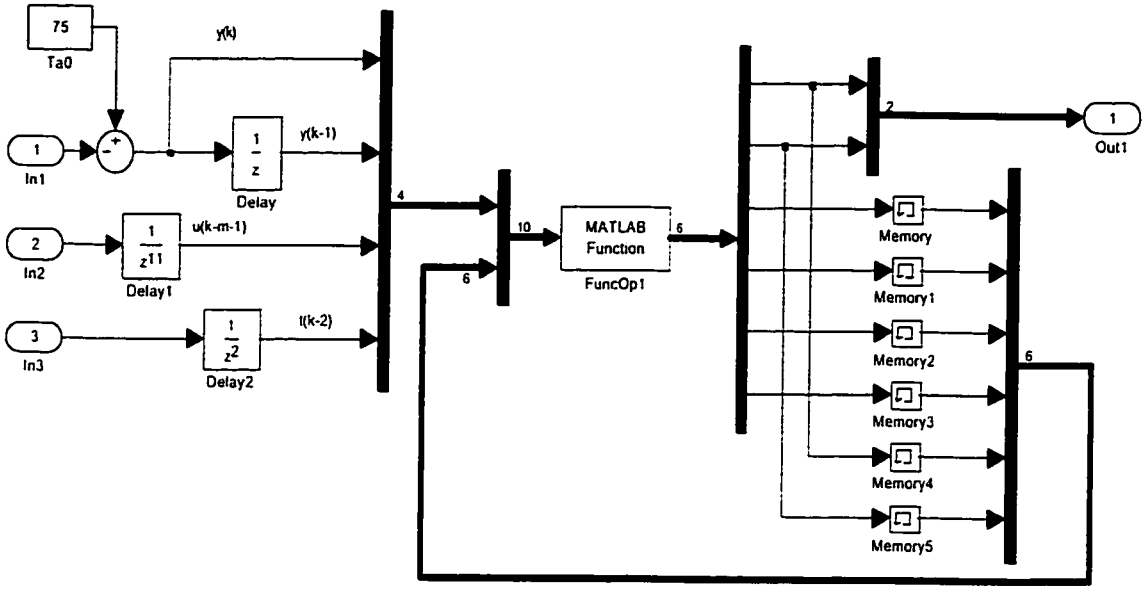

Figure 3.12 Implementation of RLS identification in Simulink (identifier – A)

FuncOp1 is a Matlab function that computes a new set of $\hat{\theta}$ and P. In1 is the input (the output of the DAT system). In2 is the input (the input of the DAT system). In3 is the time, and can also be a flag to indicate when the algorithm is working. Out1 is the output that returns the estimated $\hat{\theta}$.

Figure 3.13 shows the result of the implementation. In this simulation the DAT model and the operating conditions were defined as follows:

$$G(z) = \frac{Y(z)}{U(z)} = \frac{0.7388z^{-11}}{1-0.7688z^{-1}},\ Ta0 = 75\ ^{\circ}F$$

and $N = 25$, $\alpha = 0.8$, $\hat{\theta}(N) = [a \quad b]^T = [0 \quad 0]^T$, and $P(N) = \begin{bmatrix} \beta & 0 \\ 0 & \beta \end{bmatrix}$ where $\beta = 80000$.

The online identification was done in a closed loop. Table 3.3 shows the magnitudes of the estimated parameters a and b.

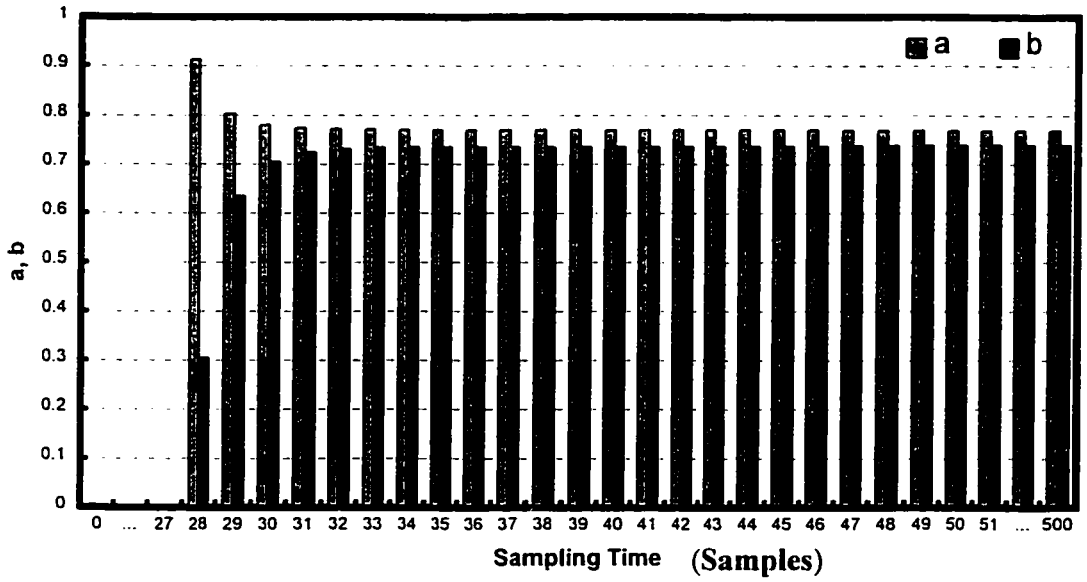

Figure 3.13 Online responses of a and b

l	$0-27$	28	29	30	31	32	33	34	35
a	0	0.9136	0.8035	0.7800	0.7737	0.7713	0.7706	0.7703	0.7703
b	0	0.3049	0.6345	0.7050	0.7241	0.7312	0.7334	0.7341	0.7343
l	36	37	38	39	40	41	42	43	44
a	0.7703	0.7703	0.7702	0.7701	0.7700	0.7699	0.7699	0.7698	0.7698
b	0.7343	0.7343	0.7345	0.7348	0.7351	0.7353	0.7355	0.7356	0.7356
l	45	46	47	48	49	50	51	–	500
a	0.7698	0.7697	0.7695	0.7692	0.7690	0.7689	0.7688	0.7688	0.7688
b	0.7357	0.7360	0.7368	0.7376	0.7381	0.7384	0.7388	0.7388	0.7388

Table 3.3 An example of the identification results

From Table 3.3, we see that after a few samples the estimated values of a, b converge rapidly and reach steady state.

3.6 Simulation of the FOPDT System

3.6.1 Open-loop Responses

Figure 3.14 shows the open-loop responses at several inputs, $u = 2$ *gpm*, $u = 4$ *gpm* and $u = 6$ *gpm* with system parameters $a = 0.9558$ and $b = 0.1326$. As shown in the figure, as the water flow rate (u) is increased the DAT decreases which is what one would expect in a practical system. From these responses, we conclude that the FOPDT model is responding properly to the changes in input.

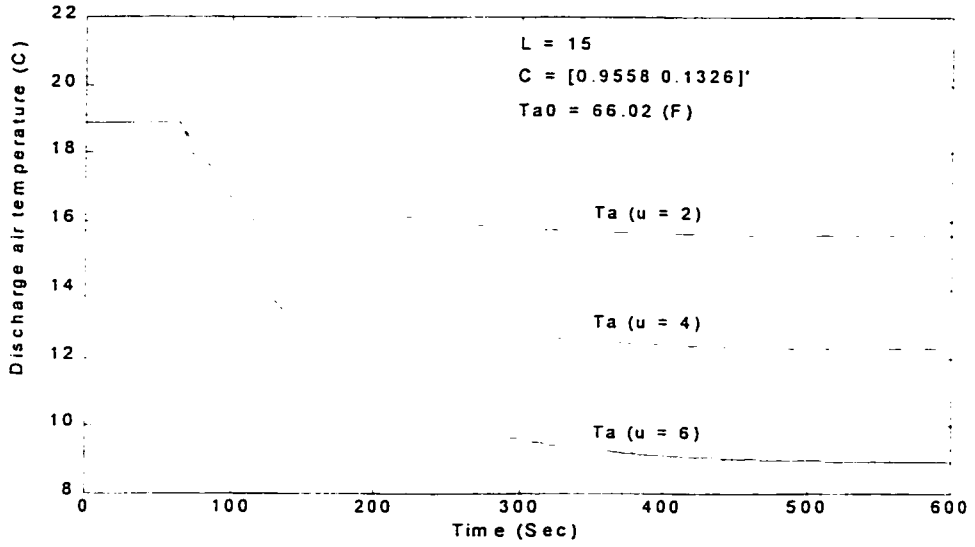

Figure 3.14 Open-loop responses for different inputs

Figure 3.15 shows the open-loop responses at $u = 3$ *gpm* for different values of a and b. It points to the fact that the response of the model is sensitive to parameter a when its magnitude is close to 1. The other parameter b has a greater operating range.

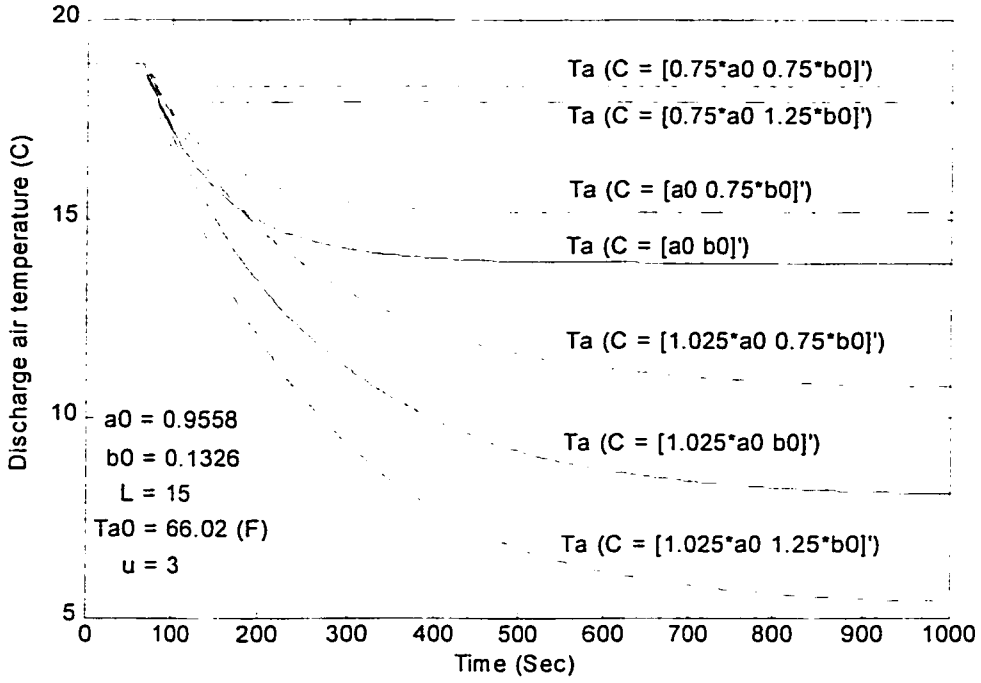

Figure 3.15 Open-loop responses for different parameters of the model

3.6.2 Online Identification and Simulation

The plant model with the following parameters was simulated.

$$G(z) = \frac{Y(z)}{U(z)} = \frac{bz^{-16}}{1 - az^{-1}},\ Ta0 = 66.02\ ^{\circ}F,\ \text{and sampling time } T = 4\ \text{sec},$$

where $a = \begin{cases} 0.9558 & t < 1000\,\text{sec} \\ 0.9797 & t \geq 1000\,\text{sec} \end{cases}$ and $b = \begin{cases} 0.1326 & t < 1000\,\text{sec} \\ 0.1658 & t \geq 1000\,\text{sec} \end{cases}$.

First we consider the online identification in the open loop. The implementation in Simulink is shown in Figure 3.16. The results of the above implementation scheme are shown in Figure 3.17. We can see that in the early identification process, the new set of parameters may be far from the final converged values. To avoid this, we may need two samples of the vector θ and two estimated samples of $\hat{\theta}$ as shown below.

When $|\theta(k)-\theta(k-1)| \le \delta$ (error vector), then $\hat{\theta}(k)=\theta(k)$; otherwise, $\hat{\theta}(k)=\hat{\theta}(k-1)$.

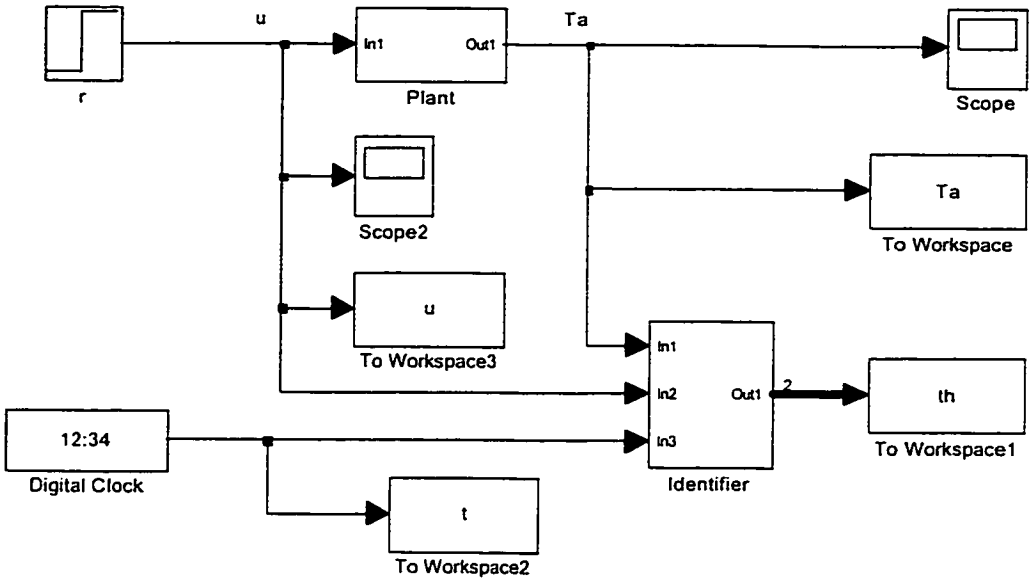

Figure 3.16 Online identification in open loop

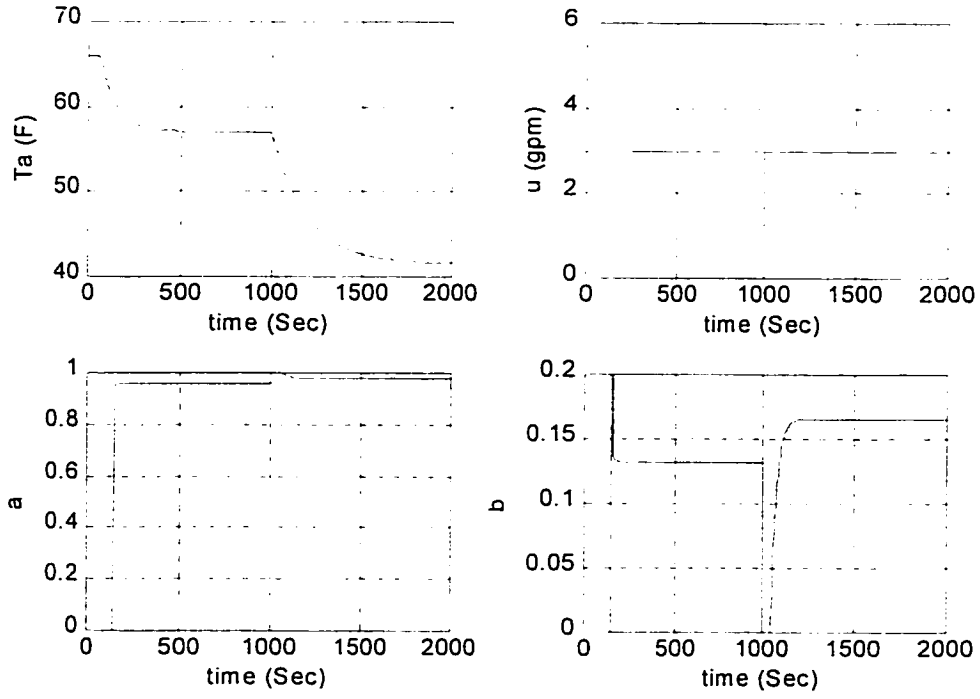

Figure 3.17 Identification result by the identifier – A

The implementation scheme for this strategy is shown below.

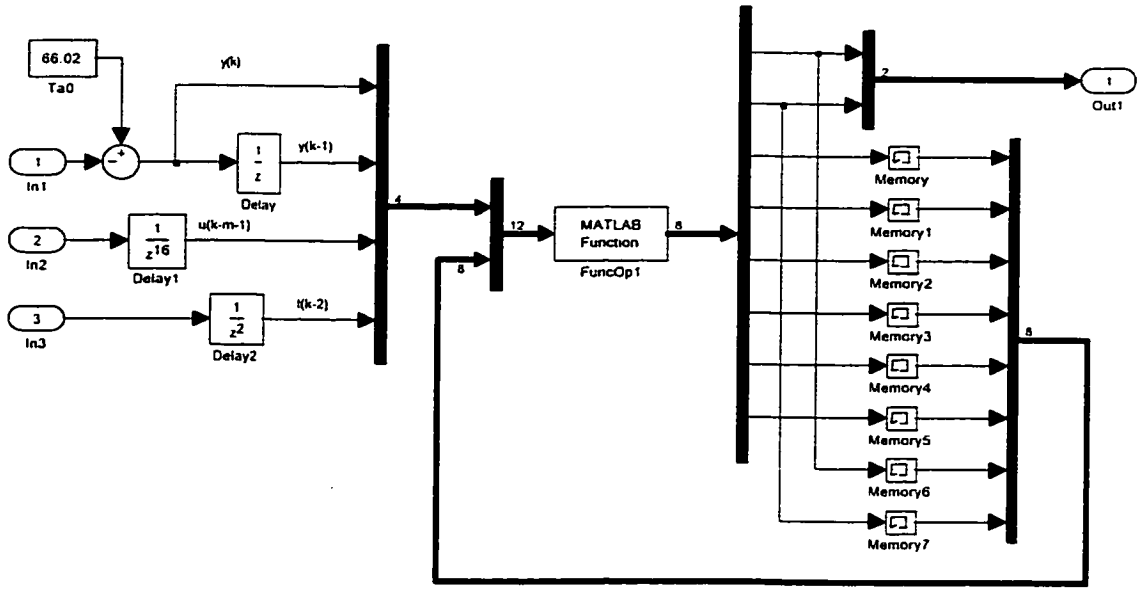

Figure 3.18 Advanced implementation scheme (identifier – B)

The results of the simulation with $\delta = [0.01 \quad 0.001]^T$ are shown in Figure 3.19. We see that the identification process converges fast and is smoother.

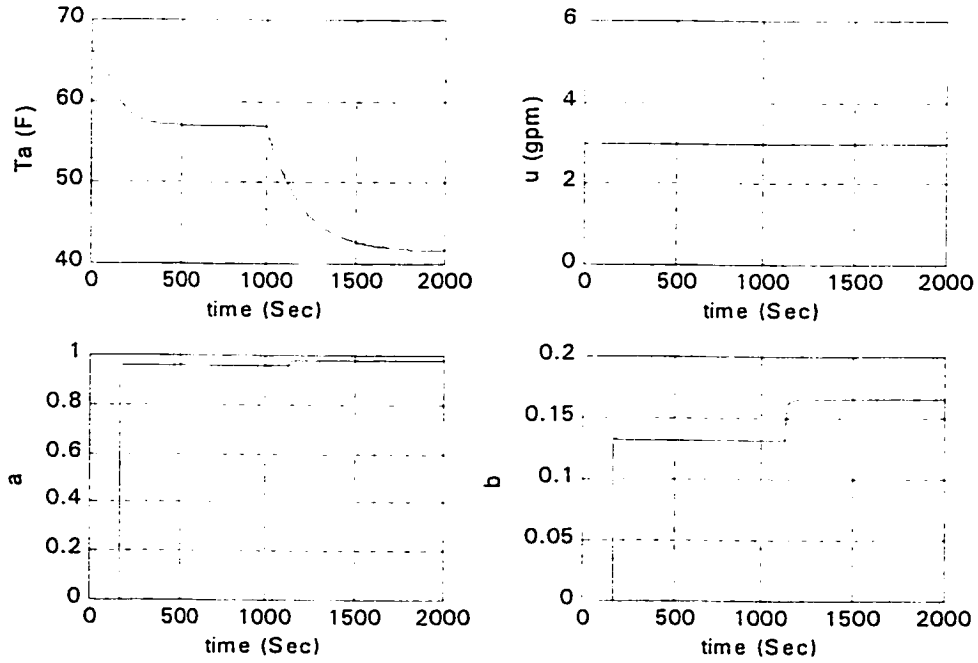

Figure 3.19 Identification result by the identifier – B

Now we consider the online identification in the closed loop mode. The implementation in Simulink is shown below.

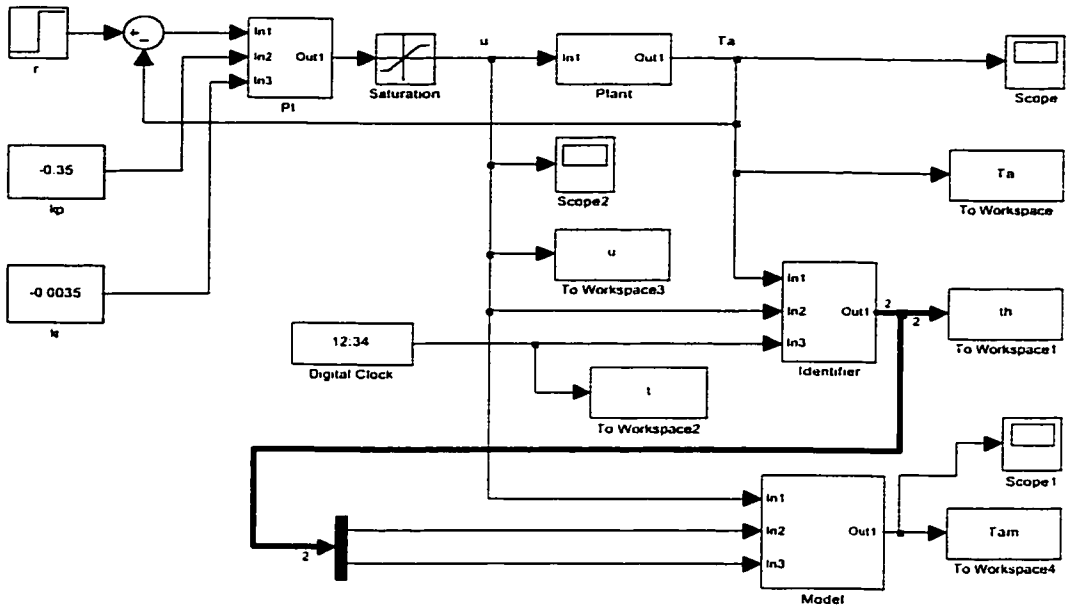

Figure 3.20 Online identification in closed loop mode

The model response (Tam) is not the same as the plant response (Ta) because the identification has time delay. We can see the delay in Figure 3.21. Table 3.4 shows the magnitudes of the parameters a and b.

l	0 – 42	43	44	45	46	47–279	280	281	282
a	0	0.9556	0.9557	0.9557	0.9558	0.9558	0.9804	0.9802	0.9801
b	0	0.1330	0.1328	0.1327	0.1327	0.1326	0.1613	0.1621	0.1627
l	283	284	285	286	287	288	289	290	291
a	0.9800	0.9800	0.9799	0.9799	0.9798	0.9798	0.9798	0.9798	0.9798
b	0.1632	0.1636	0.1640	0.1643	0.1645	0.1647	0.1649	0.1650	0.1651
l	292	293	294	295	296	297	298–301	302–312	313–500
a	0.9798	0.9797	0.9797	0.9797	0.9797	0.9797	0.9797	0.9797	0.9797
b	0.1652	0.1653	0.1654	0.1654	0.1655	0.1655	0.1656	0.1657	0.1658

Table 3.4 Identification of parameters a and b in closed loop mode

Figure 3.22 shows the results with a 30 ($2 \times l$) samples delay between the input of the plant (u) and the input (In1) of the model.

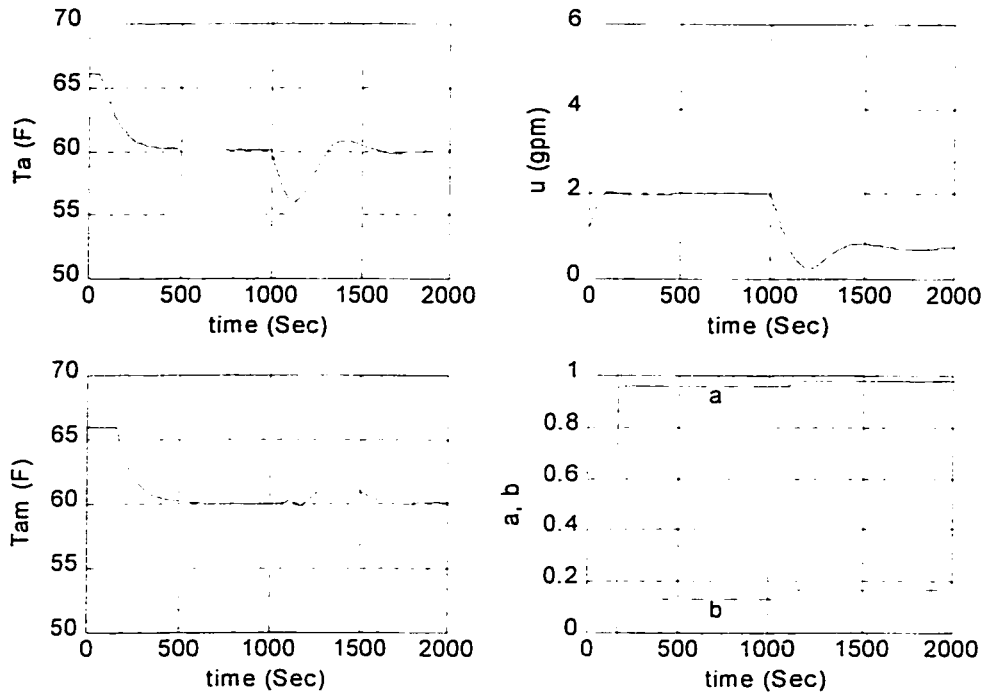

Figure 3.21 **Closed loop mode identification results**

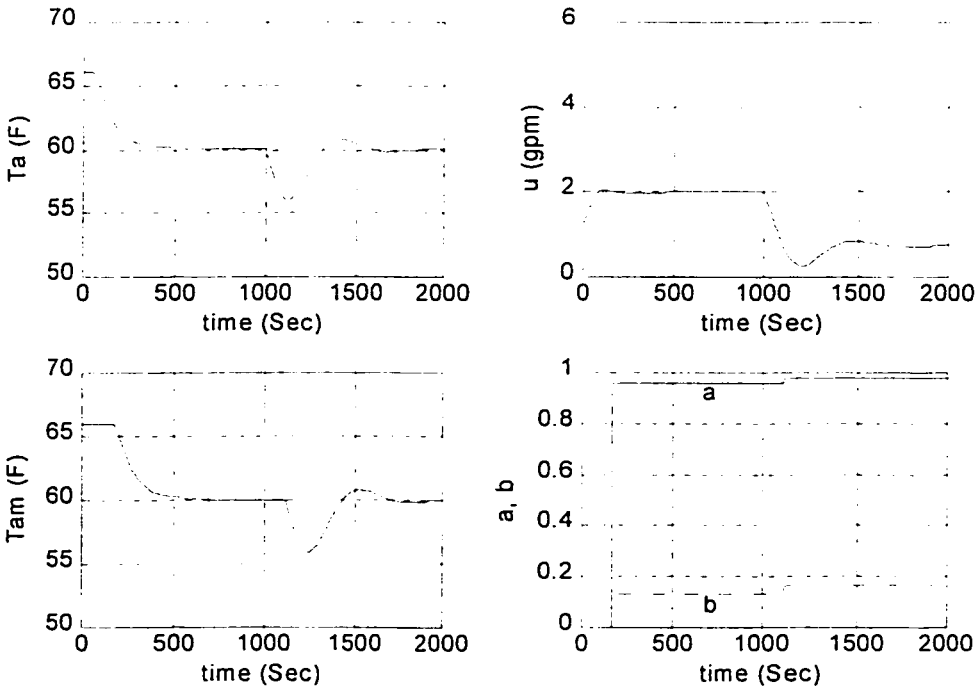

Figure 3.22 **Simulation result with delay of u for the model**

From Figure 3.22, we can say that the model response has almost the same shape compared to the plant response. This fact means that the online identification becomes accurate after $2 \times l$ sampling times counted from the start for the next update of the model.

3.7 Summary

Using the available input-output experimental data sets, both second-order and first-order models of the DAT system were developed. The simulation results show that the first-order model predicts the output responses of the DAT system as well as the second-order model. Based on this observation, the first-order model was chosen for online identification and control study. A simplified RLS algorithm for FOPDT model was derived and simulated for online implementation on an existing EMC system. Simulation results show that the simplified RLS algorithm is able to predict plant parameters efficiently showing faster convergence and smooth responses.

4. Real-time PID Control Design

Real-time PID control design methods will be discussed in this chapter, which include two heuristic adaptive control strategies and a self-tuning adaptive control strategy. We also assume that the controller is a digital PI controller and its structure can be expressed as follows.

$$K_d(z) = K_{pd} + \frac{K_{id}T}{2}\frac{z+1}{z-1} = \frac{\left(K_{pd} + \frac{K_{id}T}{2}\right)z + \left(\frac{K_{id}T}{2} - K_{pd}\right)}{z-1} \qquad (4.1)$$

This chapter is organized as follows. In section 4.1, two heuristic adaptive PI control strategies for a DAT system are presented. In addition, the design methods are described and simulation results for three sets of parameters of the plant and comparisons with constant PI control are given. In section 4.2, a self-tuning adaptive control strategy is proposed, and developed H_∞ PI tuning rules for discrete-time systems are given. Then, the comparisons with optimal LQR and Ziegler-Nichols adaptive PI controls, and the three adaptive PI control strategies are presented via simulation results. Also, the effects of noise and disturbances are shown. Finally, in sections 4.3, a summary of these developments is given.

4.1 Heuristic Adaptive Control

Based on the idea of tracking control and the characteristics of a DAT system, we have developed two structures for adaptive PI control. The first structure (see Figure 4.1) is based on the difference between the plant output and the model output for updating the

PI parameters. The second structure (see Figure 4.6) is based on the difference between the plant input and the control model output for updating the PI parameters.

4.1.1 Model Following Adaptive PI Control

We consider the structure for the model following adaptive PI control as shown in Figure 4.1.

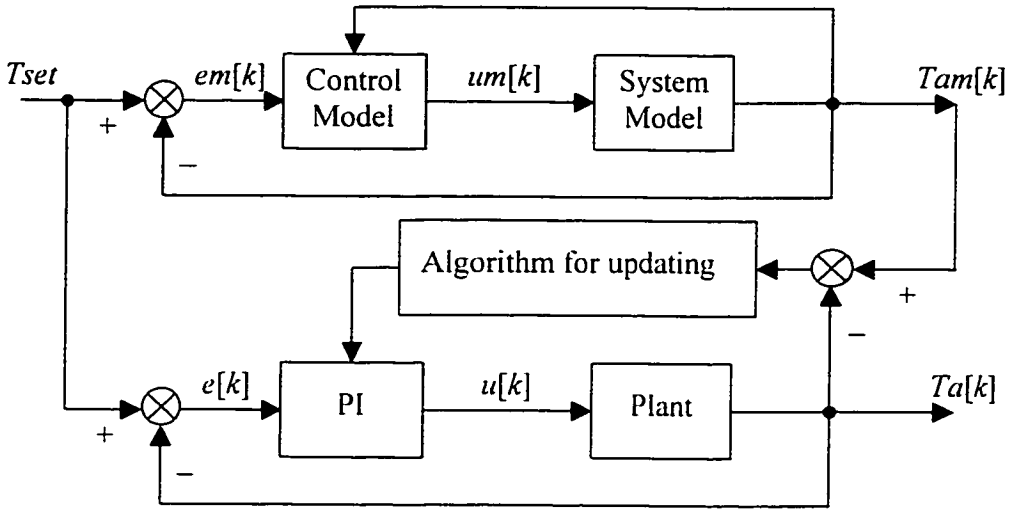

Figure 4.1 Model following adaptive PI control

The control algorithm is given by

$$um[k+1] = \begin{cases} u_{\max} & um_c \geq u_{\max} \\ um_c & u_{\min} < um_c < u_{\max} \\ u_{\min} & um_c \leq u_{\min} \end{cases} \qquad (4.2)$$

where $um_c = A_m um[k] - B_m K_m[k](Tset - Tam[k])$, $K_m[k] = \dfrac{1.1Ta1 - Tam[k]}{Ta1 - Tw1}$,

$$Tam[0] = Ta[0], \quad um[0] = \begin{cases} u_{\max} & K_{pd}[0]em[0] \geq u_{\max} \\ K_{pd}[0]em[0] & u_{\min} < K_{pd}[0]em[0] < u_{\max} \\ u_{\min} & K_{pd}[0]em[0] \leq u_{\min} \end{cases}.$$

In the above equations Tal is the inlet air temperature, Twl is the inlet water temperature, K_{pd} is the proportional gain of the PI controller, and K_m is an adaptive gain. The bigger the difference between Tal and Twl, the smaller will be the value of K_m. Similarly a lower Tam will lead to bigger value of K_m; A_m and B_m are adjustable coefficients for tuning um.

In a practical DAT system, $Tal > Twl$ and $Tal \geq Tam[k]$; so $K_m[k] > 0$ for all $k > 0$. Consider the case of $A_m = 1$ and $B_m > 0$. We note that when $Tam[k] > Tset$, more output from the controller than its previous value is needed and when Tam nears $Tset$ the control output should be slowed down. Similarly, when $Tam[k] < Tset$, we need less output from the controller than its previous value. Equation (4.2) satisfies these requirements.

To track the model output Tam in the model following adaptive PI control, we update the PI parameters depending on the difference between the plant output Ta and the model output Tam. The algorithm given below is proposed for updating the PI parameters.

$$K_{pd}[k+1] = \begin{cases} K_{pd\max} & K_{pdmc} \geq K_{pd\max} \\ K_{pdmc} & K_{pd\min} < K_{pdmc} < K_{pd\max} \\ K_{pd\min} & K_{pdmc} \leq K_{pd\min} \end{cases} \qquad (4.3)$$

$$K_{id}[k+1] = 0.01 K_{pd}[k+1] \qquad (4.4)$$

where $K_{pdmc} = K_{pd}[k] + \varsigma_m (Tam[k] - Ta[k])$; ς_m is a positive constant.

The algorithm assumes that the maximum and minimum ranges of the PI parameters are known apriori. Also the limiting values of the PI parameters must be negative in order to keep the closed loop system in a stable range.

The algorithm functions as follows.

When $Tam[k] > Ta[k]$, the next absolute values of K_{pd} and K_{id} does not increase. This may slow down the decrease in Ta or increase Ta. Also, when $Tam[k] < Ta[k]$, the next absolute values of K_{pd} and K_{id} does not decrease. This may lead to bigger u and faster decrease in Ta. In the case of $Tam[k] = Ta[k]$, the parameters keep the same values as before.

To explore the operating characteristics of the above control structure, we consider the example given below.

$$G(z) = \frac{Y(z)}{U(z)} = \frac{bz^{-16}}{1 - az^{-1}},\ Ta0 = 66.02\ ^{\circ}F,\ \text{sampling time}\ T = 4\,\text{sec}.$$

Figure 4.2 shows the implementation scheme using Simulink.

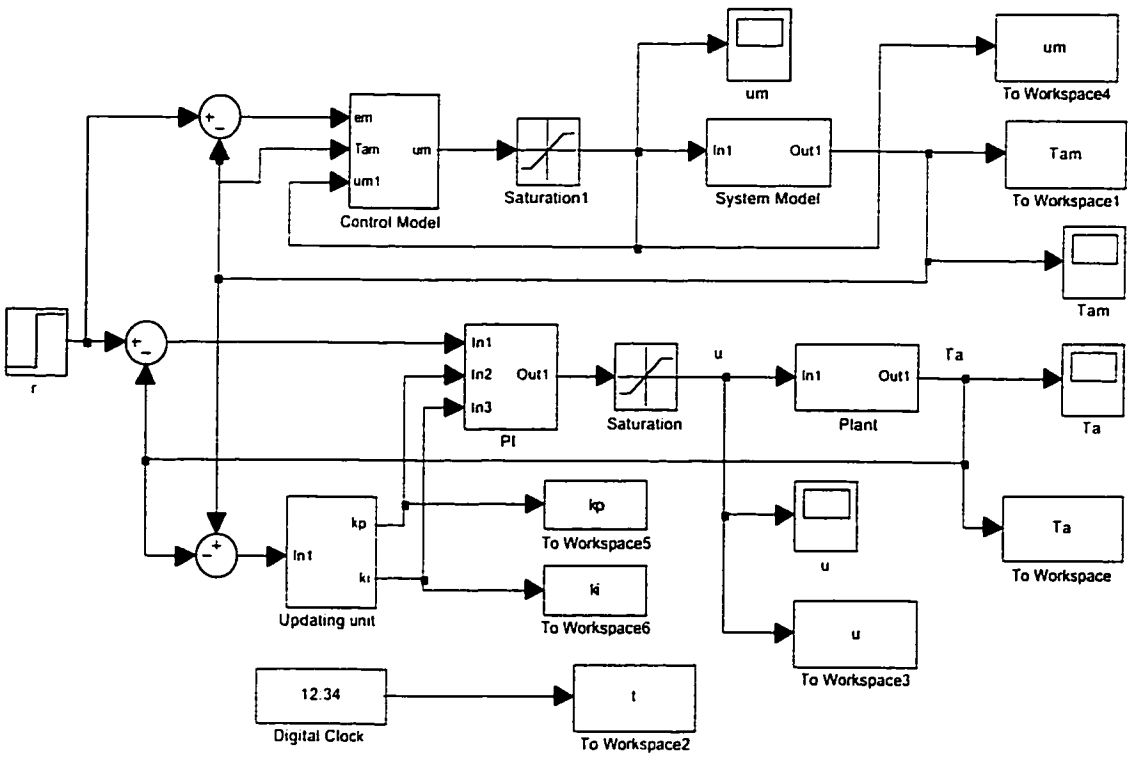

Figure 4.2 Simulation structure for model following adaptive PI control

Figures 4.3 – 4.5 show the model following adaptive PI control responses and the parameter updating results respectively for three sets of plant parameters:

Case-1: $a = 0.9558$ and $b = 0.1326$, for all time

Case-2: $a = \begin{cases} 0.9558 & t < 1000\,\text{sec} \\ 0.9758 & t \geq 1000\,\text{sec} \end{cases}$ and $b = \begin{cases} 0.1326 & t < 1000\,\text{sec} \\ 0.1626 & t \geq 1000\,\text{sec} \end{cases}$

Case-3: $a = \begin{cases} 0.9558 & t < 1000\,\text{sec} \\ 0.9358 & t \geq 1000\,\text{sec} \end{cases}$ and $b = \begin{cases} 0.1326 & t < 1000\,\text{sec} \\ 0.1026 & t \geq 1000\,\text{sec} \end{cases}$.

In each case, we chose the system model to be the same as in Case-1. The other parameters of the control algorithm were set as follows:

$A_m = 1$ and $B_m = 0.025$ for the control model, and $\varsigma_m = 0.01$, $K_{pd}[0] = -0.35$, $K_{id}[0] = -0.0035$, $K_{pd} \in [-0.8 \quad -0.1]$ for PI parameter updating algorithm, and $Tset = 60\,^{\circ}F$.

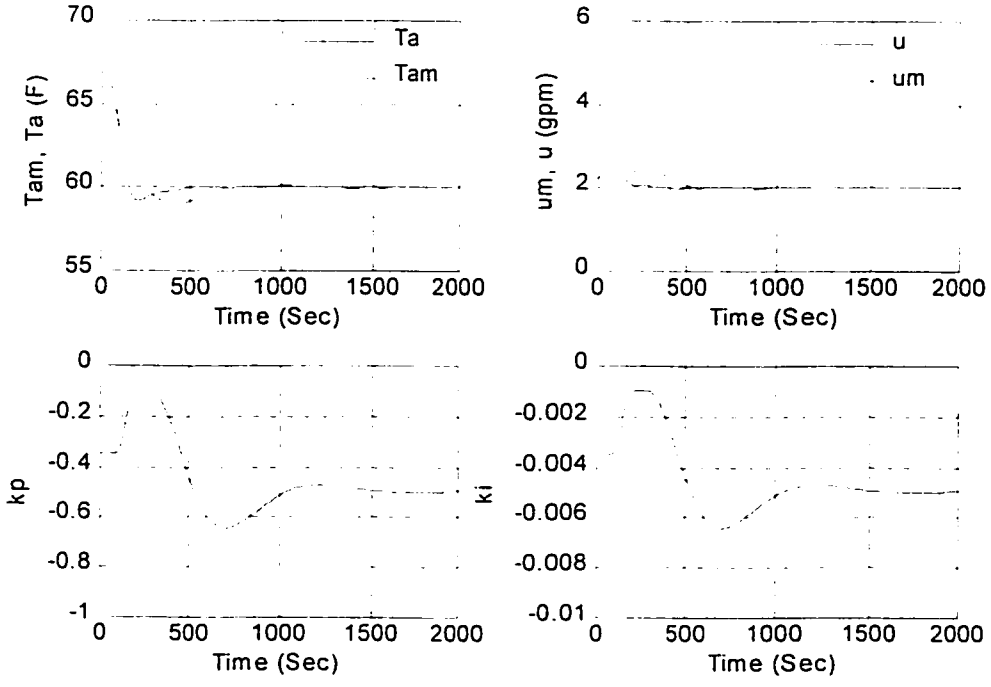

Figure 4.3 Model following adaptive PI control result for Case-1

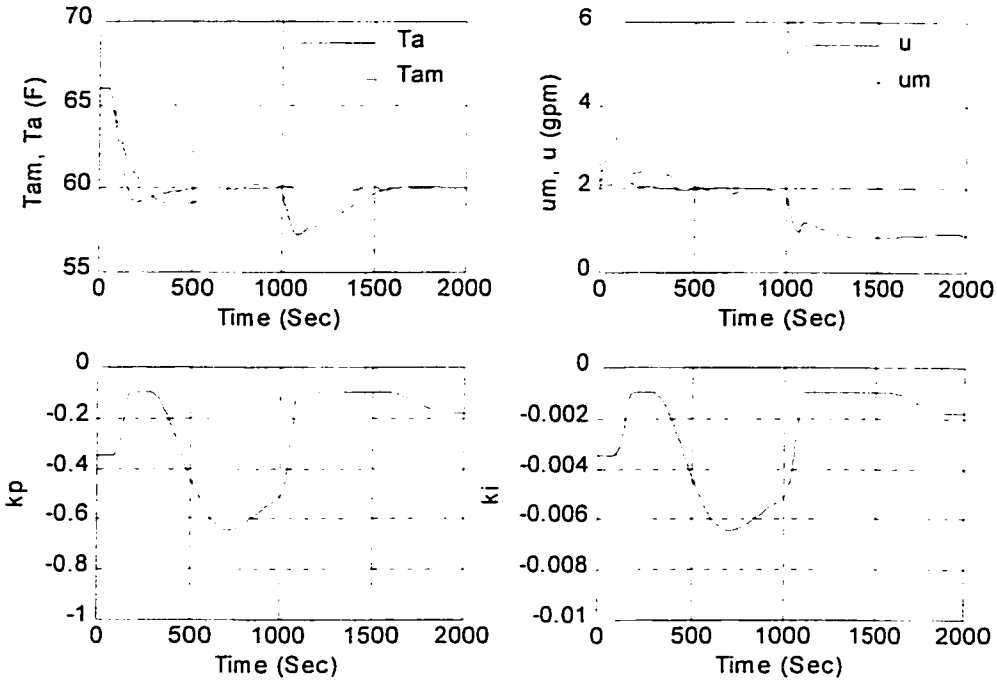

Figure 4.4 Model following adaptive PI control result for Case-2

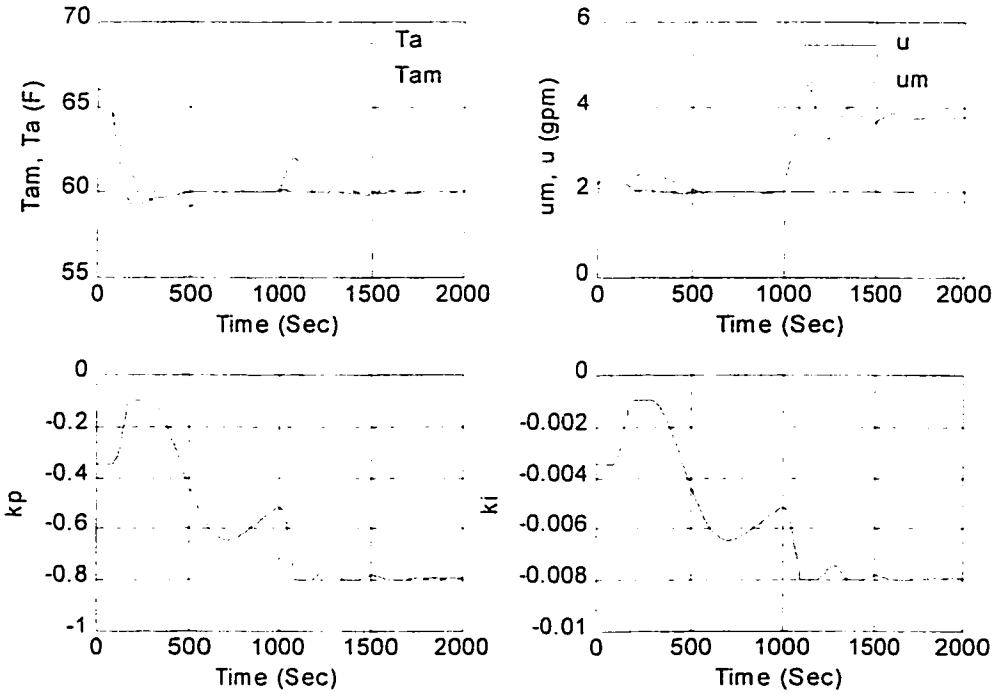

Figure 4.5 Model following adaptive PI control result for Case-3

The results in Figures 4.3 – 4.5 show that the model following adaptive PI control system works well given a proper range of K_{pd} and suitable values of parameter A_m, B_m and ς_m. The choice of A_m, B_m and ς_m is not very difficult and can be obtained with one or two trials. The results show that even though the plant parameters are different in the three cases simulated, the controller is able to adapt these changes and give good temperature control.

4.1.2 Control-Model Based Adaptive PI Control

From the standpoint of practical implementation, the model following structure proposed in Figure 4.1 requires higher memory and computations. In order to improve the operating efficiency, a more simplified approach is proposed in this section. The simplified approach is referred to as control-model based adaptive PI control and is depicted in Figure 4.6. As shown in Figure 4.6, this approach does not need the DAT system model.

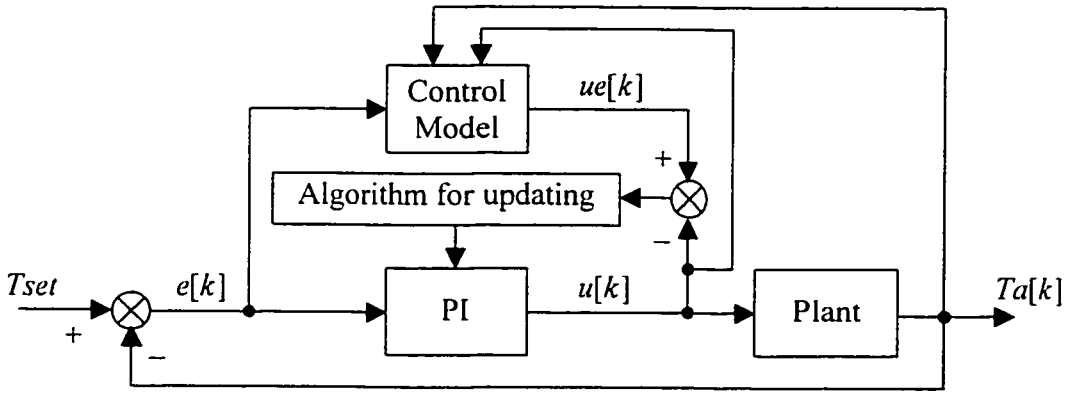

Figure 4.6 Control-model based adaptive PI control

The algorithm is described in the following equations:

$$ue[k+1] = \begin{cases} u_{\max} & ue_c \geq u_{\max} \\ ue_c & u_{\min} < ue_c < u_{\max} \\ u_{\min} & ue_c \leq u_{\min} \end{cases} \qquad (4.5)$$

where $ue_c = A_{cm}u[k] - B_{cm}ku[k](Tset - Ta[k])$, $\qquad ku[k] = \dfrac{1.001Tal - Ta[k]}{Tal - Twl}$.

$$ue[0] = \begin{cases} u_{\max} & K_{pd}[0]e[0] \geq u_{\max} \\ K_{pd}[0]e[0] & u_{\min} < K_{pd}[0]e[0] < u_{\max} \\ u_{\min} & K_{pd}[0]e[0] \leq u_{\min} \end{cases}.$$

In practical systems, $Tal > Twl$ and $Tal \geq Ta[k]$; so $ku[k] > 0$ for all $k > 0$. Consider the case with $A_{cm} = 1$ and $B_{cm} > 0$. When $Ta[k] > Tset$, we have $ue[k+1] > u[k]$; this means that we need more power in the next step to let Ta go to $Tset$; or $ue[k+1] = u[k] = u_{\max}$. Also, when $Ta[k] < Tset$, we have $ue[k+1] < u[k]$; this implies that we need to reduce the power in the next step to let Ta go to $Tset$; or $ue[k+1] = u[k] = u_{\min}$.

To achieve the desired power change, we change the PI parameters of K_{pd} and K_{id} which depend on the difference in $ue[k]$ and $u[k]$. The algorithm given below is proposed for updating the PI parameters.

$$K_{pd}[k+1] = \begin{cases} K_{pd\max} & K_{pdc} \geq K_{pd\max} \\ K_{pdc} & K_{pd\min} < K_{pdc} < K_{pd\max} \\ K_{pd\min} & K_{pdc} \leq K_{pd\min} \end{cases} \qquad (4.6)$$

$$K_{id}[k+1] = 0.01K_{pd}[k+1] \qquad (4.7)$$

where $K_{pdc} = K_{pd}[k] - \varsigma(ue[k] - u[k])$; ς is a positive constant.

The assumptions for the algorithm concerning the PI parameters are the same as that for the model following algorithm. From equation (4.5), we have for $A_{cm} = 1$,

$$ue[k] - u[k-1] = -B_{cm} ku[k-1](Tset - Ta[k-1]) = -B_{cm} ku[k-1]e[k-1]. \qquad (4.8)$$

Using the PI control structure given in equation (4.1) we can write

$$u[k-1] - u[k] = \left(K_{pd}[k-1] + \frac{T}{2} K_{id}[k-1] \right) e[k-1] + \left(\frac{T}{2} K_{id}[k] - K_{pd}[k] \right) e[k]. \qquad (4.9)$$

Combining equations (4.8) and (4.9), we obtain

$$ue[k] - u[k] = \left(K_{pd}[k-1] + \frac{T}{2} K_{id}[k-1] - B_{cm} ku[k-1] \right) e[k-1] + \left(\frac{T}{2} K_{id}[k] - K_{pd}[k] \right) e[k]. \qquad (4.10)$$

The equation shows that in a stable PI control system, we can attain $ue[k] = u[k]$ for large k since in the steady state, a stable PI control system has $e[k] = e[k-1] = 0$.

When $ue[k] = u[k]$, we have $K_{pd}[k+1] = K_{pd}[k]$ and $K_{id}[k+1] = K_{id}[k]$, i.e., they become constant from equations (4.6) and (4.7). In other words, if $ue[k] \neq u[k]$, K_{pd} and K_{id} will change allowing u to track ue until $ue[k] = u[k]$.

Simulations were conducted to compare the output responses of the adaptive control strategies (Figure 4.1 and 4.6) using the same sets of plant parameters as those used in Case-1, 2 and 3 in Figure 4.3 – 4.5.

Figure 4.7 shows the implementation scheme using Simulink. Figures 4.8 to 4.10 show the control-model based adaptive PI control responses for three cases, Case-1 to Case-3 respectively. In each case, we chose $A_{cm} = 1$ and $B_{cm} = 0.025$ for the control

model, and $\varsigma = 0.08$, $K_{pd}[0] = -0.35$, $K_{id}[0] = -0.0035$, $K_{pd} \in [-0.8 \quad -0.1]$ for the updating algorithm, and $Tset = 60\,^{\circ}F$.

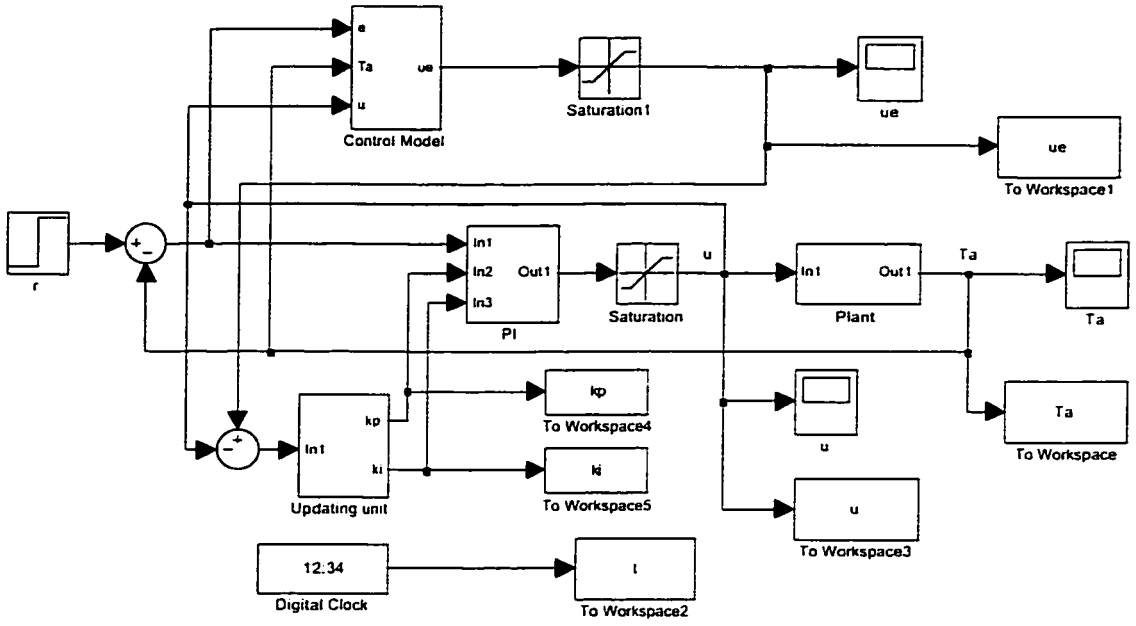

Figure 4.7 **Simulation structure for control-model based adaptive PI control**

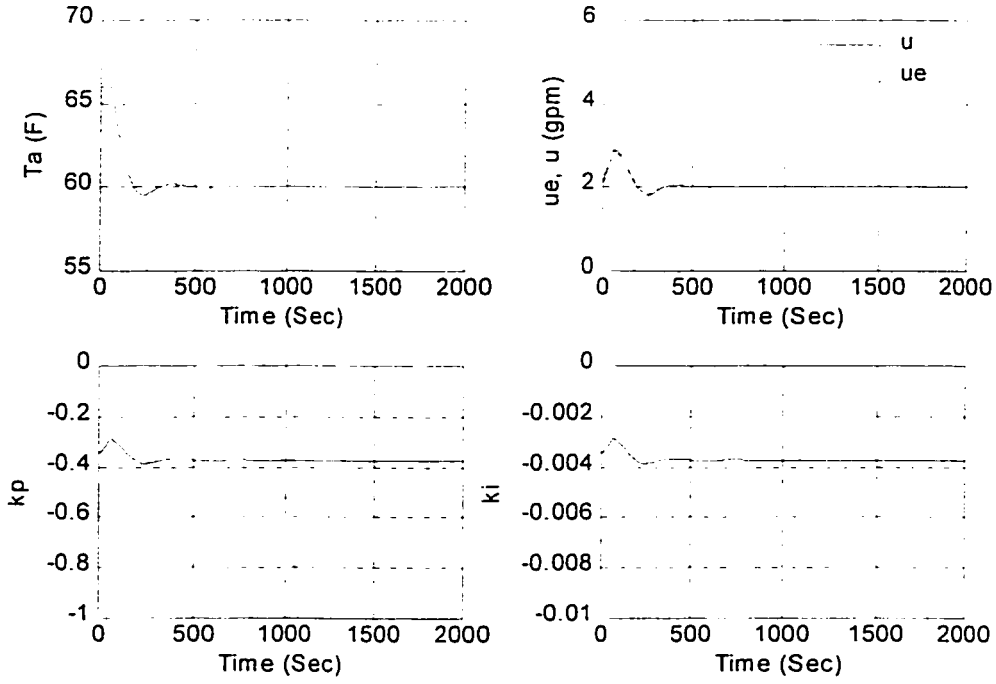

Figure 4.8 **Control-model based adaptive PI control result for the Case-1**

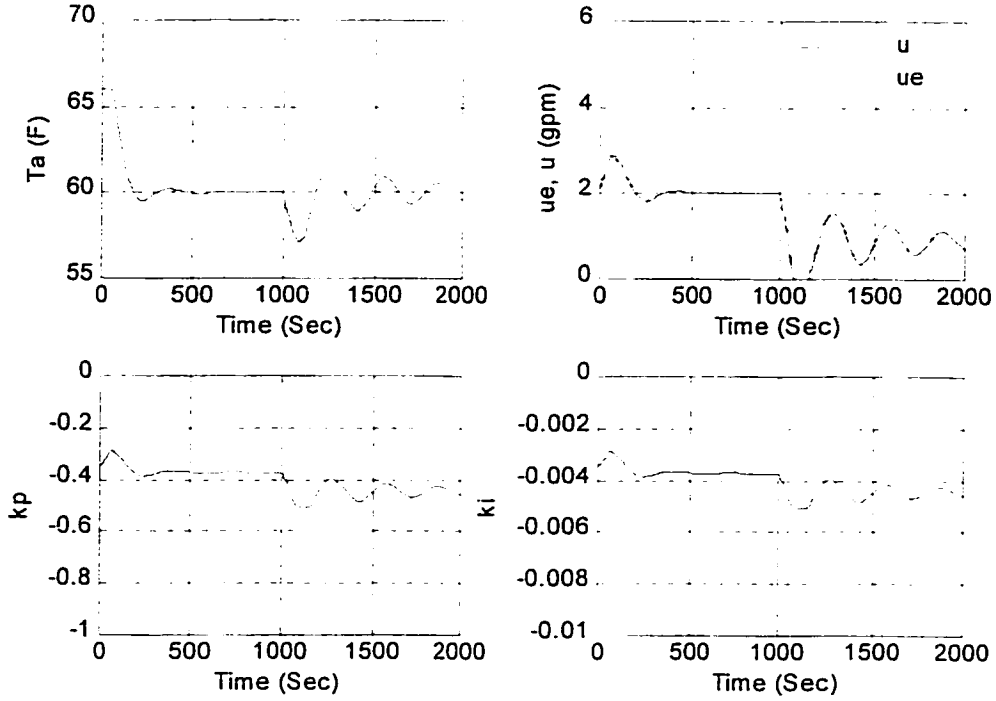

Figure 4.9 **Control-model based adaptive PI control result for the Case-2**

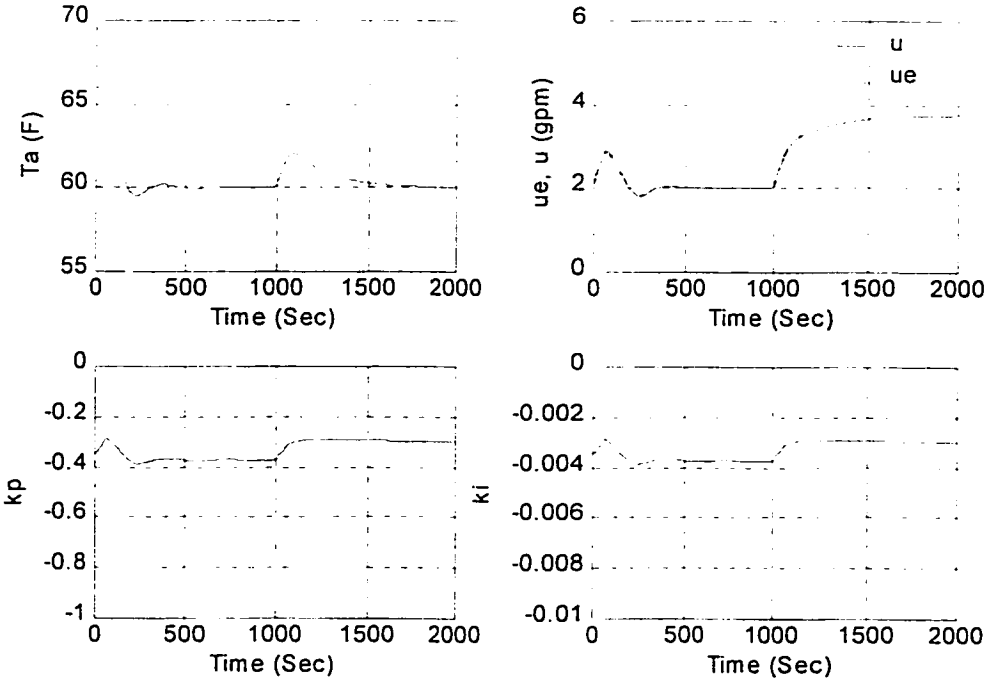

Figure 4.10 **Control-model based adaptive PI control result for the Case-3**

The results compare well with those shown in Figures 4.3 – 4.6 even though the control-model based PI adaptive control scheme does not require the DAT system model for its implementation.

4.1.3 Comparison of the Adaptive PI and Constant PI Control

Simulation runs were made to compare the proposed adaptive control strategies with constant gain PI control. In these simulations we chose three sets of PI parameters as initial values for model following adaptive (MFA) PI control and control-model based adaptive (CMBA) PI control and as constant gains for the PI controller for Case-1 of the plant model described in sections 4.1.1 and 4.1.2. The three sets of PI parameters were set as follows:

$[K_{pd}[0] \quad K_{id}[0]] = [-0.35 \quad -0.0035]$ (Set1), $[K_{pd}[0] \quad K_{id}[0]] = [-0.1 \quad -0.001]$ (Set2) and $[K_{pd}[0] \quad K_{id}[0]] = [-0.8 \quad -0.008]$ (Set3).

Figures 4.11 to 4.13 show the comparison of the model following adaptive (MFA) PI control and constant gain PI control using each set.

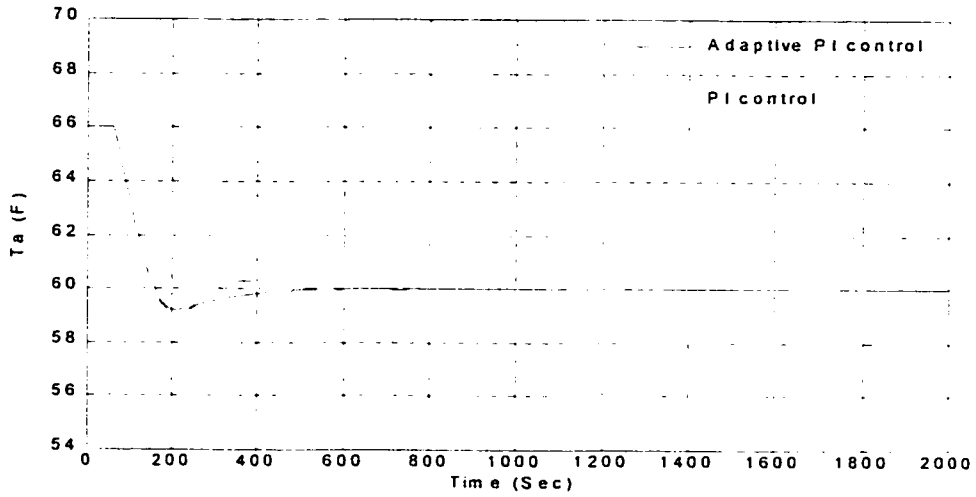

Figure 4.11 Comparison of MFA PI control and PI control using the Set1

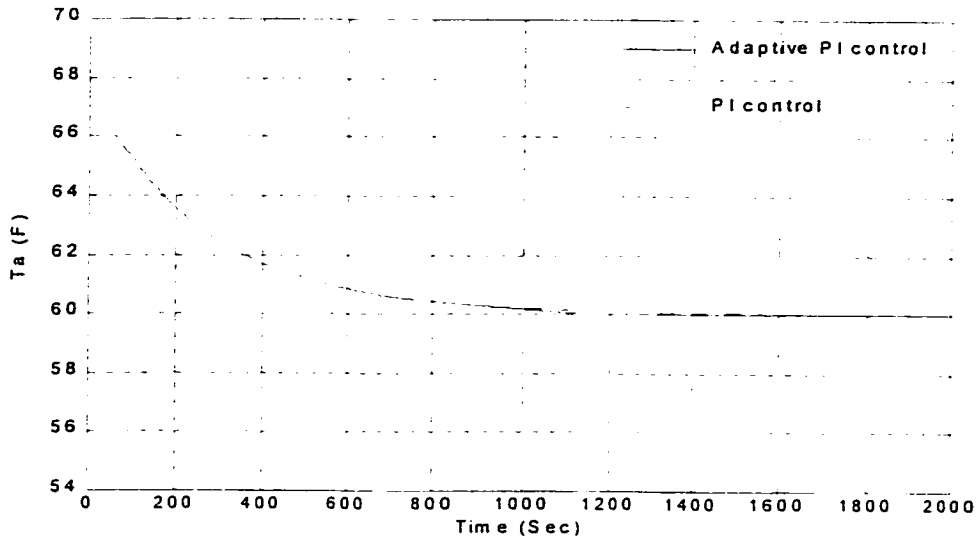

Figure 4.12 Comparison of MFA PI control and PI control using the Set2

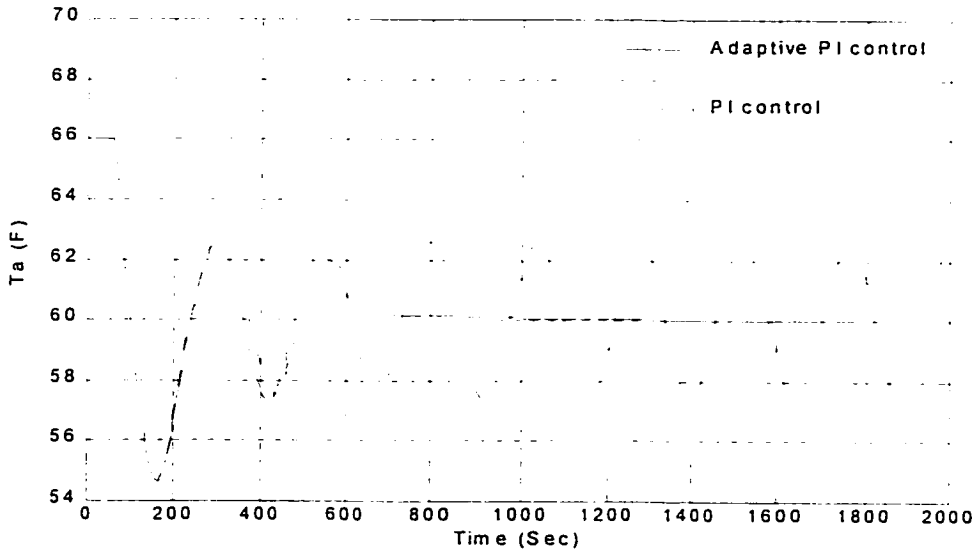

Figure 4.13 Comparison of MFA PI control and PI control using the Set3

It is apparent from Figure 4.13 that, even where the PI parameters were poorly chosen (as in Set3), the MFA PI controller adapts to the plant rapidly and gives stable responses, in contrast to the constant gain PI controller which needs retuning in the absence of which it exhibits severe oscillations.

Similar sets of results shown in Figure 4.14 – 4.16 show the comparison of control-model based adaptive (CMBA) and constant gain PI control. The results indicate that the developed adaptive PI control strategies (even using some values outside the stable range of a constant PI gain control system) make the system responses better than for constant gain PI control.

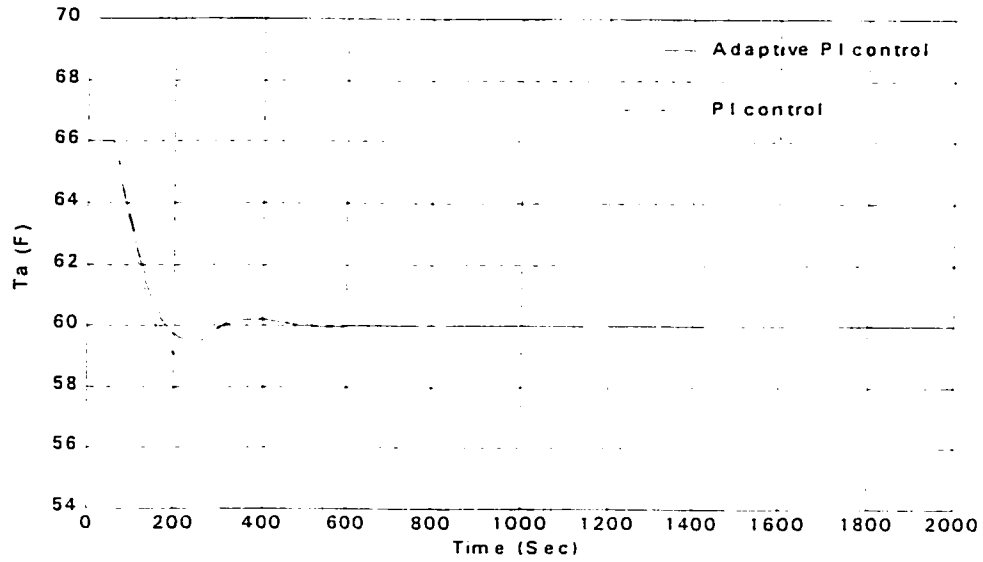

Figure 4.14 **Comparison of CMBA PI control and PI control using the Set1**

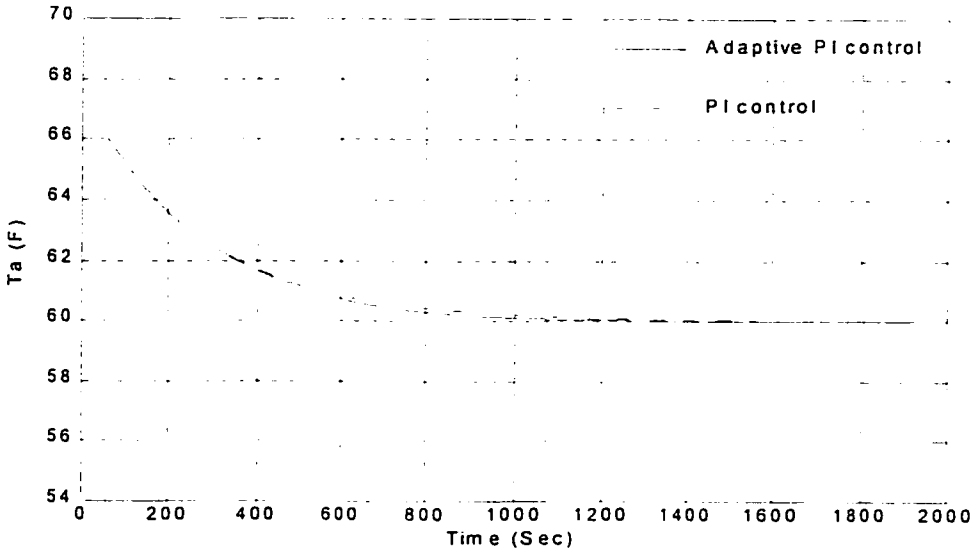

Figure 4.15 **Comparison of CMBA PI control and PI control using the Set2**

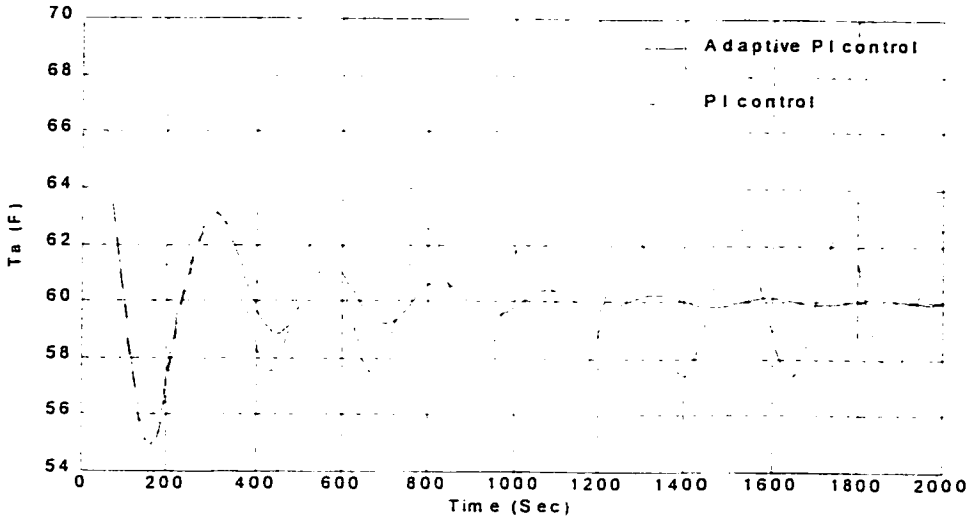

Figure 4.16 Comparison of CMBA PI control and PI control using the Set3

A question in the implementation of the proposed adaptive control strategies is how to select a proper range for the PI parameters. We address this issue in the following section.

4.1.4 Limiting Values of PI Parameters

Limiting values are needed in the PI parameter updating algorithms developed in sections 4.1.1 and 4.1.2. The limiting values must be negative to keep the PI closed loop system in stable state.

To choose the limiting values of the PI parameters, we consider the plant that is the same as in Figure 3.6, and then the closed PI control loop can be expressed as follows.

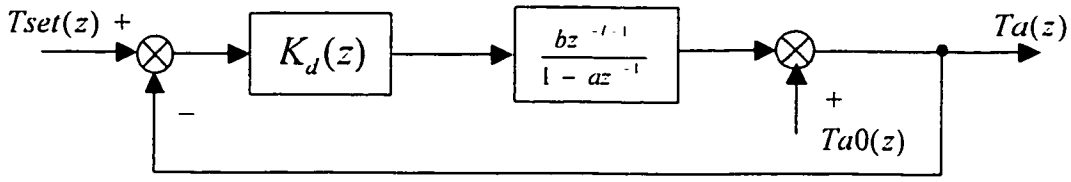

Figure 4.17 PI control loop for stability

Using equation 4.1, we attain the follow expression for Ta from the above loop.

$$Ta(z) = \frac{z^{l+2} - (1+a)z^{l+1} + az^{l}}{z^{l+2} - (1+a)z^{l+1} + az^{l} - b\left(K_{pd} + \frac{K_{id}T}{2}\right)z - b\left(\frac{K_{id}T}{2} - K_{pd}\right)} Ta0(z)$$

$$- \frac{b\left(K_{pd} + \frac{K_{id}T}{2}\right)z + b\left(\frac{K_{id}T}{2} - K_{pd}\right)}{z^{l+2} - (1+a)z^{l+1} + az^{l} - b\left(K_{pd} + \frac{K_{id}T}{2}\right)z - b\left(\frac{K_{id}T}{2} - K_{pd}\right)} Tset(z) \qquad (4.11)$$

Let $$D(z) = z^{l+2} - (1+a)z^{l+1} + az^{l} - b\left(K_{pd} + \frac{K_{id}T}{2}\right)z - b\left(\frac{K_{id}T}{2} - K_{pd}\right) \qquad (4.12)$$

and then the stability of the closed loop system is dependent on the coefficients of the polynomial $D(z)$.

To get the stable range of PI parameters in closed loop, we consider three cases. In the case with $l = 0$, we have

$$D(z) = z^{2} - \left[1 + a + b\left(K_{pd} + \frac{K_{id}T}{2}\right)\right]z + a - b\left(\frac{K_{id}T}{2} - K_{pd}\right). \qquad (4.13)$$

If we assume that $a_1 = -\left[1 + a + b\left(K_{pd} + \frac{K_{id}T}{2}\right)\right]$ and $a_2 = a - b\left(\frac{K_{id}T}{2} - K_{pd}\right)$.

Using Jury's stability test, the stability of the closed loop system requires that $-1 < a_2 < 1$, $1 + a_2 > a_1$ when $a_1 \geq 0$ and $1 + a_2 > -a_1$ when $a_1 \leq 0$. Using the relation of $K_{pd} = 0.01K_{id}$, the requirement becomes

$$-\frac{2(1+a)}{b(2-0.01T)} < K_{pd} < \frac{2(1-a)}{b(2-0.01T)},\ K_{pd} > -\frac{1+a}{b} \text{ and } K_{pd} < 0$$

under the condition of $T < 200$ Sec, i.e.

$$-\frac{1+a}{b} < K_{pd} < 0 \text{ and } T < 200 \text{ Sec.} \qquad (4.14)$$

If we let $l = 1$, we have

$$D(z) = z^3 - (1+a)z^2 + \left[a - b\left(K_{pd} + \frac{K_{id}T}{2}\right)\right]z - b\left(\frac{K_{id}T}{2} - K_{pd}\right). \qquad (4.15)$$

Considering $K_{pd} = 0.01K_{id}$, we know that the roots of

$$1 - K_{pd}\frac{\left(1 + \frac{0.01T}{2}\right)z + \left(\frac{0.01T}{2} - 1\right)}{z-1}\frac{b}{z^2 - az} = 0 \qquad (4.16)$$

are the same as the roots of $D(z) = 0$. Therefore, let

$$KG(z) = \frac{-b\left(1 + \frac{0.01T}{2}\right)z - b\left(\frac{0.01T}{2} - 1\right)}{z^3 - (1+a)z^2 + az} \qquad (4.17)$$

we can use the Nyquist criterion to get the stable range of K_{pd} for the closed loop system.

For $l \geq 2$, we have

$$D(z) = z^{l+2} - (1+a)z^{l+1} + az^l - b\left(K_{pd} + \frac{K_{id}T}{2}\right)z - b\left(\frac{K_{id}T}{2} - K_{pd}\right). \qquad (4.18)$$

Similar to the case of $l = 1$, here we let

$$KG(z) = \frac{-b\left(1 + \frac{0.01T}{2}\right)z - b\left(\frac{0.01T}{2} - 1\right)}{z^{l+2} - (1+a)z^{l+1} + az^l} \qquad (4.19)$$

we can also use the Nyquist criterion for

$$1 + K_{pd} KG(z) = 0 \qquad (4.20)$$

to get the stable range of K_{pd} for the closed loop system.

Now, we give an example for choosing limiting values for the PI parameters in the proposed adaptive PI control algorithms. We consider the same plant as the one given in section 4.1.1 and assume that $l = 15$, $T = 4$ and that the possible values for a and b are in the ranges $a \in [0.9358 \quad 0.9758]$ and $b \in [0.1026 \quad 0.1626]$ which cover Case-1 to Case-3 described in section 4.1.1.

To choose the limiting values for the above example, we use $a_{\max} = 0.9758$ and $b_{\max} = 0.1626$ for getting the Nyquist graph, and then we have

$$KG(z) = \frac{-0.1659z + 0.1593}{z^{17} - 1.9758z^{16} + 0.9758z^{15}} = \frac{-0.1659z + 0.1593}{z^{15}(z-1)(z-0.9758)}.$$

To avoid numerical difficulties associated with the pole $z = 1$, we use $z = 0.9999$ then the transfer function $KG(z)$ becomes

$$KG(z) \approx \frac{-0.1659z + 0.1593}{z^{15}(z-0.9999)(z-0.9758)}.$$

Figure 4.18 shows the Nyquist graphs for the above transfer function. We also can consider $z - 1 = re^{j\phi}$ where $r << 1$ and $-\frac{\pi}{2} \le \phi \le \frac{\pi}{2}$ for plotting z close to 1 to get almost the same Nyquist graphs. By Nyquist criterion, we obtain

$$1.9039 < -\frac{1}{K_{pd}} < \infty \qquad \Rightarrow \qquad -0.5252 \le K_{pd} < 0.$$

Therefore, we can take $K_{pd\min} = -0.525$ and $K_{pd\max} = 0.1K_{pd\min} = -0.0525$.

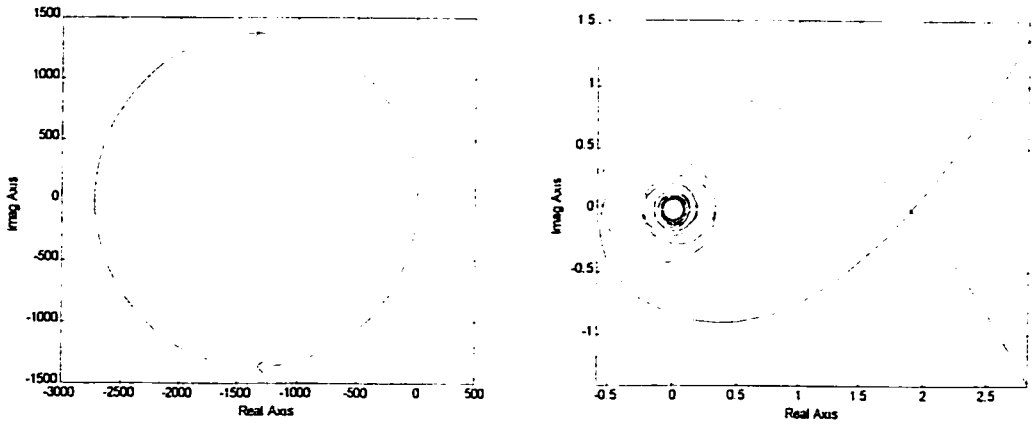

Figure 4.18 Nyquist plots

4.2 H_∞ Adaptive PI Control

In this section, we explore a PI tuning rule for first-order plus dead-time (FOPDT) models in discrete-time systems which is dependent on PID tuning rules based on loop-shaping H_∞ control for FOPDT models in continuous-time systems. We refer to this method as H_∞ PI tuning rules. We also study an adaptive control structure, in which H_∞ PI tuning rules will be used. Finally, comparisons between H_∞ adaptive control and (i) LQR optimal control, (ii) Ziegler-Nichols PI control, (iii) model following adaptive (MFA) PI control and (iv) the control-model based adaptive (CMBA) PI control will be made using the FOPDT model of the DAT system.

4.2.1 PID Tuning Rules Based on Loop-shaping H_∞ Control

PID tuning rules based on loop-shaping H_∞ control for first-order plus dead-time (FOPDT) models in continuous-time systems were proposed by Tan et al. [39]. We first briefly describe the rule in this section. Details can be found in [39].

Given a plant $p(s)$ that can be described by an FOPDT model:

$$p(s) = \frac{k_s}{T_s s + 1} e^{-\tau s} \tag{4.21}$$

the PID structure based on loop-shaping H_∞ control becomes

$$K(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \frac{1}{1 + T_f s} \tag{4.22}$$

and the PID tuning rules are given by

$$K_p = \frac{0.265\lambda + 0.307}{k_s} \left(\frac{T_s}{\tau} + 0.5 \right) \tag{4.23}$$

$$T_i = T_s + \frac{\tau}{2} \tag{4.24}$$

$$T_d = \frac{T_s \tau}{2T_s + \tau} \tag{4.25}$$

$$T_f = \frac{\tau}{5.314\lambda + 0.951} \tag{4.26}$$

where λ is a design parameter.

4.2.2 H_∞ PI Tuning Rules

In this section, we transfer the PID tuning based on loop-shaping H_∞ control in continuous-time systems to PI tuning in discrete-time systems by using the equivalent transform method. First, we assume that we have already designed a PID controller for a continuous-time system by PID tuning based on loop-shaping H_∞ control. For the plant shown in Figure 3.7, the controller becomes

$$K(s) = -K_p\left(1 + \frac{1}{T_i s} + T_d s\right)\frac{1}{1 + T_f s} \qquad (4.27)$$

and the parameters can be tuned by using equations (4.23) to (4.26). Secondly, using the transformation

$$s = \frac{2}{T}\frac{z-1}{z+1} \qquad (4.28)$$

we have $K(z) = -K_p\left(1 + \frac{T}{2T_i}\frac{z+1}{z-1} + \frac{2T_d}{T}\frac{z-1}{z+1}\right)\frac{1}{1 + \frac{2T_f}{T}\frac{z-1}{z+1}}$

$$= -K_p \frac{z^2 - 1 + \frac{T}{2T_i}(z^2 + 2z + 1) + \frac{2T_d}{T}(z^2 - 2z + 1)}{(z-1)\left(\left(\frac{2T_f}{T} + 1\right)z + 1 - \frac{2T_f}{T}\right)}$$

$$= -K_p \frac{\left(1 + \frac{T}{2T_i} + \frac{2T_d}{T}\right)z^2 + \left(\frac{T}{T_i} - \frac{4T_d}{T}\right)z - 1 + \frac{T}{2T_i} + \frac{2T_d}{T}}{(z-1)\left(\left(\frac{2T_f}{T} + 1\right)z + 1 - \frac{2T_f}{T}\right)}. \qquad (4.29)$$

Then, using $K_d(z)$ to substitute for $K(z)$, we have

$$-K_p \frac{\left(1 + \frac{T}{2T_i} + \frac{2T_d}{T}\right)z^2 + \left(\frac{T}{T_i} - \frac{4T_d}{T}\right)z - 1 + \frac{T}{2T_i} + \frac{2T_d}{T}}{(z-1)\left(\left(\frac{2T_f}{T} + 1\right)z + 1 - \frac{2T_f}{T}\right)} = \frac{\left(K_{pd} + \frac{K_{id}T}{2}\right)z + \left(\frac{K_{id}T}{2} - K_{pd}\right)}{z-1}$$

i.e. $$-K_p\left(\left(1 + \frac{T}{2T_i} + \frac{2T_d}{T}\right)z^2 + \left(\frac{T}{T_i} - \frac{4T_d}{T}\right)z - 1 + \frac{T}{2T_i} + \frac{2T_d}{T}\right)$$

$$= \left(\left(\frac{2T_f}{T} + 1\right)z + 1 - \frac{2T_f}{T}\right)\left(\left(K_{pd} + \frac{K_{id}T}{2}\right)z + \left(\frac{K_{id}T}{2} - K_{pd}\right)\right) \qquad (4.30)$$

This requires that

$$-K_p\left(1+\frac{T}{2T_i}+\frac{2T_d}{T}\right)=\left(K_{pd}+\frac{K_{id}T}{2}\right)\left(\frac{2T_f}{T}+1\right) \qquad (4.31)$$

$$-K_p\left(\frac{T}{T_i}-\frac{4T_d}{T}\right)=\left(\frac{K_{id}T}{2}-K_{pd}\right)\left(\frac{2T_f}{T}+1\right)+\left(K_{pd}+\frac{K_{id}T}{2}\right)\left(1-\frac{2T_f}{T}\right)$$

$$=K_{id}T-K_{pd}\frac{4T_f}{T} \text{ and} \qquad (4.32)$$

$$-K_p\left(-1+\frac{T}{2T_i}+\frac{2T_d}{T}\right)=\left(\frac{K_{id}T}{2}-K_{pd}\right)\left(1-\frac{2T_f}{T}\right). \qquad (4.33)$$

Also $\lim_{z\to 1}(z-1)K_d(z)=\lim_{z\to 1}(z-1)K(z)$ ➔ $K_{id}=-\frac{K_p}{T_i}$ (4.34)

in the steady state.

These equations lead to

$$K_{pd}=-K_p\frac{1+\frac{2T_d}{T}-\frac{T_f}{T_i}}{\frac{2T_f}{T}+1} \qquad (4.35)$$

from the combination of equations (4.31) and (4.34),

$$K_{pd}=-K_p\frac{T_d}{T_f} \qquad (4.36)$$

from the combination of equations (4.32) and (4.34),

and $$K_{pd}=-K_p\frac{1-\frac{2T_d}{T}-\frac{T_f}{T_i}}{1-\frac{2T_f}{T}} \qquad (4.37)$$

from the combination of equations (4.33) and (4.34).

From equations of (4.35) to (4.37), we know that under the condition

$$1-\frac{T_f}{T_i}=\frac{T_d}{T_f} \qquad (4.38)$$

we can get the same K_{pd}. The condition is satisfied when

$$\lambda=\frac{\frac{\tau}{T_s}-0.951}{5.314} \qquad \text{or} \qquad \lambda=0.1974.$$

This condition means that the PI controller $K_d(z)$ in a discrete-time system is equivalent (for fixed λ) to the PID controller tuned by the PID tuning rules based on loop-shaping H_∞ control in continuous-time systems for the FOPDT model.

Considering the robust stability of the closed loop system (λ cannot be chosen too large) and to have optimal results (equation (4.38) must be satisfied), we may choose

$$\lambda=0.1974 \qquad (4.39)$$

$$\text{for all } \tau, \text{ or } \lambda=\frac{\frac{\tau}{T_s}-0.951}{5.314} \qquad (4.40)$$

when $2T_s \le \tau \le 8.922T_s$ for getting faster response of the closed loop system.

When $\tau = lT$, we have $a=e^{-\frac{T}{T_s}}$ and $b=k_s(1-a)$. Therefore

$$T_s=-\frac{T}{\ln a} \qquad (4.41)$$

$$\text{and} \qquad k_s=\frac{b}{1-a}. \qquad (4.42)$$

Finally, the H_∞ PI tuning rules for discrete-time systems are given by

$$K_{id} = -\frac{K_p}{T_i} \tag{4.43}$$

and $$K_{pd} = -K_p \frac{T_d}{T_f} \tag{4.44}$$

where $$K_p = \frac{(0.265\lambda + 0.307)(1-a)}{b}\left(-\frac{1}{l \ln a} + 0.5\right) \tag{4.45}$$

$$T_i = -\frac{T}{\ln a} + \frac{lT}{2} \tag{4.46}$$

$$T_d = \frac{Tl}{2 - l \ln a} \tag{4.47}$$

$$T_f = \frac{lT}{5.314\lambda + 0.951}. \tag{4.48}$$

The design parameter λ can be chosen as $\lambda = 0.1974$ for all l, or $\lambda = \frac{-l \ln a - 0.951}{5.314}$ for $-\frac{2}{\ln a} \le l \le -\frac{8.922}{\ln a}$ in order to get faster response of the closed loop system.

4.2.3 H_∞ Adaptive PI Control

In this section, we present an adaptive PI control structure for the DAT system that can be expressed as an FOPDT discrete-time system model, in which H_∞ PI tuning rules are used. The proposed H_∞ adaptive PI control structure is shown in Figure 4.19.

The function of the identifier is to estimate the parameters of the plant that can be expressed as an FOPDT discrete-time system model, and the function of the PI tuner is to

update the parameter values of the PI controller depending on new parameters of the plant using H_∞ PI tuning rules during online operation.

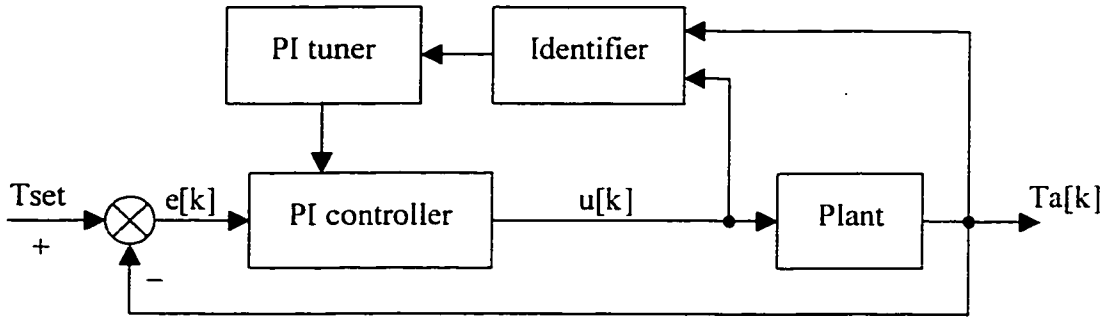

Figure 4.19 H_∞ adaptive PI control structure

The H_∞ PI tuning rules presented in section 4.2.2, which are similar to the Ziegler-Nichols tuning rule (see Appendix A), are easier to realize in PPCL (Powers Process Control Language). If we consider the implementation of the identifier as the advanced online identification proposed in section 3.6.2, the H_∞ adaptive PI control is realizable in PPCL under online operating conditions.

4.2.4 Simulation and Comparison

We consider the same plant model as the one used in Case-1 in section 4.1.1. The implementation scheme in Simulink is shown in Figure 4.20. Figure 4.21 shows the H_∞ adaptive PI control response for a setpoint change. The PI parameters were updated by the H_∞ PI tuning rules. Figure 4.22 shows the responses using an LQR optimal adaptive PI controller (The design details of the LQR optimal PI controller are presented in Appendix B). The LQR results correspond to the following values: $N = 200$,

$Q = \begin{bmatrix} 0.001 & 0 \\ 0 & 80000 \end{bmatrix}$ and $R = 0.1$. Also shown in Figure 4.23 are the Ziegler-Nichols based adaptive PI control responses in which the PI parameters were updated by using the Ziegler-Nichols tuning rules for FOPDT models (The tuning rules for a discrete-time system are presented in Appendix A). In these simulations, the implementation of the identifier is the same as the one used in section 3.6.2; the initial values of the PI parameters are $K_{pd}[0] = -0.35$ and $K_{id}[0] = -0.0035$; and

$$Tset = \begin{cases} 60^\circ F & 0 \le t < 1000\,\text{sec} \\ 62^\circ F & t \ge 1000\,\text{sec} \end{cases}.$$

Figure 4.24 shows the results for the model following adaptive PI control, and Figure 4.25 shows the results for the control-model based adaptive PI control with the same initial values of PI parameters and *Tset* as the other cases.

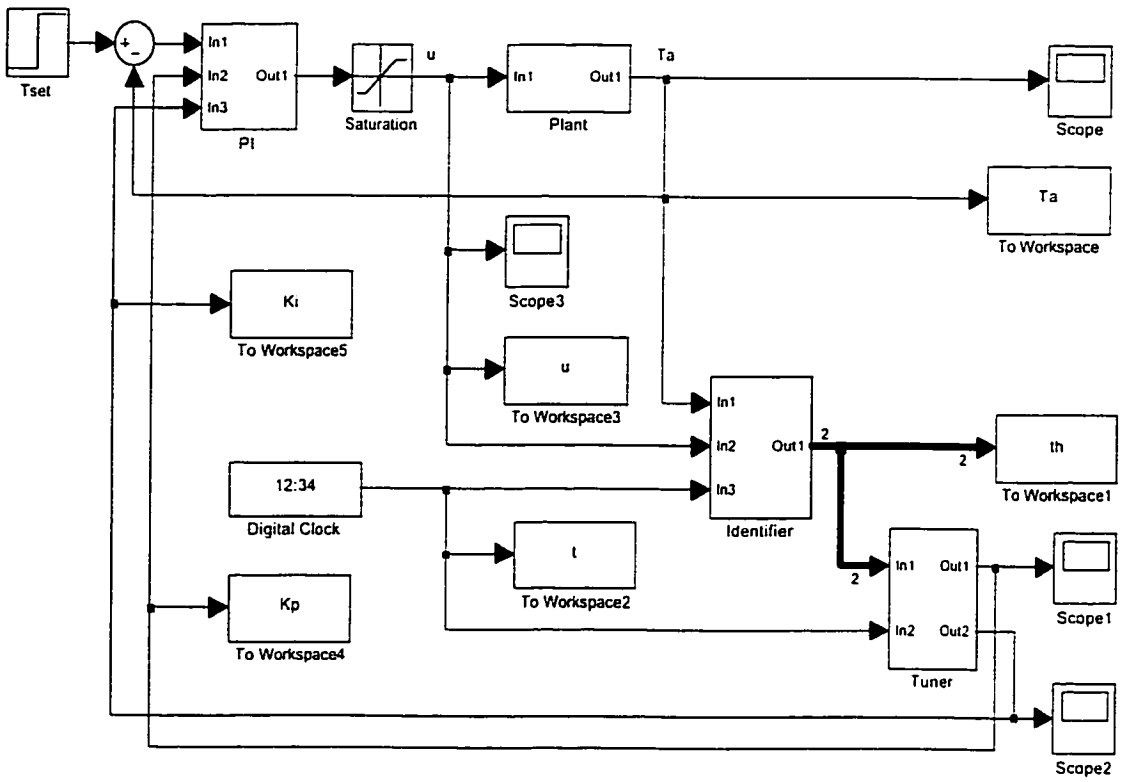

Figure 4.20 Implementation of H_∞ adaptive PI control in Simulink

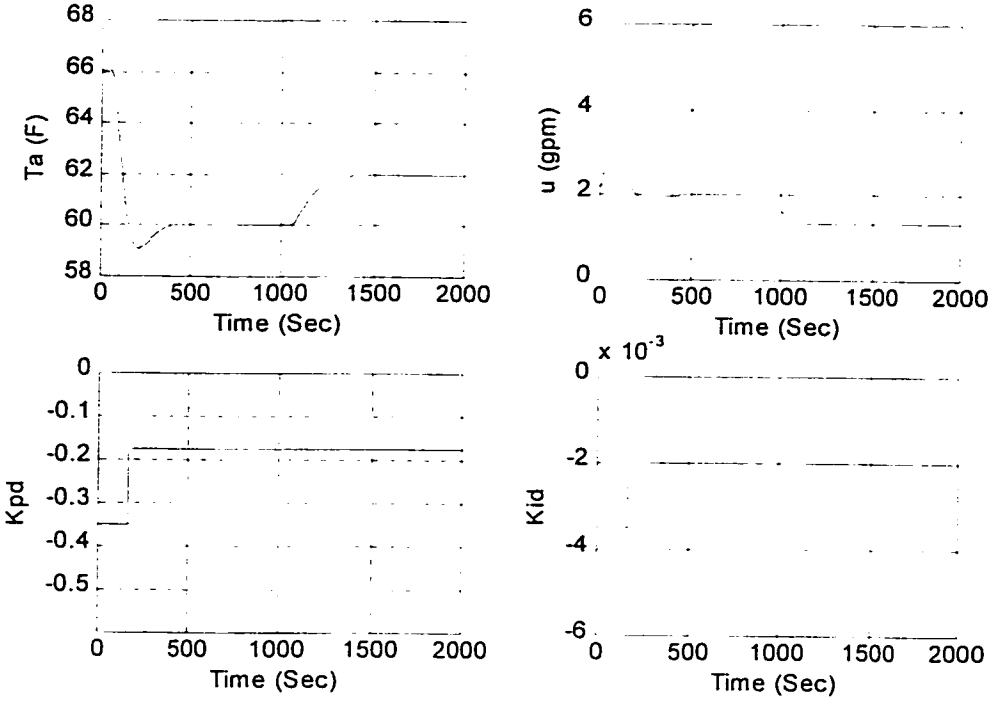

Figure 4.21 H_∞ **adaptive PI control response for setpoint change**

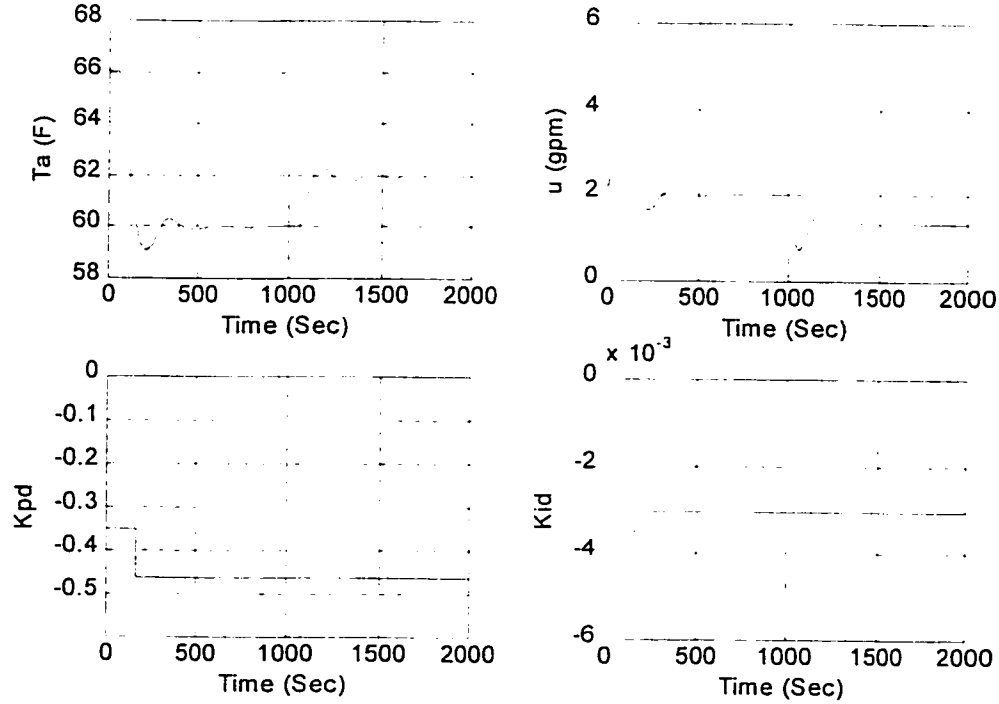

Figure 4.22 **LQR adaptive PI control response for setpoint change**

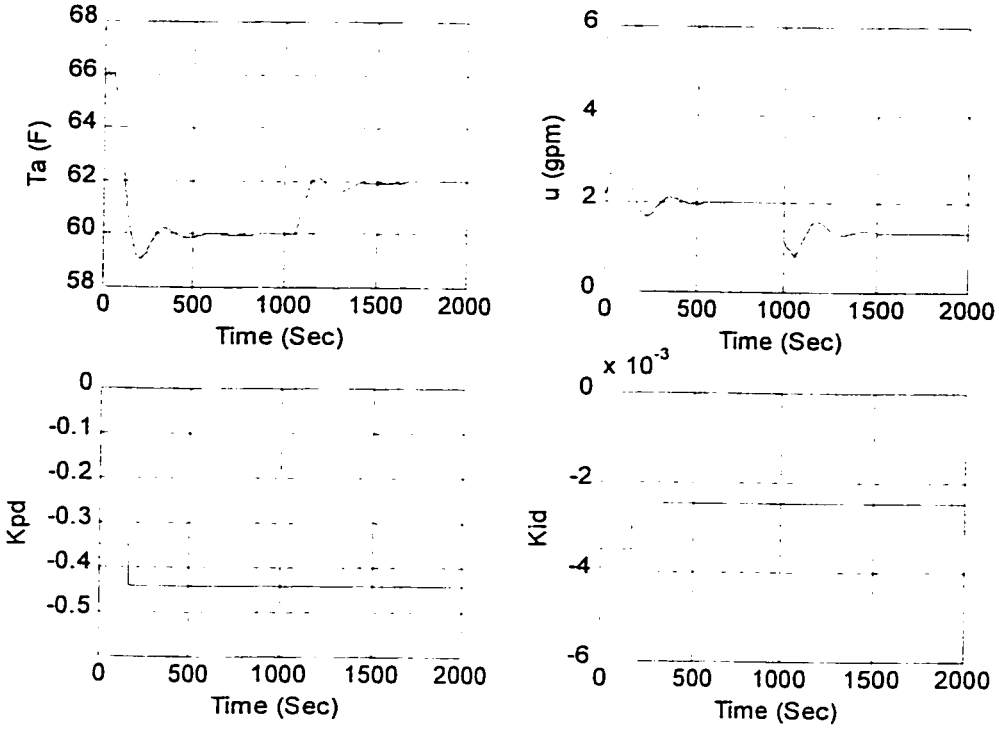

Figure 4.23 **Ziegler-Nichols adaptive PI control response for setpoint change**

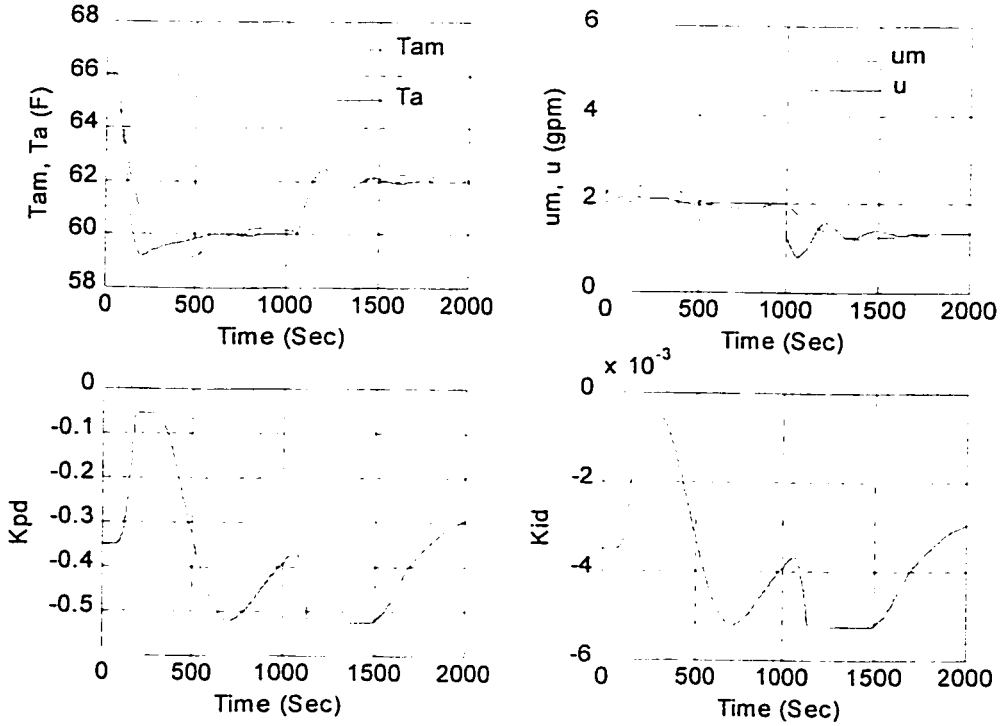

Figure 4.24 **MFA PI control response for setpoint change**

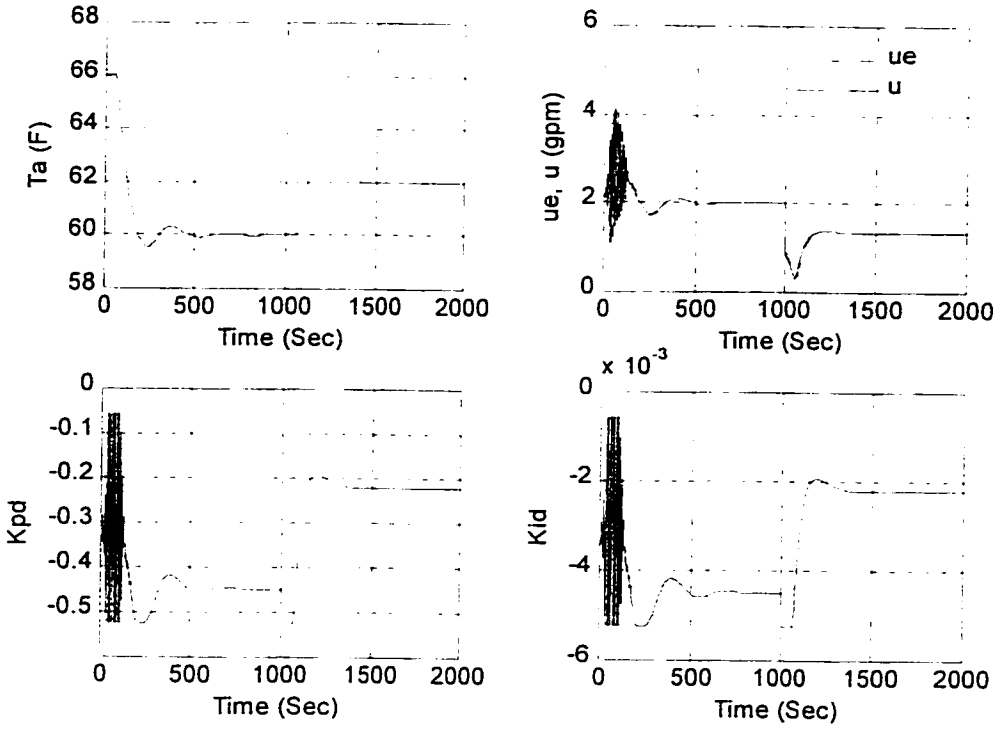

Figure 4.25 CMBA PI control response for setpoint change

The simulation results in Figures 4.21 – 4.25 show that the H_∞ adaptive PI control response for setpoint change (we focus on the curve of response from 1000 Sec to 2000 Sec) is the best one even compared with the LQR optimal control case. The response has a smaller overshoot and a shorter settling time. The PI parameter tuning in the LQR optimal control case is very difficult to implement in real time on the EMCS because we need a lot of computation time to get a new set of PI parameters and it is hard to decide which values we should choose for Q and R in order to get a good response. Fortunately, the H_∞ PI tuning rules in the H_∞ adaptive PI control are very easy to implement like the Ziegler-Nichols PI tuning rules in the Ziegler-Nichols adaptive PI control. Online identification is needed in H_∞ adaptive PI control, LQR optimal control and Ziegler-Nichols adaptive PI control, but it is not needed in CMBA PI control. It can

be both either within the online identification or without in MFA PI control. The CMBA PI control is the easiest control strategy to implement.

Figure 4.26 shows the H_∞ adaptive PI control responses for a setpoint change. In this simulation three different plant models were used. These are:

Case 1: $[a \quad b] = [0.9358 \quad 0.1026]$ and $l = 15$

Case 2: $[a \quad b] = [0.9558 \quad 0.1326]$ and $l = 15$

Case 3: $[a \quad b] = [0.9758 \quad 0.1626]$ and $l = 15$.

In all three cases, the initial values of the PI parameters were $K_{pd}[0] = -0.35$ and $K_{id}[0] = -0.0035$; and

$$Tset = \begin{cases} 60^\circ F & 0 \le t < 1000 \text{sec} \\ 62^\circ F & t \ge 1000 \text{sec} \end{cases}.$$

As shown in Figure 4.26 after 1000 Sec, the responses for the three cases are the same. This means that we can obtain the same response for different parameters (a and b) of the FOPDT model using H_∞ adaptive PI control.

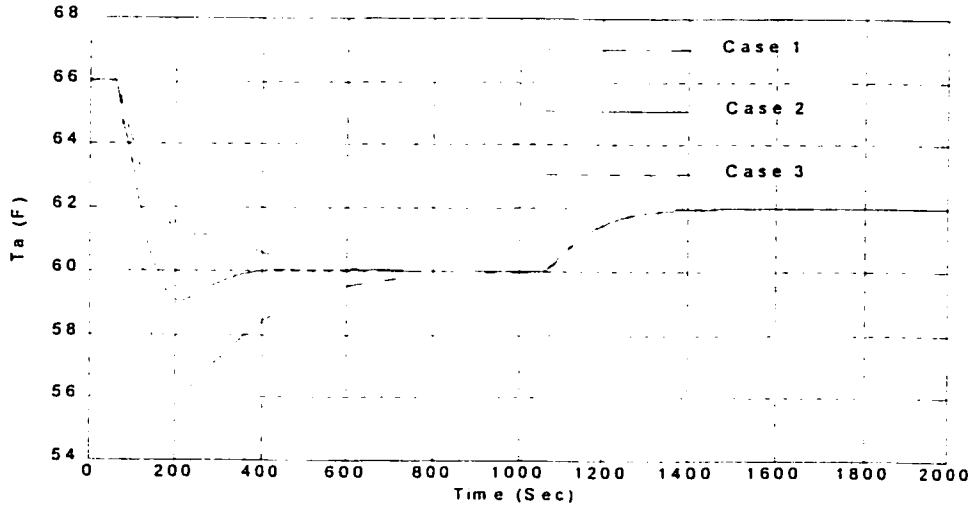

Figure 4.26 **H_∞ adaptive PI control responses to a setpoint change in different cases**

4.2.5 The Effect of Noise and Disturbance

In order to study the performance of H_∞ adaptive PI controller subject to disturbances and noise, several simulations runs were made. Figure 4.27 shows the implementation scheme.

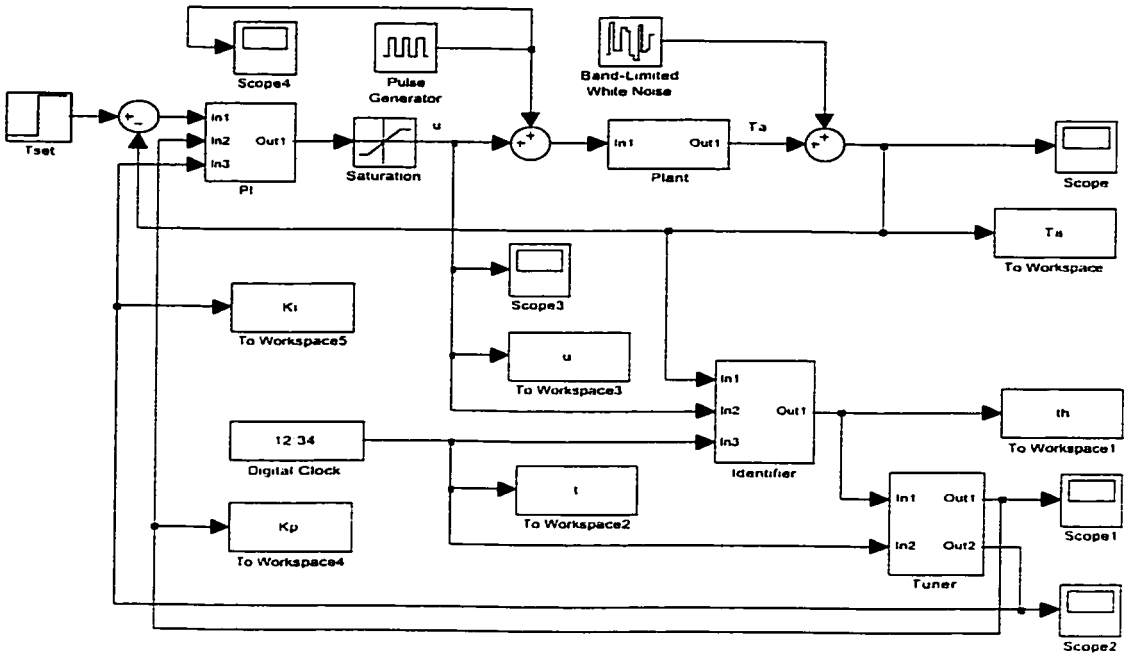

Figure 4.27 Implementation scheme for testing the effects of disturbances and noise

Using $Tset = \begin{cases} 60^\circ F & 0 \le t < 1000\,\text{sec} \\ 61^\circ F & t \ge 1000\,\text{sec} \end{cases}$, the output responses subjected to cyclic disturbances and band-limited white noise with different levels are presented in Figures 4.29 – 4.34; and the output response without disturbances and noise is shown in Figure 4.28. It can be noted that the H_∞ adaptive PI controller remains stable and is able to reject the effects of disturbances very well. From Figure 4.29, we see that the H_∞ adaptive PI controller has the ability to reject the effects of disturbances better than the pattern recognition adaptive controller (PRAC) does – comparing with the Figure 7 in reference [25].

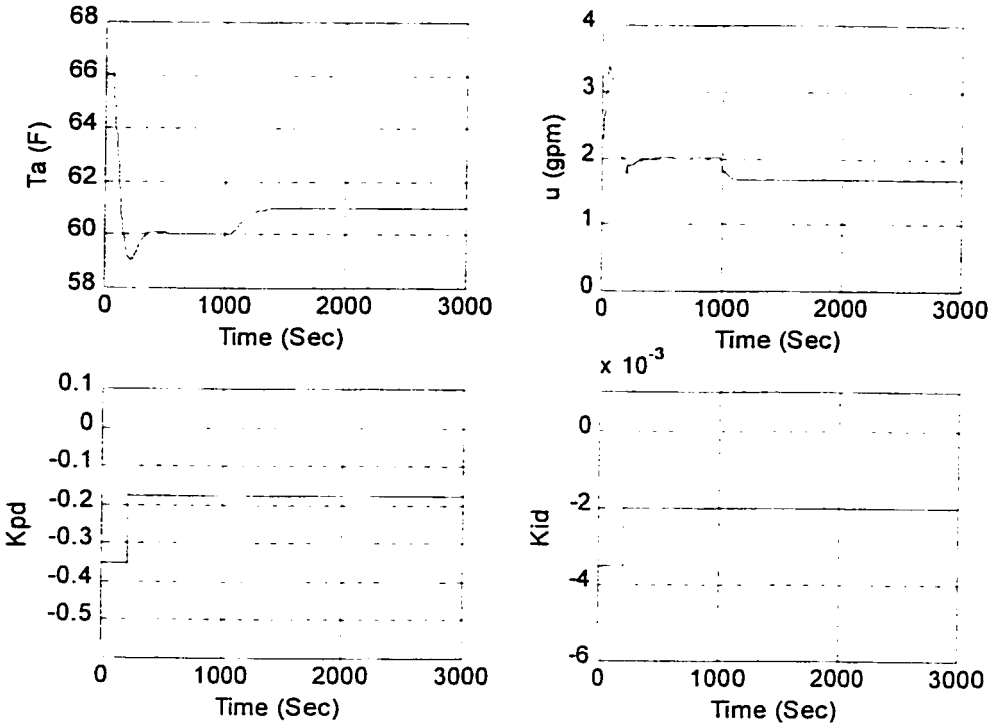

Figure 4.28 **Response without disturbances and noise**

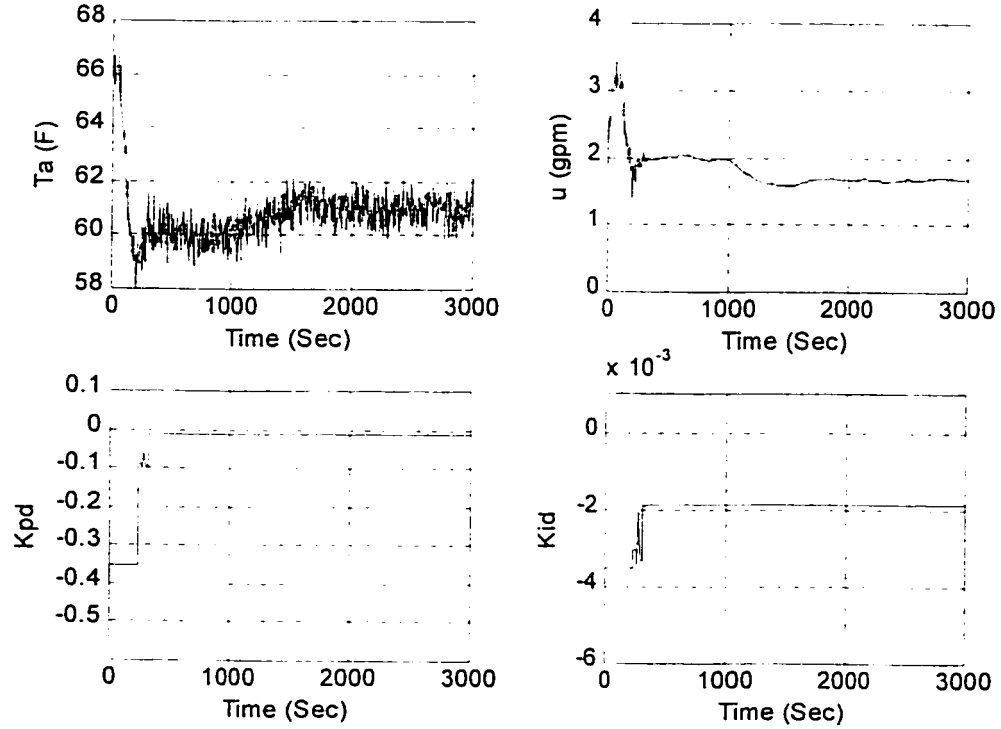

Figure 4.29 **Response under noise level between 0 and 1.0**

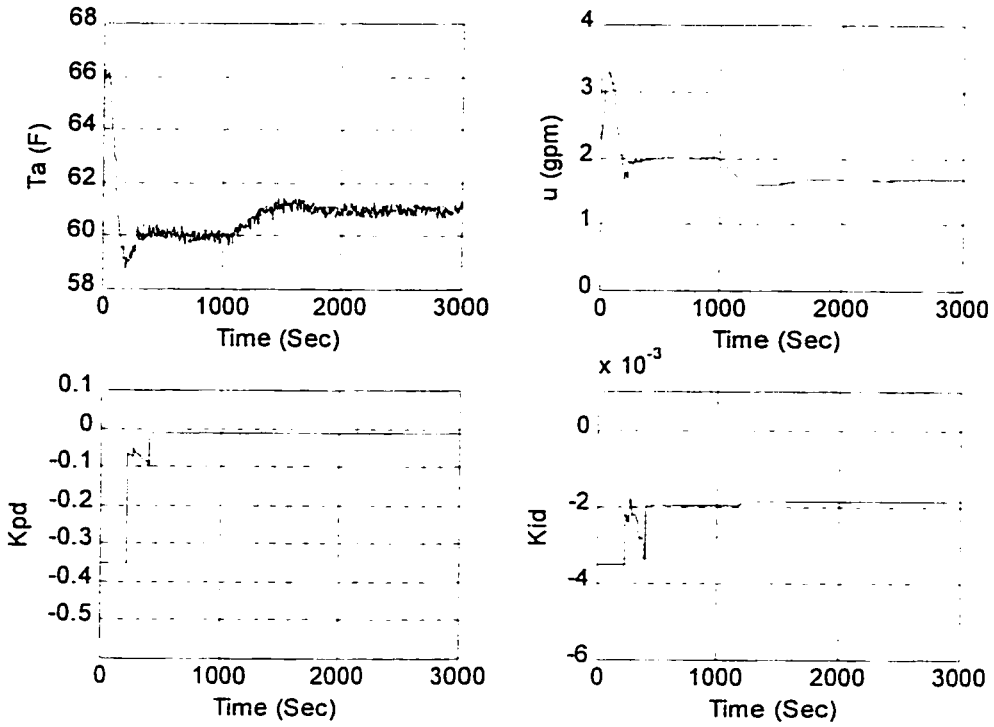

Figure 4.30 Response under noise level between 0 and 0.1

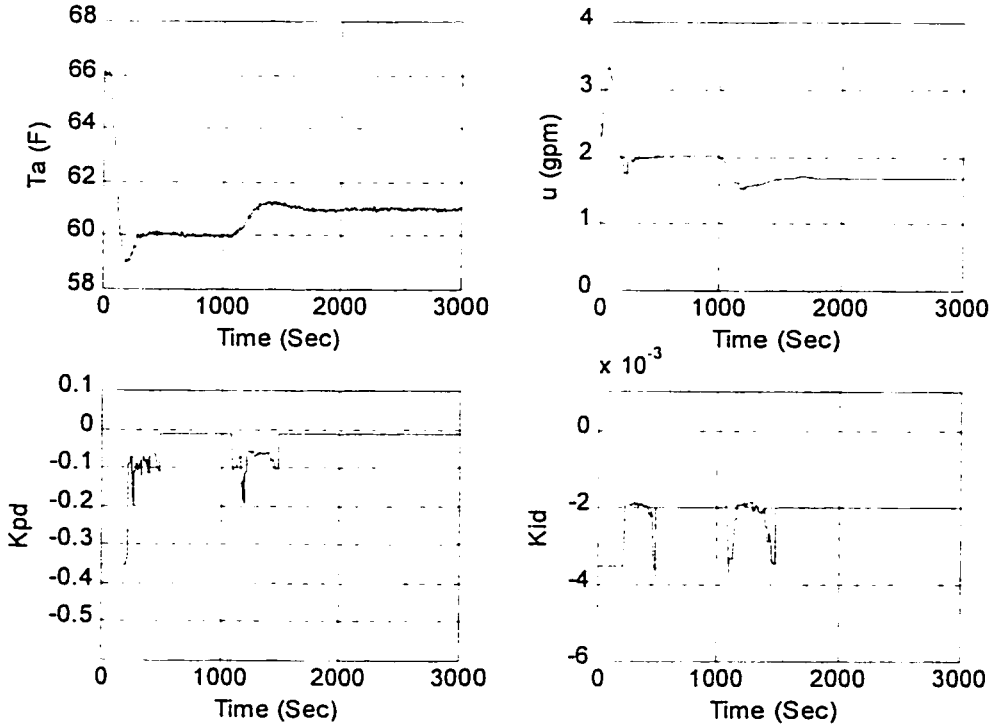

Figure 4.31 Response under noise level between 0 and 0.01

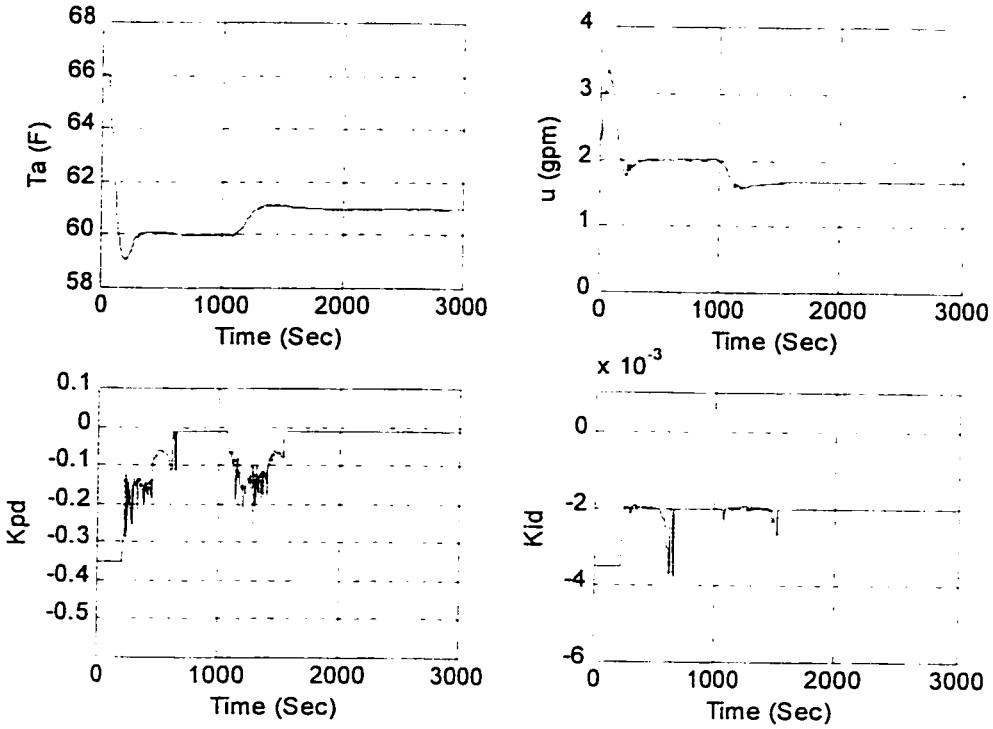

Figure 4.32 **Response under noise level between 0 and 0.001**

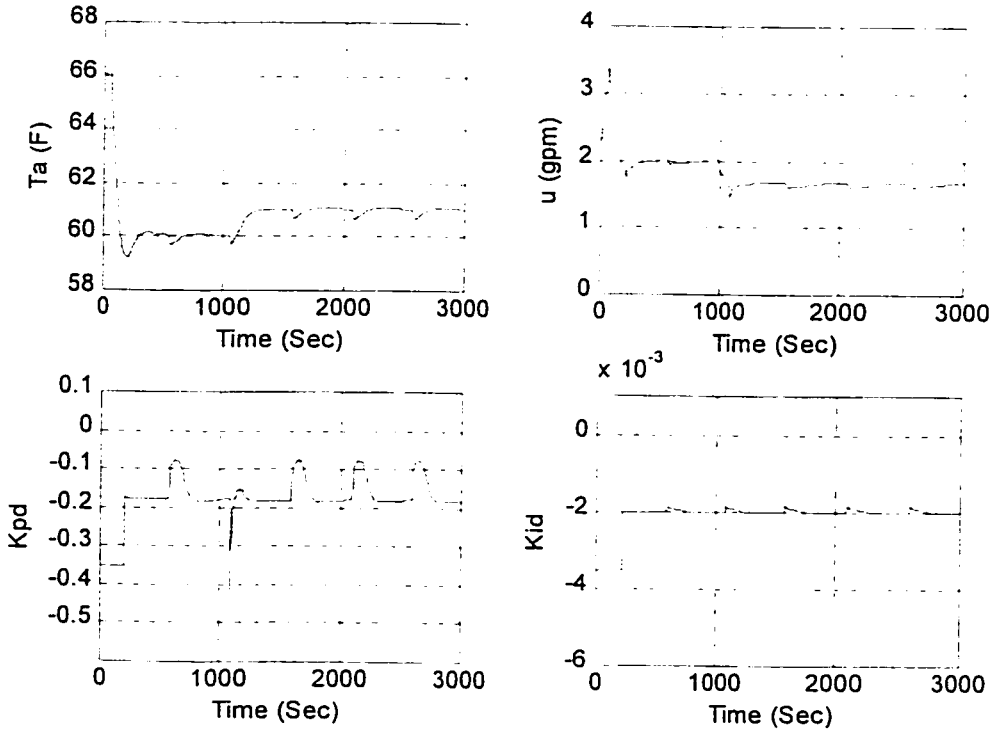

Figure 4.33 **Response with disturbance: 1 amplitude, 500 sec period and duty cycle 2%**

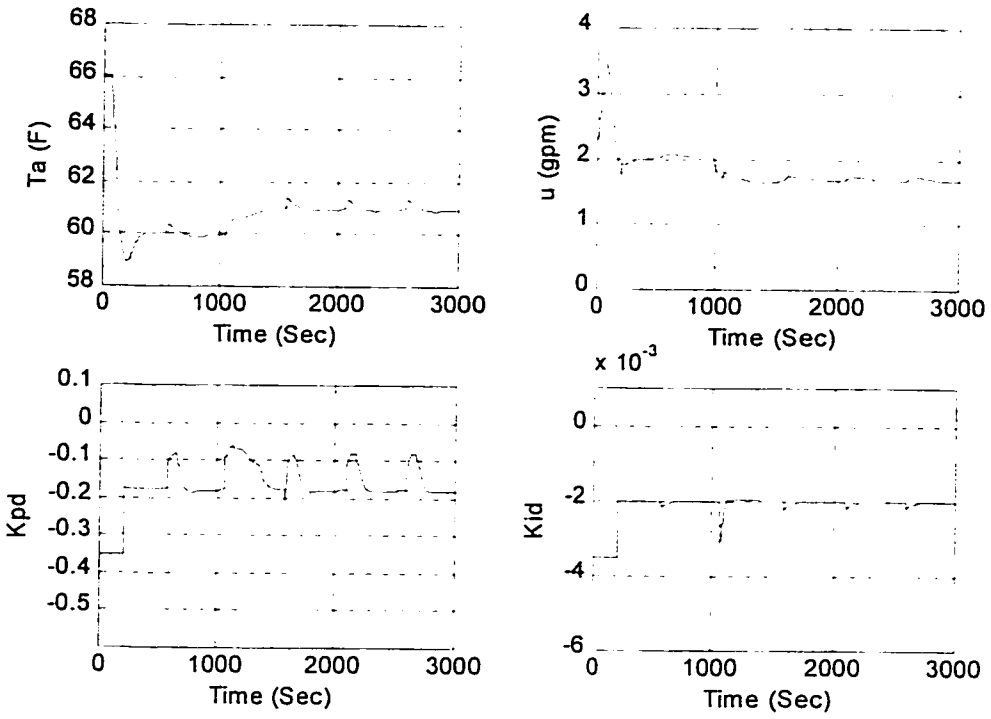

Figure 4.34 Response with disturbance: -1 amplitude, 500 sec period and duty cycle 2%

4.3 Summary

Three adaptive PI control strategies were developed for a DAT system. These are model following adaptive (MFA) PI control, control-model based adaptive (CMBA) PI control, and H_∞ adaptive PI control. The design methods for the three control strategies were also explained. In addition, simulation results for the three control strategies were shown and comparisons were made between them and with simulation results for constant PI control, LQR based adaptive PI control and Ziegler-Nichols based adaptive PI control. Finally, the performance of the H_∞ adaptive PI controller subject to disturbances and noise was studied. The results show that the H_∞ adaptive PI controller has the ability to reject disturbances very well, and may perform better than the pattern

recognition adaptive controller (PRAC). The proposed CMBA PI control is the easiest strategy to implement among the proposed adaptive PI control strategies, and has good performance to setpoint changes and robustness to different parameters of FOPDT models. The H_∞ adaptive PI control is a self-tuning control strategy, which performs very well in tracking setpoint changes and rejection of disturbances and noise.

5. Experimental Results

5.1 Introduction

In the previous chapter, three real-time adaptive PI control methods were developed: the H_∞ adaptive PI control, the CMBA PI control and the MFA PI control. The CMBA PI control is easier to implement compared to the H_∞ adaptive PI control, which requires online identification. In this chapter, we focus on the implementation of CMBA and H_∞ adaptive control methods. First, the experimental test facility (the thermal environment and control laboratory) where the real time tests were conducted will be described. It will be shown that both the CMBA PI control and the H_∞ adaptive PI control can be used to control the discharge air temperature in HVAC systems to achieve acceptable performance and enhance the robustness properties of the controllers.

5.2 The VAV-HVAC Test Facility

A two-zone VAV-HVAC Test Facility was designed, installed and commissioned to test and develop improved control strategies for HVAC systems. The test facility is located in the Thermal Environment Control Lab of Concordia University. The design and commissioning details of the test facility are described in [13]. Here a brief description of the test facility and the functions of the control-loops useful in understanding the results presented in this thesis will be described.

5.2.1 Mechanical System

A schematic diagram of the two-zone VAV test facility is shown in Figure 5.1. It consists of two $10 \times 10 \times 10$ ft thermal simulation chambers. Each chamber has a diffuser and a return air grille. There are two VAV boxes for both of the chambers, thereby allowing the supply air volume to the chamber to be varied. Each VAV box has an electric heater inside, which is used for re-heating and humidity control. An automatically operated fan motor speed control is used to adjust total airflow rate in the system. Two airflow dampers are used to modulate the airflow into the rooms. Chilled water is supplied to the cooling coil from a 2-ton water-cooled chiller and a storage tank unit. These control functions are performed by the following six local control-loops.

1) Discharge air temperature control
2) Variable speed fan control
3) Airflow rate control to room-1 (VAV box-1 damper control)
4) Airflow rate control to room-2 (VAV box-2 damper control)
5) Reheat control for room-1
6) Reheat control for room-2

When the system is operating, supply air is cooled to a specific temperature in the cooling coil. The cooled air is then driven by the fan to VAV boxes where the air flow rate and humidity or even temperature can also be changed by modulating dampers and/or by reheating by electric heaters if necessary to meet the cooling load requirements of the zone.

The description of the sensor legends and the actuator and transducer (interface) legends in Figure 5.1 are shown in Table 5.1 and 5.2 separately.

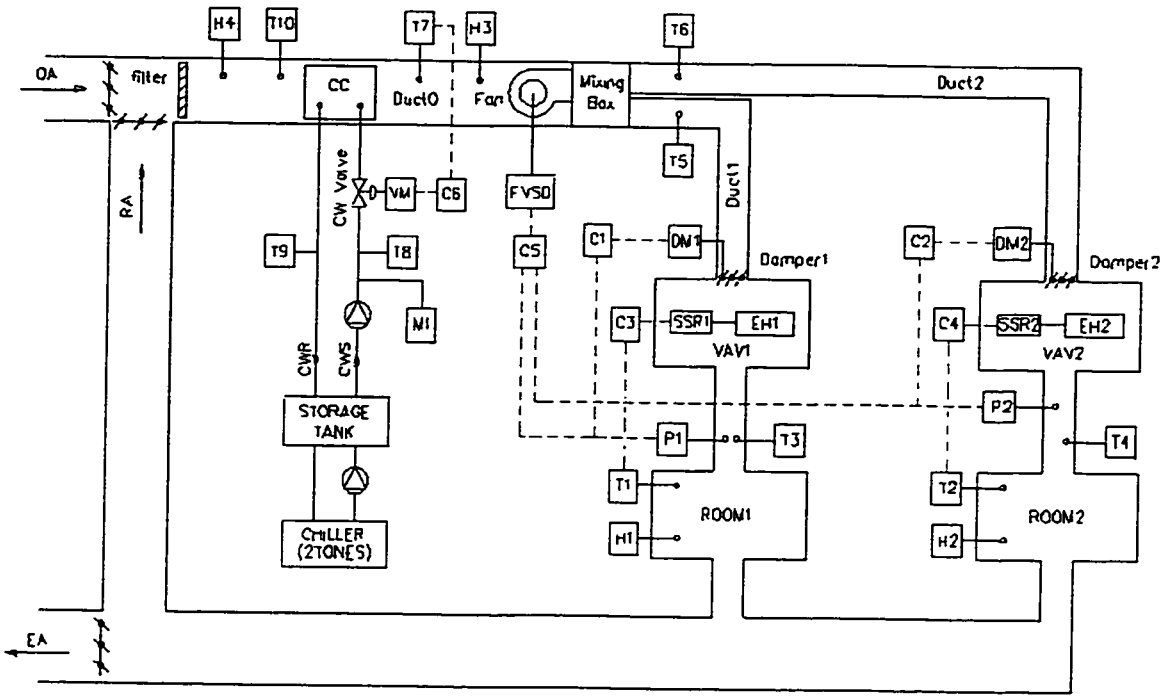

Figure 5.1 **Schematic diagram of the VAV-HVAC system and controls**

Number	Sensor symbol	Variable measured	Units
1	T_1	Room1 air temperature	°C
2	T_2	Room2 air temperature	°C
3	T_3	Supply air temperature to room1	°C
4	T_4	Supply air temperature to room2	°C
5	T_5	Temperature of air entering the VAV box-1	°C
6	T_6	Temperature of air entering the VAV box-2	°C
7	T_7	Discharge air temperature	°C
8	T_8	Supply chilled water temperature	°C
9	T_9	Return chilled water temperature	°C
10	T_{10}	Air temperature entering the cooling coil	°C
11	H_1	Room1 relative humidity	%
12	H_2	Room2 relative humidity	%
13	H_3	Discharge air relative humidity	%
14	H_4	Relative humidity of air entering the coil	%
15	P_1	Pressure (flow station) of air entering room1	in WG
16	P_2	Pressure (flow station) of air entering room2	in WG
17	M_1	Mass flow rate of chilled water	gpm

Table 5.1 **Descriptions of the Sensor Legends**

Number	Symbol	Controlling Variable	Interface	Signal Range
1	DM1	Room1 air temperature	Not needed	0~10VDC
2	DM2	Room2 air temperature	Not needed	0~10VDC
3	SSR1	Supply air temperature to room1	Needed	0~10VDC
4	SSR2	Supply air temperature to room2	Needed	0~10VDC
5	FVSD	Duct2 Pressure	Needed	0~2.8VDC
6	VM	Discharge air temperature	Not needed	0~10VDC

Table 5.2 Description of the Actuator and Transducer (Interface) Legends

5.2.2 Refrigeration System

The refrigeration system consists of a chiller and a storage tank. It can provide 2 tons of refrigeration and uses Freon-22 as refrigerant. Pressure in the suction line was set to 87 psig so that a water temperature of 6.5 °C can be obtained under design conditions. When the HVAC system operates under conditions different from design, there is no automatic control to keep water at a constant temperature so control is performed through a manual ON/OFF switch. Antifreeze solution was added to the water (1/3:1) to prevent corrosion and deterioration of the evaporator and the cooling coil.

The real time adaptive control algorithms developed in this thesis were implemented on the Discharge Air Temperature (DAT) control loop. In the following, a detailed description of the DAT control-loop is given.

5.2.3 The Discharge Air Temperature Control System

In the DAT system (see Figure 5.2), outdoor air (OA) and room return air (RA) are mixed and filtered before entering the cooling coil. Air entering the cooling coil at temperature (CCITS) is cooled to the discharge air temperature (CCOTS), which should track the desired discharge air temperature (VALSP), by modulating the flow rate of

chilled water (CWFRG) by a motorized three-way valve controlled by the controller output (VALVE in voltage, VALOP in percentage). The airflow rate in the system is measured by measuring the differential pressure (DPS), which is converted to airflow rate in cubic feet per minute (CFM). Also shown in the figure are the sensors measuring the temperature of water entering (CWSTS) and leaving (CWRTS) the cooling coil.

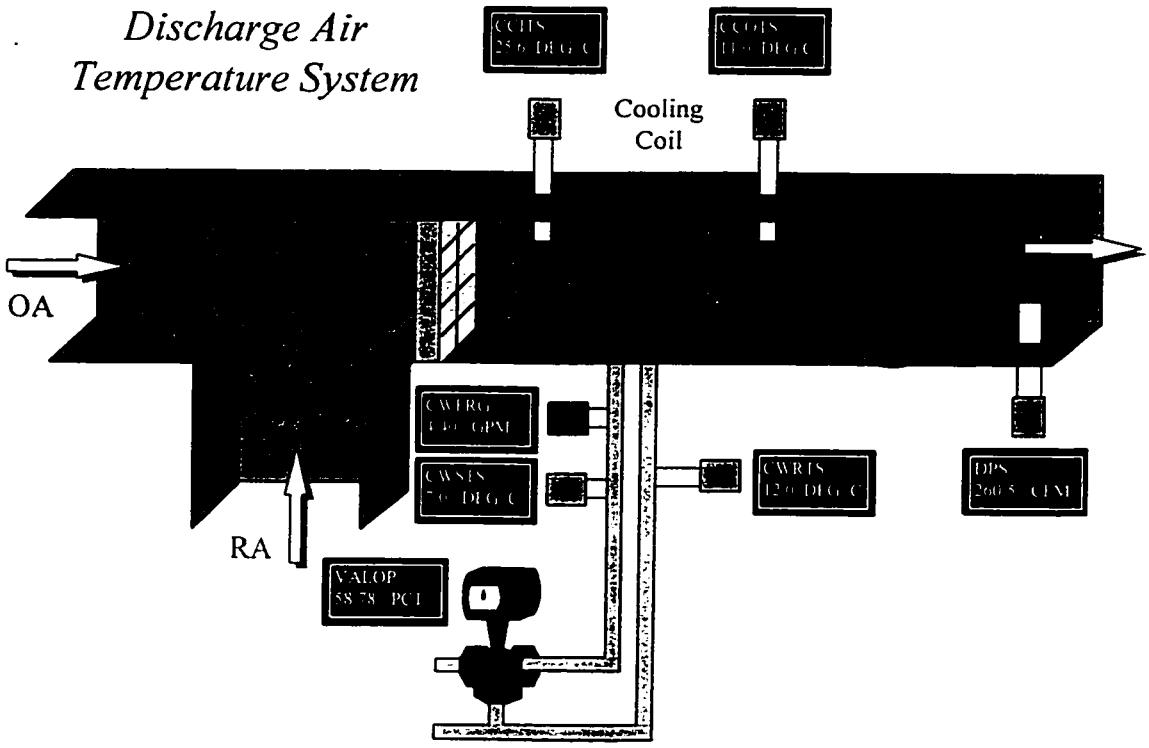

Figure 5.2 The Discharge air temperature (DAT) control system

5.2.4 Energy Management Control System (EMCS)

The monitoring, control and data acquisition functions are handled by the Energy Management Control System (EMCS). The overall system consists of one PC, one Modular Building Controller (MBC), several sensing devices, interfaces, and actuators such as damper motors, fan variable speed drive, valve motor, etc.

The Modular Building Controller (MBC) is an integral part of the EMCS [13]. It is a high performance, modular Direct Digital Control (DDC) supervisory field panel. The field panel operates stand-alone or networked to perform complex control, monitoring and energy management functions without relying on a higher-level processor. Combined with Powers Process Control Language (PPCL) [46], MBC plus a PC where APOGEE Insight software [46] was installed can implement many powerful functions such as:

- Custom program sequences to match equipment control applications
- Data acquisition and processing
- Dynamic graphics
- Advanced Proportional Integral Derivative (PID) loop tuning algorithms for HVAC Built-in energy management applications and DDC programs for complete facility management
- Optimizing the use of the equipment
- Energy Management and Control
- Comprehensive alarm management, historical data trend collection, operator control and monitoring function

The HVAC system is a multi-loop control system, and the discharge air temperature control is the most difficult part in it, because many disturbances affect the control performance in real time. For example, we can change the damper position and the fan speed as needed. By doing so we can create various operating conditions, and/or disturbances to test the performance of the CMBA control loop or the H_∞ PI adaptive control loop. For instance, entering air temperature (CCITS), supply water temperature (CWSTS) and airflow rates (DPS) act as disturbances on the CMBA control loop. Several

experiments were conducted in which setpoint changes and multiple disturbances were simultaneously acting on the CMBA control loop and the H_∞ PI adaptive control loop. Note that the EMCS needs a scaling factor of 1000.0 for the implementation of PID parameters. The results are presented in the following sections.

5.3 CMBA PI Control System

In this section, the implementation of the CMBA PI control as an application of EMCS is presented, and then the experimental results for different environmental conditions are shown. The experimental results indicate that the CMBA PI control has acceptable performance for the discharge air temperature control in HVAC systems. The comparison with constant PI control is also presented.

5.3.1 Interface Modules for the CMBA PI Control System

To monitor the status of the CMBA PI control loop and update various parameters in real time, graphics modules were developed. Accessing each module or switching back and forth between them was achieved by adding an interface icon to each module. The CMBA PI control system includes four interface modules: CMBA Control Loop, Control Model, Parameter Update and PID shown in Figure 5.3 – 5.6.

The CMBA Control Loop (Figure 5.3) is designed as the main interface to show the control system structure and display the real values of the setpoint – the desired discharge air temperature (VALSP) and the actual discharge air temperature (CCOTS). We can also click on any one of the blocks with an arrow to go to the related sub-interface module.

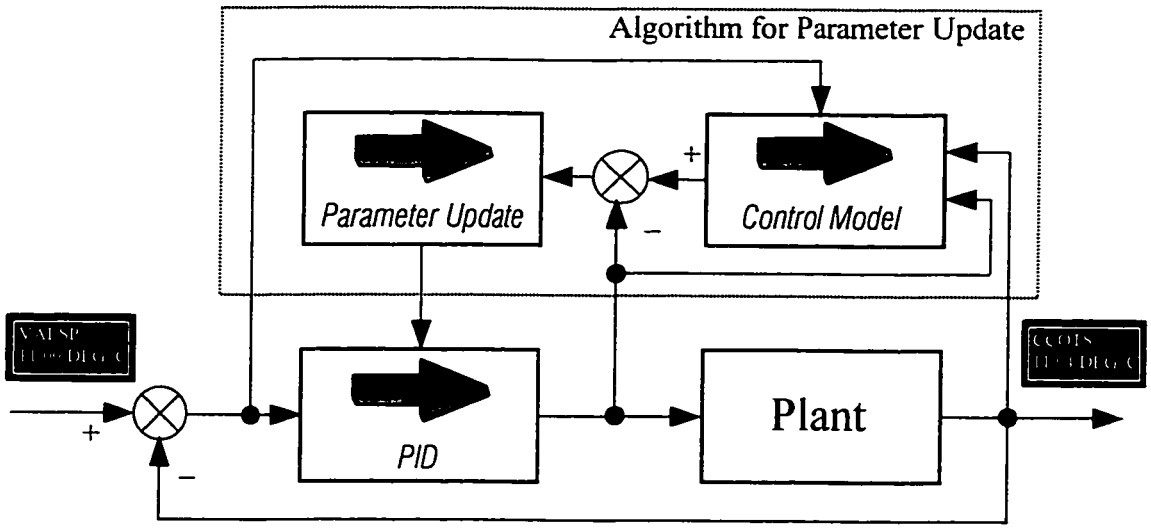

Figure 5.3 Interface of the CMBA Control Loop

The Control Model interface module (Figure 5.4) is designed to set the parameters before or during the operation of the system, and display the related inputs and outputs. A click on any parameter icon opens a dialog interface. In the dialog interface, we can change the parameter value.

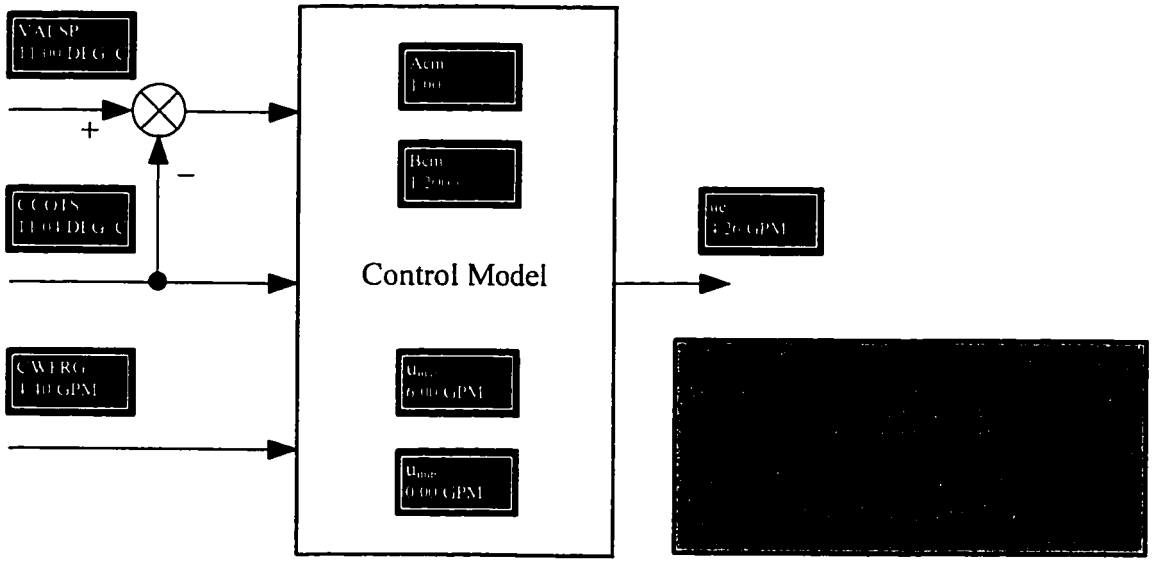

Figure 5.4 Interface module of computing and displaying control model output

The interface module of Parameter Update (Figure 5.5) is designed to set the parameters used in PI parameter computing, and display the computed PI values and related inputs. In Figure 5.5, Zeta is the ς in section 4.1.2, and VALKPE and VALKIE are the computed PI parameters used in the CMBA control method.

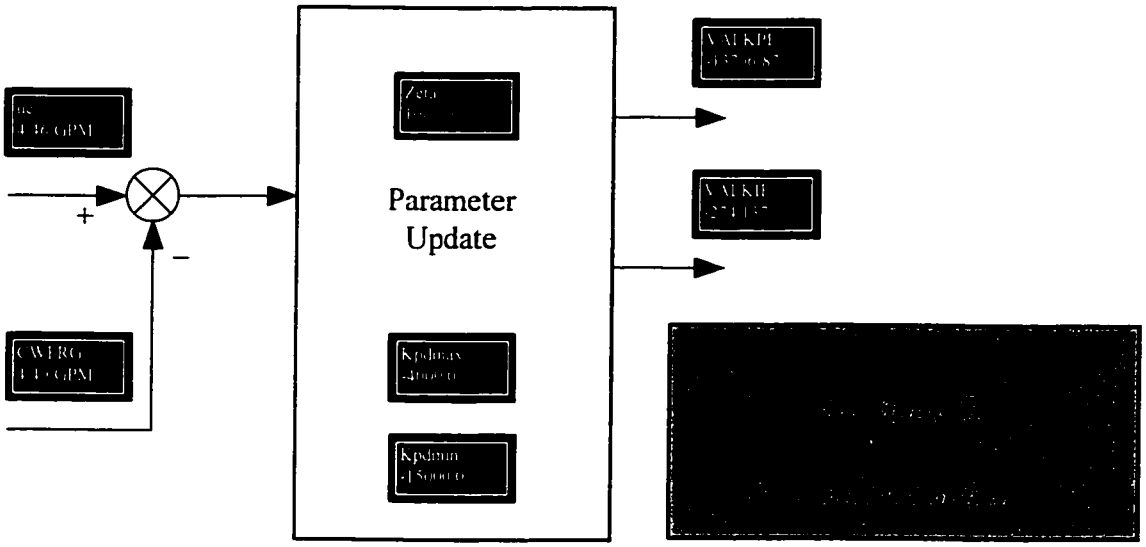

Figure 5.5 Interface module of computing and updating PI parameters

The interface module for PID (Figure 5.6) is designed to show the updated PI parameters and display the input and output of the PID controller. In Figure 5.6, VALE is the difference between VALSP and CCOTS. VALKP, VALPI and VALPD are the parameters of the PID controller in the discharge air temperature control loop. Each of the three sub-interfaces has a block that can be click to go back to the interface of CMBA Control.

We have used the EMCS for the two applications – the CMBA PI control and the H_∞ adaptive PI control, and taken 4 seconds as the sampling time. The performance of setpoint changes and the effects of the environmental condition changes were examined. The results were plotted using the plot function of the EMCS.

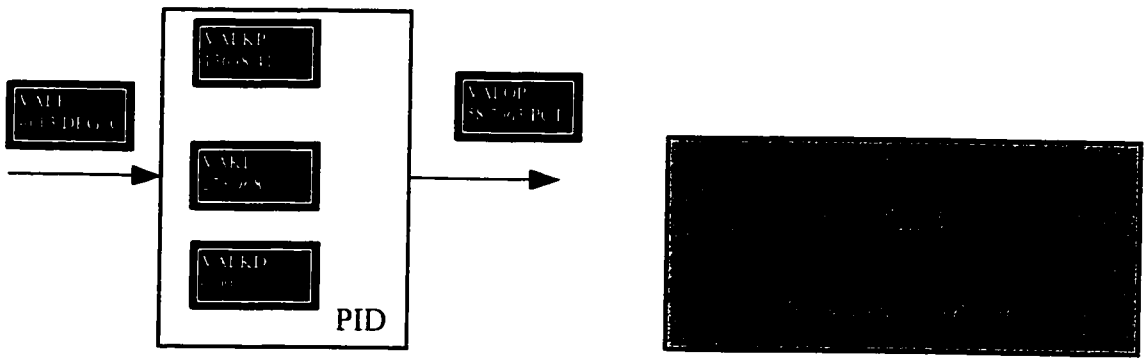

Figure 5.6 **Interface module for displaying updated PID parameters**

5.3.2 Results of Step Changes in Setpoint

In this sub-section, experimental results showing the CMBA PI control performance subject to setpoint changes are presented. A setpoint change means a step change in the desired discharge air temperature. Figures 5.7 – 5.8 and Figures 5.11 – 5.12 show the CMBA PI control responses for four different operating conditions and control parameters referred to as case 1 through case 4. The operating conditions corresponding to each of these cases are summarised in Table 5.3.

The operating conditions listed in Table 5.3 correspond to about 65% of the full-load conditions on the DAT system. Under these conditions the DAT setpoint was either decreased by 2 °C (high to low) or increased by 2 °C (low to high). Such setpoint changes are typical in response to an increase in cooling load or a decrease in the cooling load acting on the system.

From Figure 5.7 it can be seen that the CMBA control tracks the setpoint and reaches steady state (11 °C) in about 7 minutes. When the setpoint is increased (11 to 13 °C), the CMBA control also makes the DAT reach the new setpoint (Figure 5.8). However, in doing so, it gives rise to some overshoot (+0.9 °C) and takes a little longer

(20 minutes) to reach steady state. These results are summarised in Figures 5.9 and 5.10 in which the discharge air temperature (CCOTS) response is plotted along with chilled water flow rate (CWFRG) response. The results show that the actuator response is smooth. Also shown in Figure 5.10 is the evolution of PI gains during the tests. It is apparent that the CMBA control continuously updates the PI gains in response to changes in setpoints and operating conditions in order to give smooth output responses. As shown in Figure 5.10 the value gains changed between 4400 – 7200.

Test case	Date	Setpoint-change and Time	Operating conditions	Control parameters	Figure
1	May 31, 02	13 to 11 °C at 10:47:39	CFM: 250 ± 5 Inlet air temp. : 24 ± 0.25 °C Inlet water temp. : 7 ± 0.5 °C	$A_{cm} = 1$ $B_{cm} = 0.8$ $\varsigma = 100$ $K_{pd\,max} = -4000$ $K_{pd\,min} = -10000$	5.7
2	May 31, 02	11 to 13 °C at 11:00:14	Same as in case 1	Same as in case 1	5.8
3	May 31, 02	11 to 13 °C at 16:30:57	CFM: 250 ± 8 Inlet air temp. : 25 ± 1.0 °C Inlet water temp. : 7 ± 0.8 °C	$A_{cm} = 1$ $B_{cm} = 0.5$ $\varsigma = 100$ $K_{pd\,max} = -4000$ $K_{pd\,min} = -10000$	5.11
4	May 31, 02	16 to 11 °C at 17:13:04	Same as in case 3	Same as in case 3	5.12
5	May 30, 02	a) 13 to 11 °C at 13:57:54 b) 11 to 13 °C at 14:25:41	CFM: 260 ± 8 Inlet air temp. : 25 ± 1.0 °C Inlet water temp. : 8 ± 0.8 °C	$A_{cm} = 1$ $B_{cm} = 0.2$ $\varsigma = 360$ $K_{pd\,max} = -4000$ $K_{pd\,min} = -10000$	5.13

Table 5.3 CMBA PI Control: Setpoints and operating conditions

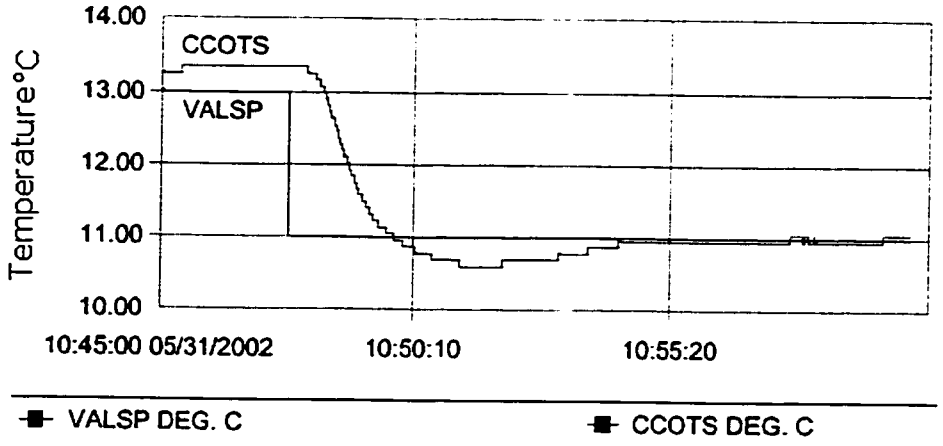

Figure 5.7 **Experimental result for the CMBA PI control for case 1**

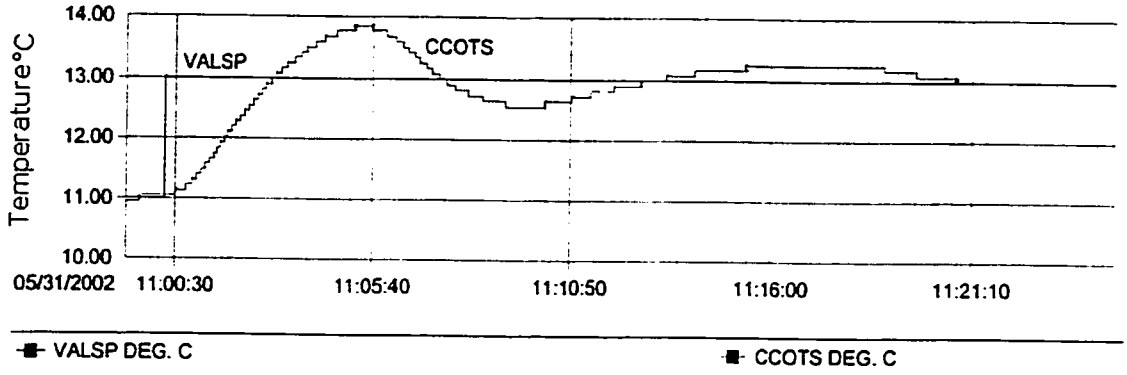

Figure 5.8 **Experimental result for the CMBA PI control for case 2**

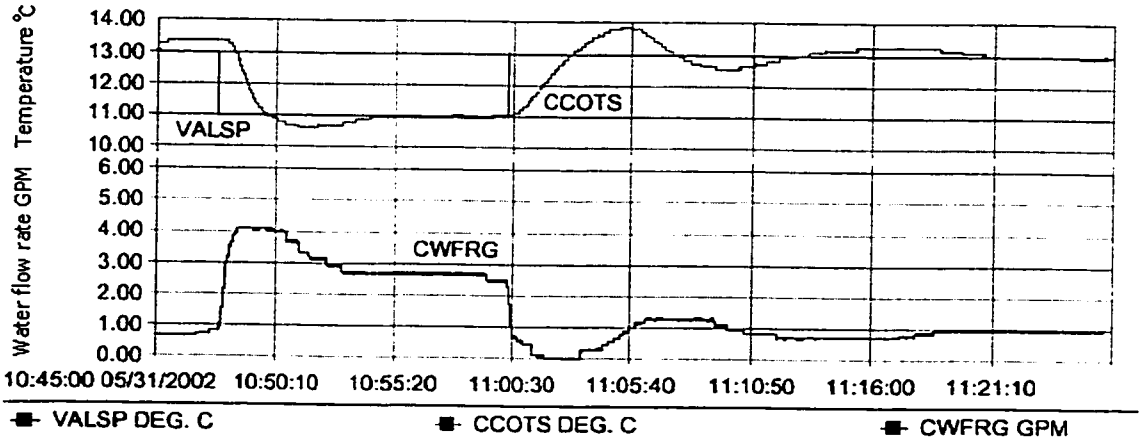

Figure 5.9 **Chilled water flow rate changes in the CMBA PI control**

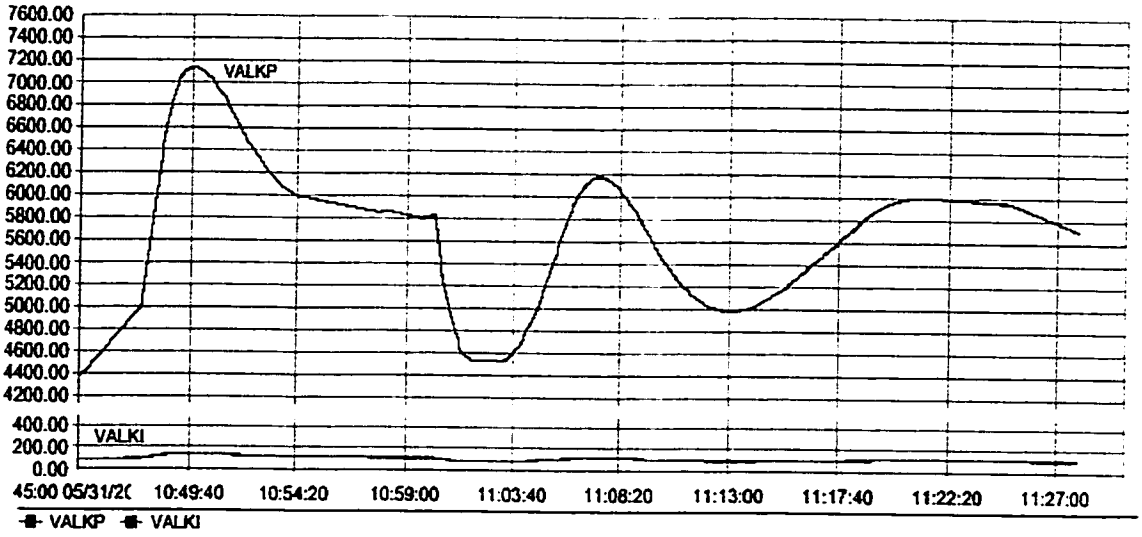

Figure 5.10 PI parameter changes in the CMBA PI control

An important parameter in optimizing the output response of CMBA PI control is B_{cm}. To examine its impact on the CMBA PI control performance, further tests (test cases 3 and 4) were conducted. The only difference between Test cases 1, 2 and Test cases 3, 4 is the magnitude of the parameter B_{cm} which was decreased from 0.8 to 0.5 as noted in Table 5.3. It is apparent from Figure 5.11 that the DAT response following an increase in the setpoint (from 11 to 13 °C) is smooth and fast. Compared to the results shown in Figure 5.8 with $B_{cm} = 0.8$, the DAT response in Figure 5.11 ($B_{cm} = 0.5$) is smooth and reaches steady state (about 8 minutes) a lot faster. Also shown in Figure 5.12 is the DAT response with $B_{cm} = 0.5$ for a 5 °C decrease in the DAT setpoint. In spite of the severity in the setpoint change, the CMBA PI control remains stable and reaches steady state without any oscillations.

The setpoint tests (Test case 5) were repeated to show further (Figure 5.13) that with $B_{cm} = 0.2$ and $\varsigma = 360$, the output responses remained smooth and stable. Therefore, it appears that the optimum range of parameter B_{cm} for good response is

between 0.2 to 0.5. The magnitude of ς is not very critical. For good control, the operating range of ς was found to be between 100 to 360.

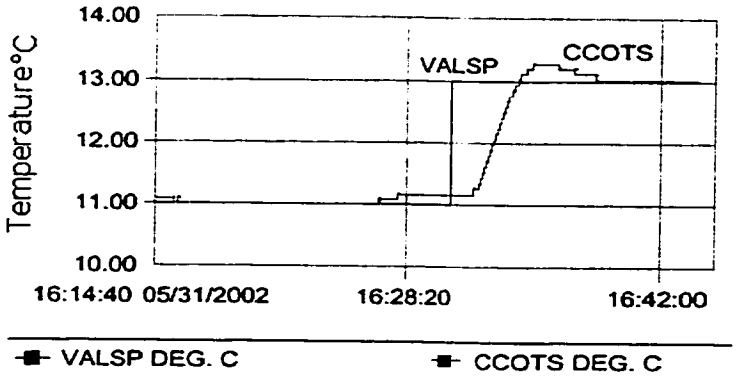

Figure 5.11 Experimental result for the CMBA PI control for case 3

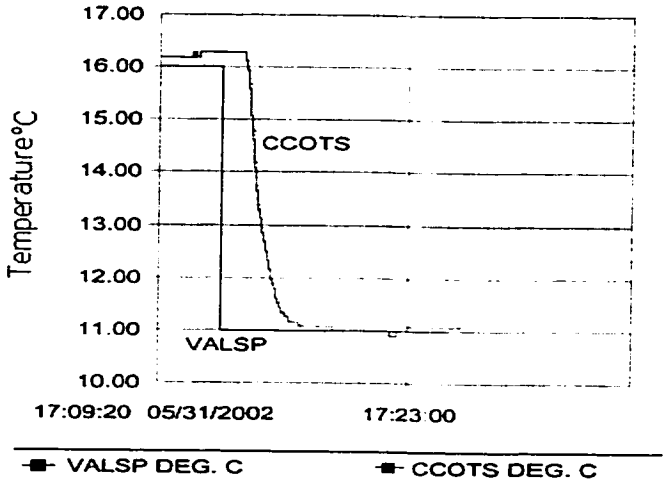

Figure 5.12 Experimental result for the CMBA PI control for case 4

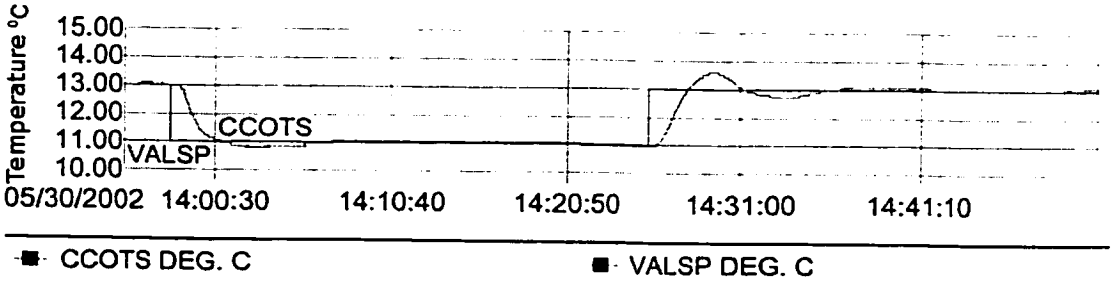

Figure 5.13 Experimental result for the CMBA PI control for case 5

5.3.3 Effects of Disturbances

In this section, the results of the CMBA PI control performance subject to multiple disturbance effects are presented. The disturbances include changes in entering air temperature (CCITS), chilled water supply temperature (CWSTS), and airflow rate (CFM). Five different tests (identified as Test cases 6 through 10) were conducted. The details of the test cases are summarised in Table 5.4.

The three disturbances (CCITS, CWSTS and CFM) identified above affect the DAT in several different ways. The impact of each of these disturbances was examined in Test cases 6 – 8. The results are shown in Figures 5.14 to 5.19. In Test case 6, the effect of increasing the inlet air temperature from 23.9 to 27 °C over a period of 15 minutes (0.2 °C /min.) is shown in Figures 5.14 and 5.15. It is apparent that the DAT system (identified as CCOTS in Figure 5.14) deviates about 0.2 °C from the setpoint. In other words, the CMBA PI control is able to reject the effects of a change in CCITS of 3 °C (which is typical in buildings) without causing any significant overshoot.

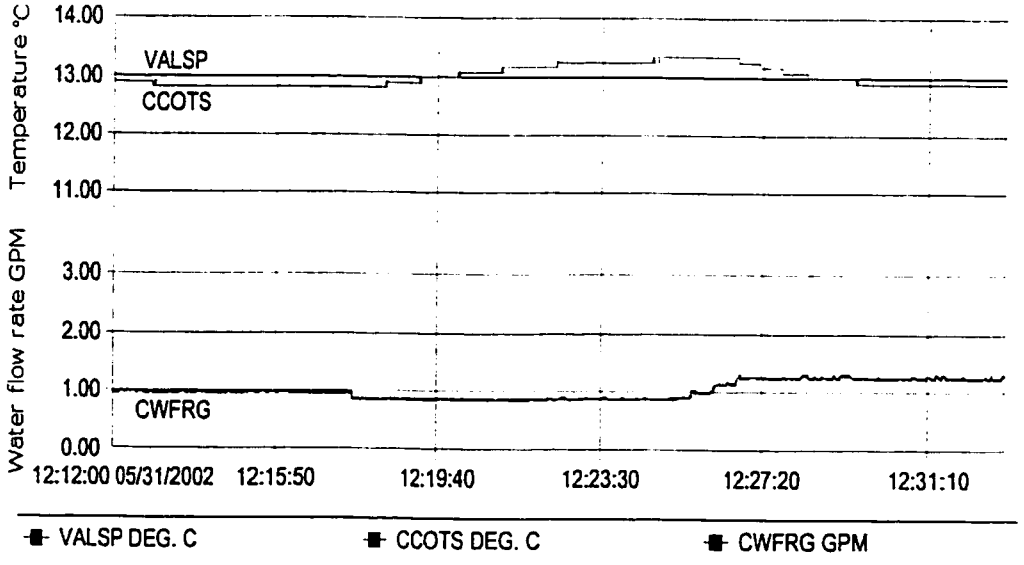

Figure 5.14 Effect of the increase in entering air temperature on the CMBA PI control

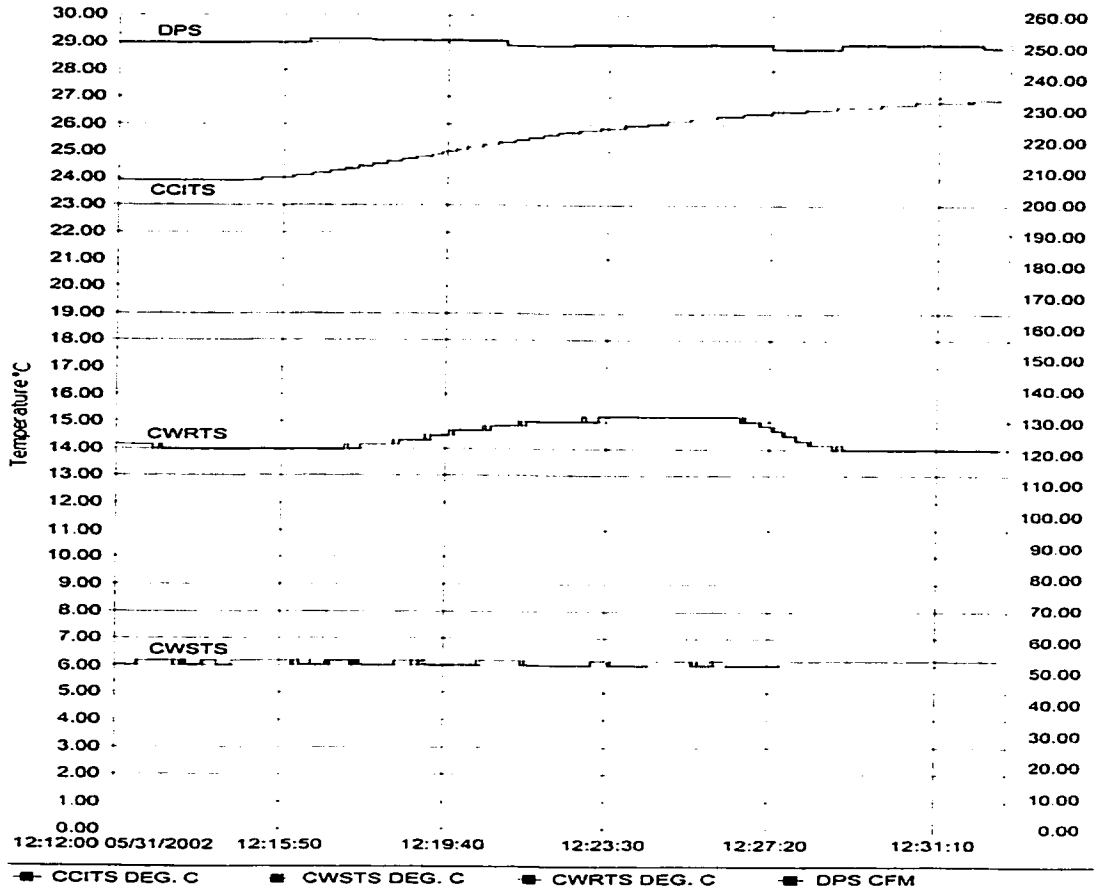

Figure 5.15 Increase in entering air temperature for the CMBA PI control

The CMBA PI control response subject to a change in inlet water temperature (CWSTS) by 2 °C (Test case 7) is shown in Figures 5.16 and 5.17. Again the output response is stable and remains close to the setpoint.

The Test case 8 results in which the airflow rate was changed from 145 to 250 cfm and from 252 to 142 cfm in a step-like manner are shown in Figures 5.18 and 5.19. As shown in Figure 5.18, the CCOTS overshoots by 0.7 °C following an increase in CFM and drops by 0.5 °C as the CFM is decreased. Of the three disturbances, the effect of changes in airflow rate seems to produce somewhat noticeable overshoot. Otherwise, the CMBA PI control is able to reject the disturbances resulting in stable and smooth output response.

Test case	Date	Time	Operating conditions	Control parameters	Figure
6	May 31, 02	12:14:00 to 12:32:00	Inlet air temp. : 23.9 to 27 °C CFM: 250 ± 5 Inlet water temp. : 7 ± 0.5 °C Setpoint: 13 °C	$A_{cm} = 1$ $B_{cm} = 0.5$ $\varsigma = 100$ $K_{pd\,max} = -4000$ $K_{pd\,min} = -10000$	5.14 5.15
7	May 31, 02	12:31:10 to 12:48:59	Inlet air temp. : 26.8 to 27.7 °C Inlet water temp. : 6 to 8 °C CFM: 250 ± 5 Setpoint: 13 °C	Same as in case 6	5.16 5.17
8	May 31, 02	15:20:00 to 16:01:00 ① 15:28:16 ② 15:41:33	Inlet water temp. : 9 to 7.3 °C Inlet air temp. : 25 ± 0.5 °C Setpoint: 13 °C ① CFM: 145 to 250 ② CFM: 252 to 142	Same as in case 6	5.18 5.19
9	May 31, 02	15:55:36 to 17:28:07 i) 15:55:36 to ii) 16:31:33 to iii) 17:02:30 to 17:28:07 a) 16:00:16 b) 16:30:57 c) 16:43:58 d) 16:59:36 e) 17:13:04 ① 16:44:12 ② 16:59:40	Inlet water temp. : 7 ± 0.8 °C Inlet air temp. : i) 25.4 to 24 °C ii) 24 to 27.1 °C iii) 27.1 to 24.4 °C Setpoint: a) 13 to 11 °C b) 11 to 13 °C c) 13 to 11 °C d) 11 to 16 °C e) 16 to 11 °C CFM: ① 252 to 143 ② 140 to 253	Same as in case 6	5.20 5.21
10	June 5, 02	13:55:00 to 14:57:22 ① 13:55:00 to ② 13:56:50 to ③ 14:16:35 to ④ 14:31:45 to ⑤ 14:43:44 to 14:57:22 a) 13:57:54	Inlet air temp. : 22 to 26.7 °C Inlet water temp. : 7.3 ± 0.4 °C CFM: ① 65 to 43 ② 43 to 425 ③ 425 to 45 ④ 45 to 299 ⑤ 299 ± 4 Setpoint: a) 13 to 11 °C	$A_{cm} = 1$ $B_{cm} = 0.5$ $\varsigma = 100$ $K_{pd\,max} = -4000$ $K_{pd\,min} = -15000$	5.22 5.23 5.24

Table 5.4 CMBA PI Control: Disturbances and operating conditions

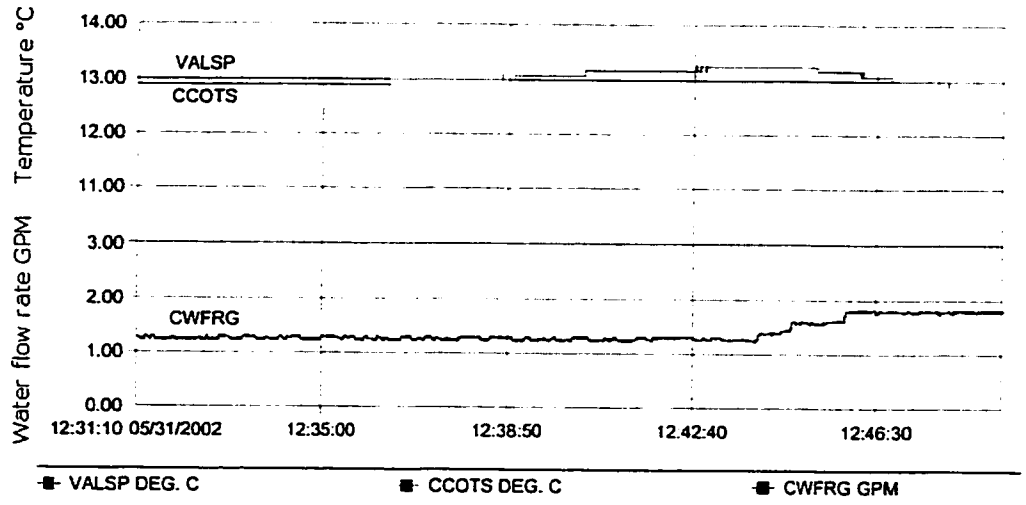

Figure 5.16 **Effect of the increase in chilled water supply temperature in the CMBA PI control**

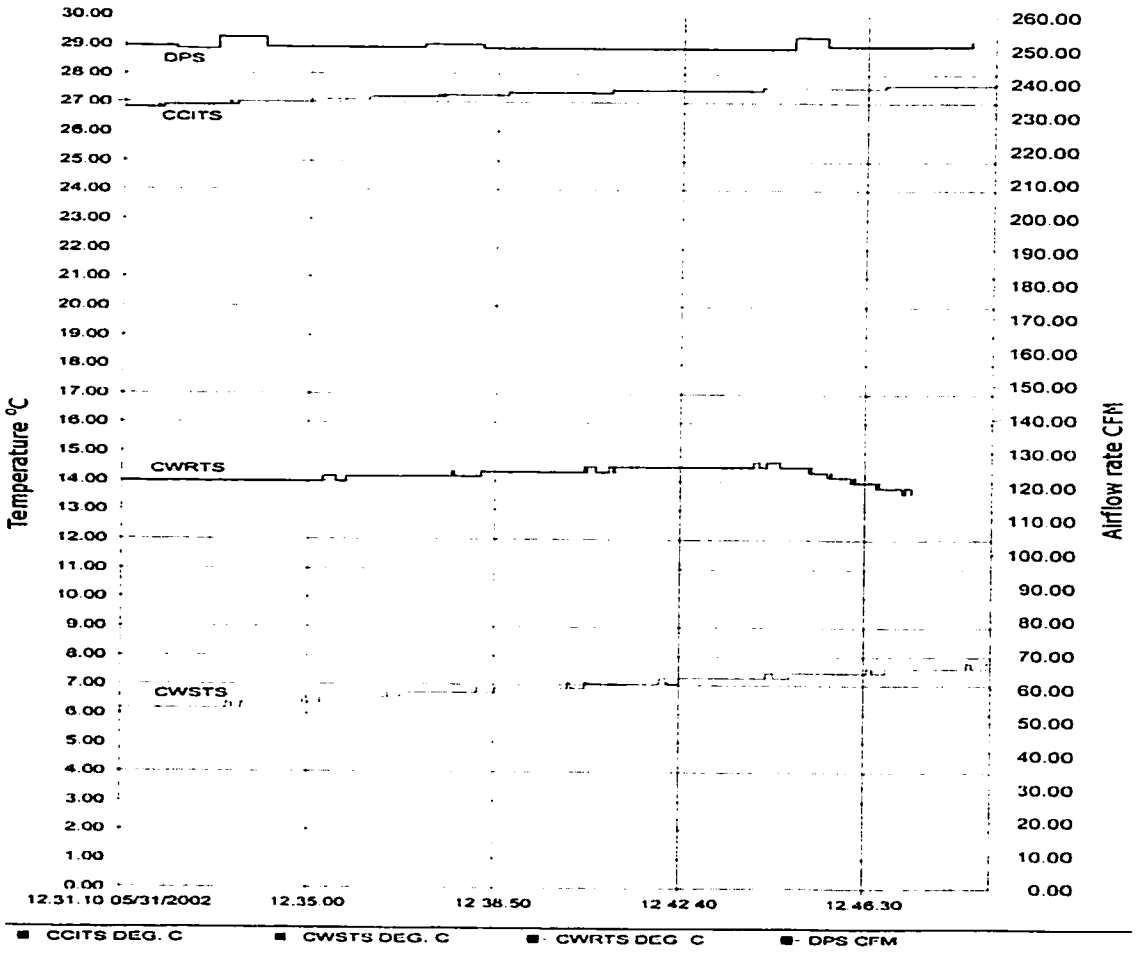

Figure 5.17 **Increase in chilled water supply temperature for the CMBA PI control**

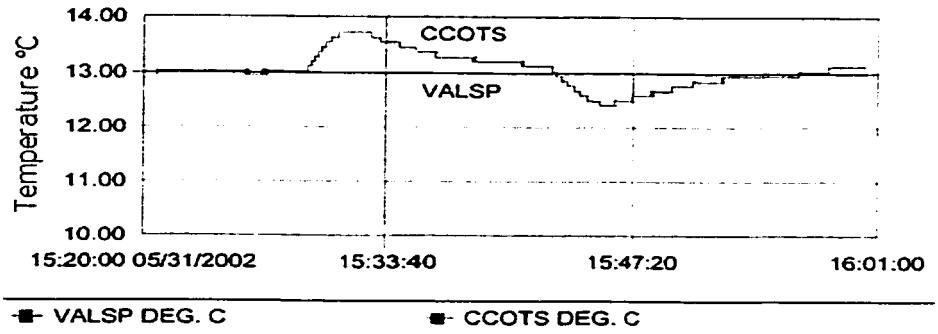

Figure 5.18 Effect of airflow rate changes on the CMBA PI control

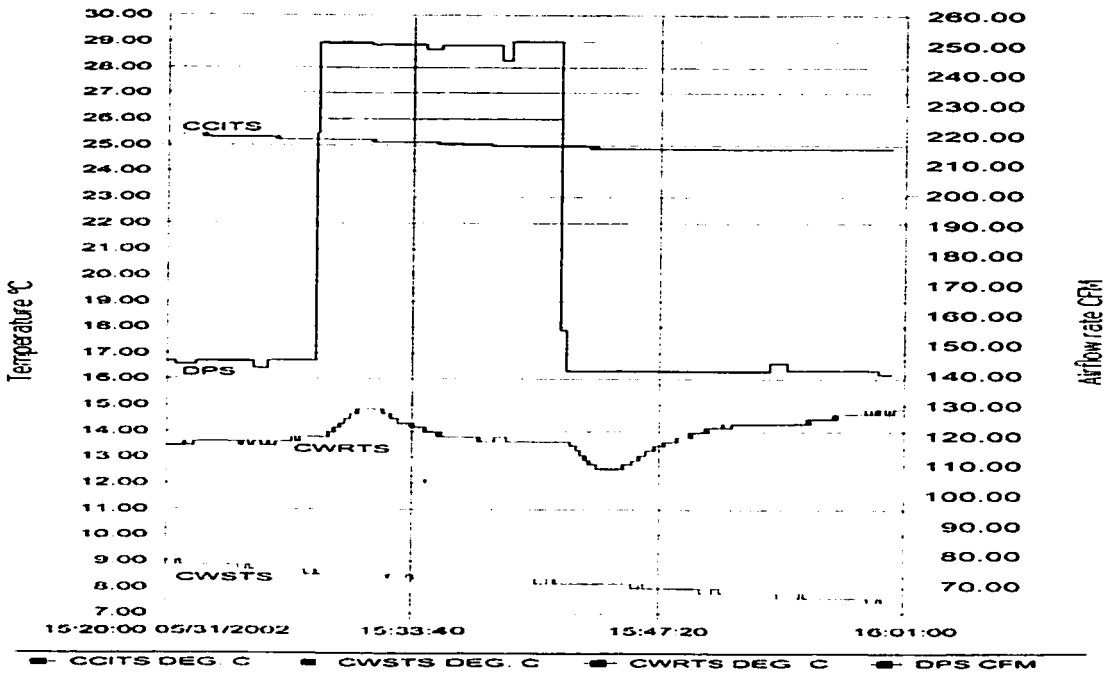

Figure 5.19 Airflow rate changes for the CMBA PI control

In Test case 9, the effect of all three disturbances acting simultaneously (multiple disturbances) together with setpoint changes were examined. The results are shown in Figures 5.20 and 5.21. It is worth noting that the CMBA PI control maintained smooth output response close to the setpoint. A setpoint change of 5 °C (from 11 to 16 °C) together with an increase in CFM from 140 to 250 and changes in CCITS and CWSTS as noted in Table 5.4 (Test case 9) and their effects were efficiently attenuated by the CMBA PI control.

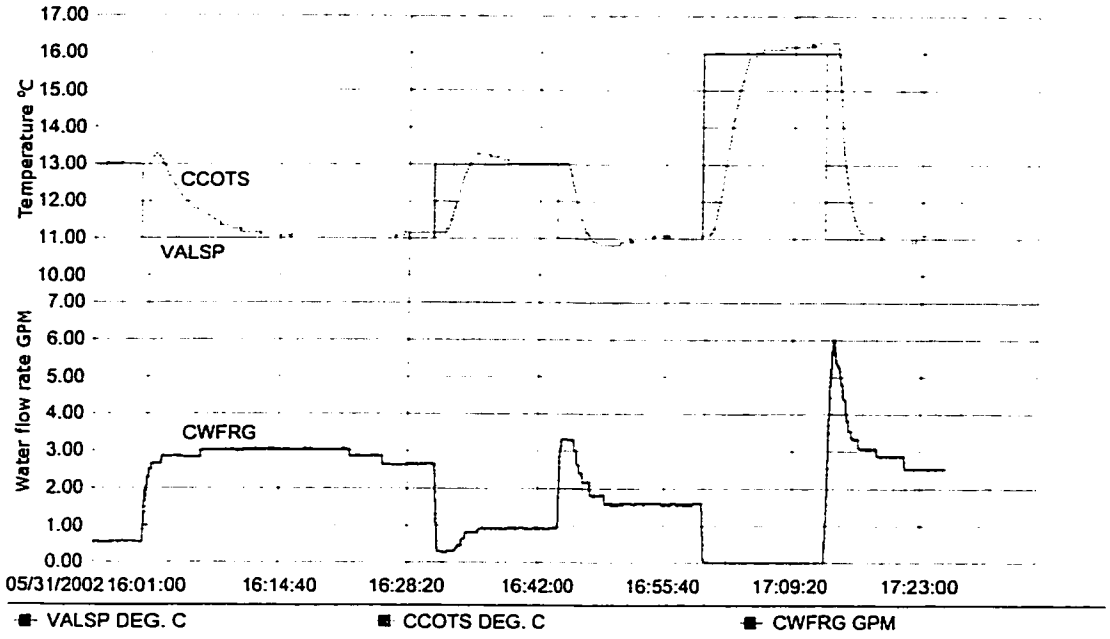

Figure 5.20 Tracking result under multi disturbance effects for the CMBA PI control

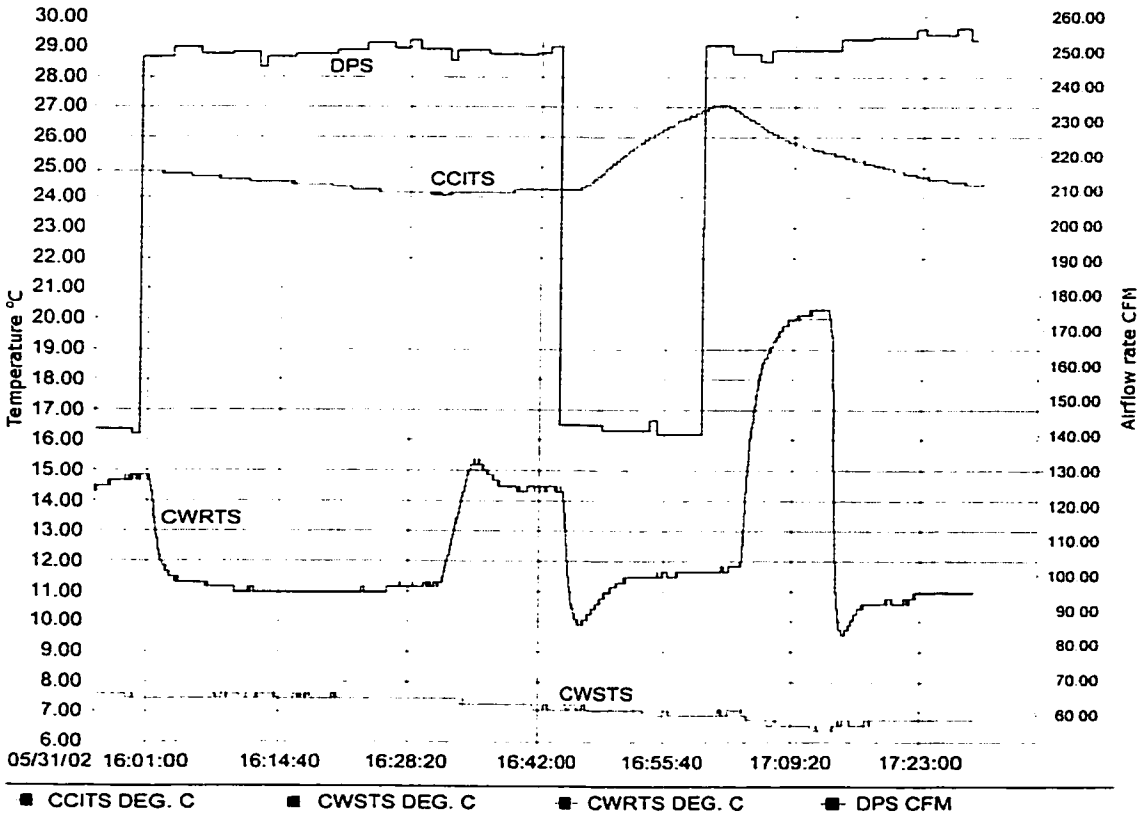

Figure 5.21 Multi disturbances for the CMBA PI control

In Test case 10, the effect of interactions between the airflow control loops and the DAT control loop were examined. The results are shown in Figures 5.22 and 5.23. In this test, the two airflow control loops were activated in PI control mode and the effects of multiple disturbances and setpoint changes as noted in Table 5.4 (Test case 10) were simulated. From Figure 5.22, it can be seen that the CMBA PI control results in an overshoot ± 1°C from setpoint and rapidly die out within 15 – 20 minutes and the output reaches a steady-state value. The evolution of the PI gains during the Test case 10 is shown in Figure 5.24. It is apparent that a continuous change in control gains was achieved to counter the effects of multiple disturbances, setpoint changes and interactions from the other control loops in the system.

The results presented in Figures 5.14 through 5.24, show the advantages of CMBA PI control in achieving good tracking over a wide range of operating conditions. Therefore, the results substantiate the strong robustness property of CMBA PI control.

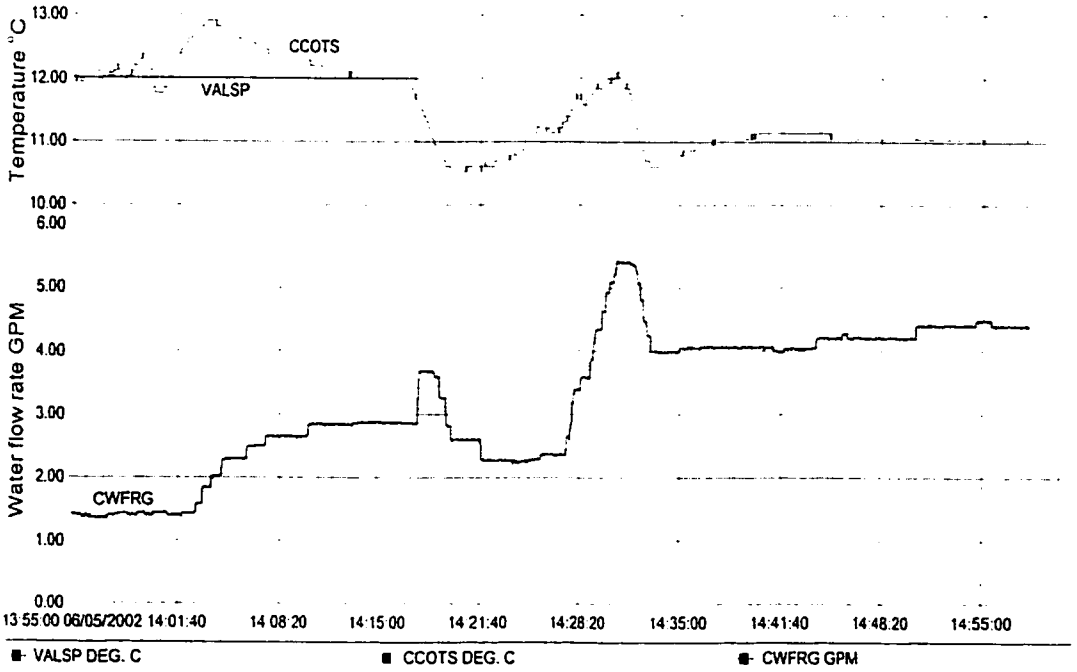

Figure 5.22 Effect of multi disturbances on the CMBA PI control under multi closed loops

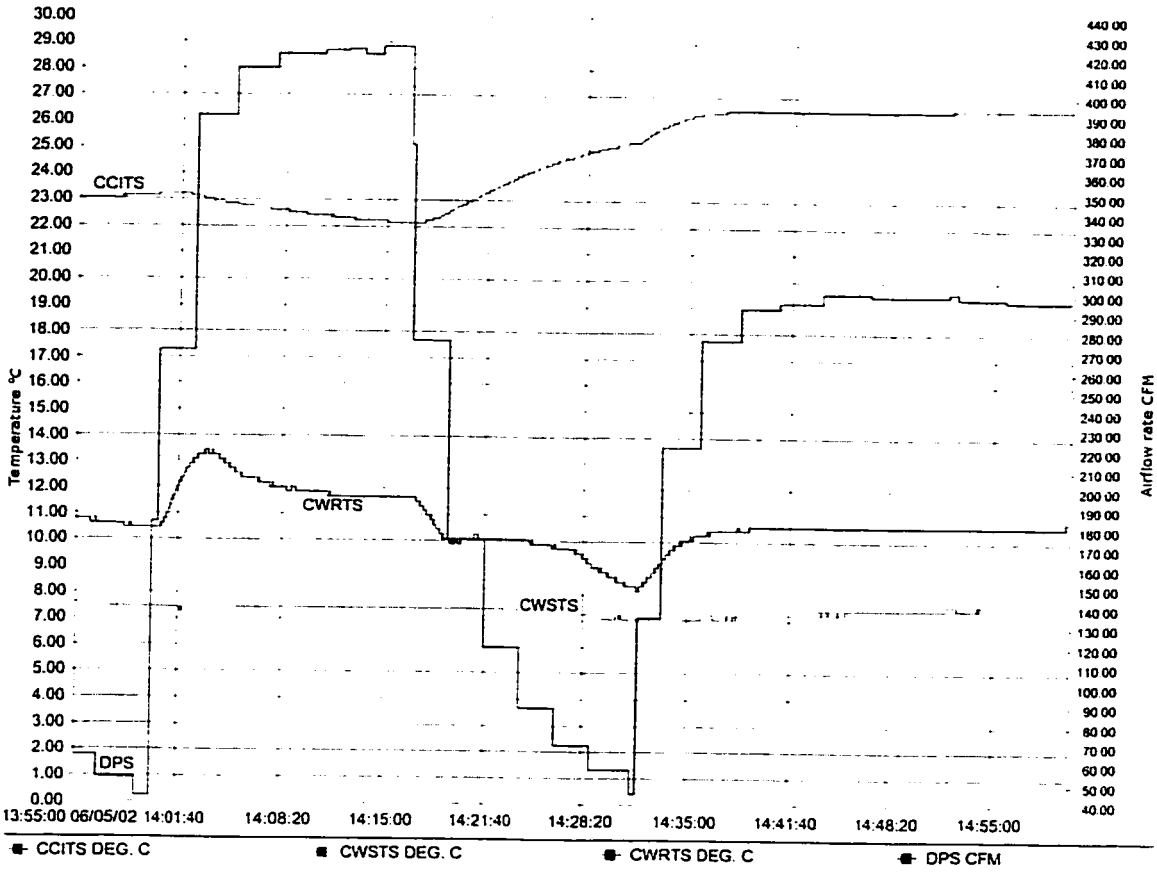

Figure 5.23 **Multi disturbances for the CMBA PI control under multi closed loops**

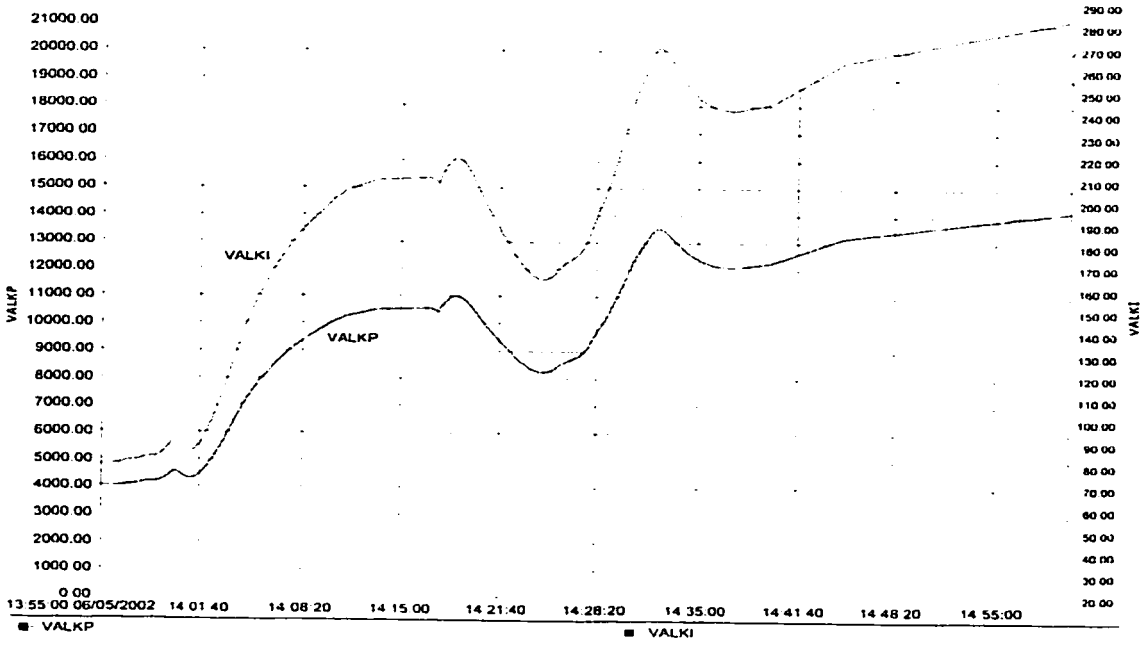

Figure 5.24 **PI parameter changes for the CMBA PI control under multi closed loops**

5.3.4 Comparison with Constant PI

In this sub-section, we first show that constant PI for discharge air temperature control is not suitable because the disturbances can induce severe oscillations in output responses. Then we show how the CMBA PI control can damp out the oscillatory output responses.

Figures 5.25, 5.27 and 5.29 show the constant PI control responses for the setpoint changes under different disturbances: Case A to Case C. The responses illustrate that a constant PI control may work well under some conditions, but it may not track setpoint change well under other conditions that the discharge system may be subjected to during its operation.

The experiments were conducted on September 5, 2002 for Cases A and B, and on September 6, 2002 for Case C. For the three cases the same set of PID parameters was used: VALKP = 6000, VALKI = 120 and VALKD = 0.

Case A: the set-point (VALSP) was changed from 11 °C to 13 °C at 13:10:40; the operating conditions were DPS = 210.0 ± 3.5 CFM, CCITS = 24.4 ± 0.25 °C, CWSTS = 8.4 ± 0.6 °C (decreasing from 9.0 to 7.8 °C).

Case B: the set-point (VALSP) was changed from 11 °C to 13 °C at 16:48:54; the operating conditions were DPS = 210.0 ± 3.5 CFM, CCITS = 23.7 ± 0.25 °C, CWSTS = 9.9 ± 1.16 °C (increasing from 8.8 to 11.0 °C).

Case C: the set-point (VALSP) was changed from 11 °C to 13 °C at 10:55:12, from 13 °C to 11 °C at 11:16:58 and from 11 °C to 13 °C at 11:57:03; the operating conditions were DPS = 425.0 ± 5.0 CFM, CCITS = 22.8 ± 0.5 °C, CWSTS = 8.0 ± 0.25

°C before 11:39:00, and DPS = 300.0 ± 5.0 CFM, CCITS = 24.0 ± 0.5 °C, CWSTS = 7.7 ± 0.3 °C after 11:51:00.

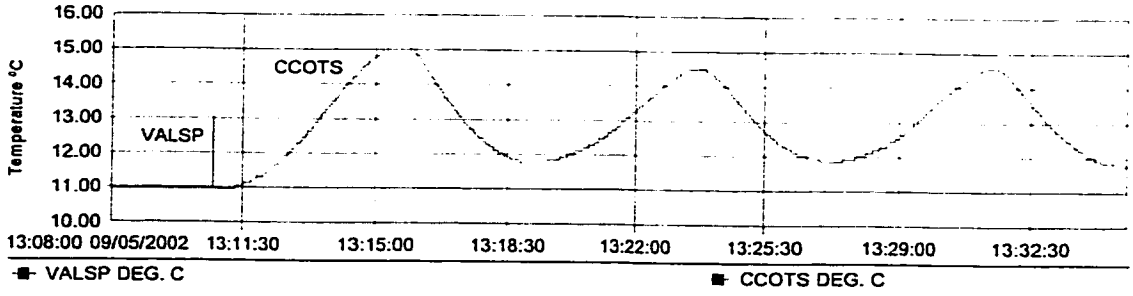

Figure 5.25 Experimental result with oscillations for constant PI control

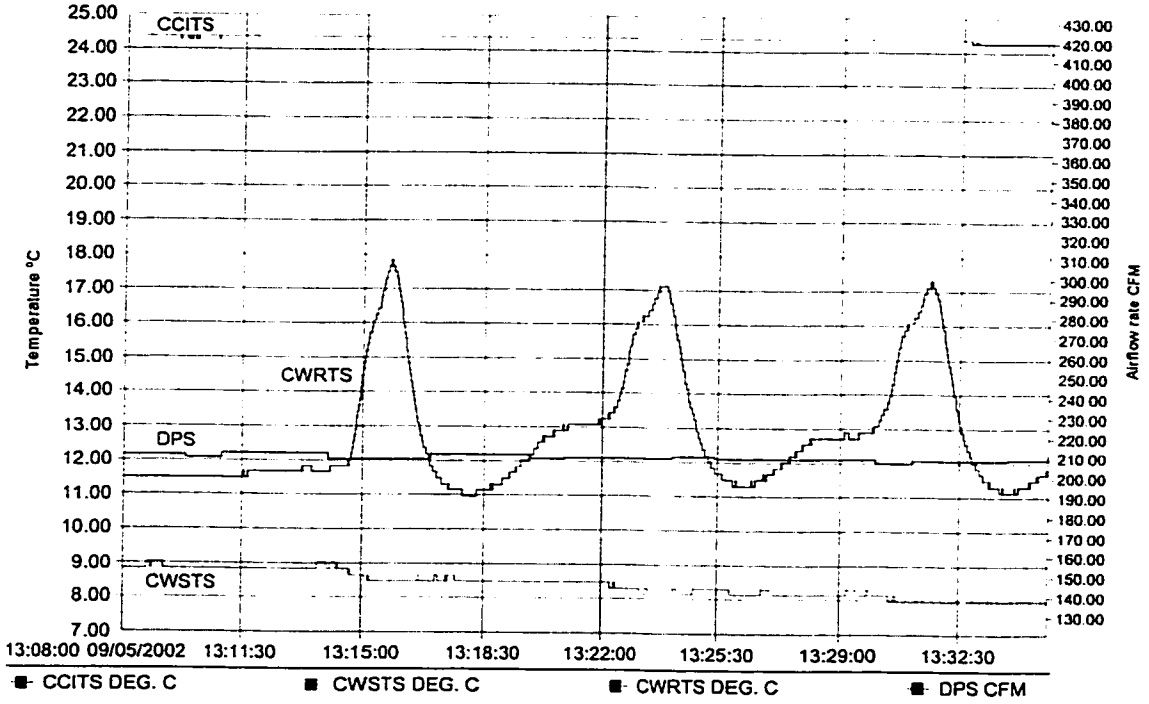

Figure 5.26 Disturbances in constant PI control with oscillations in the response

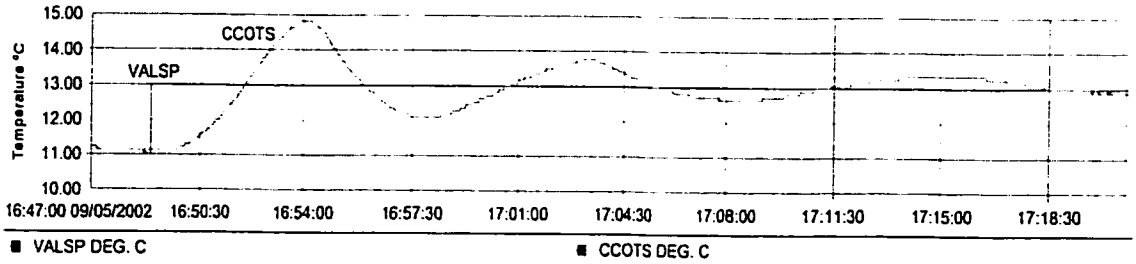

Figure 5.27 Experimental result with damping for the constant PI control

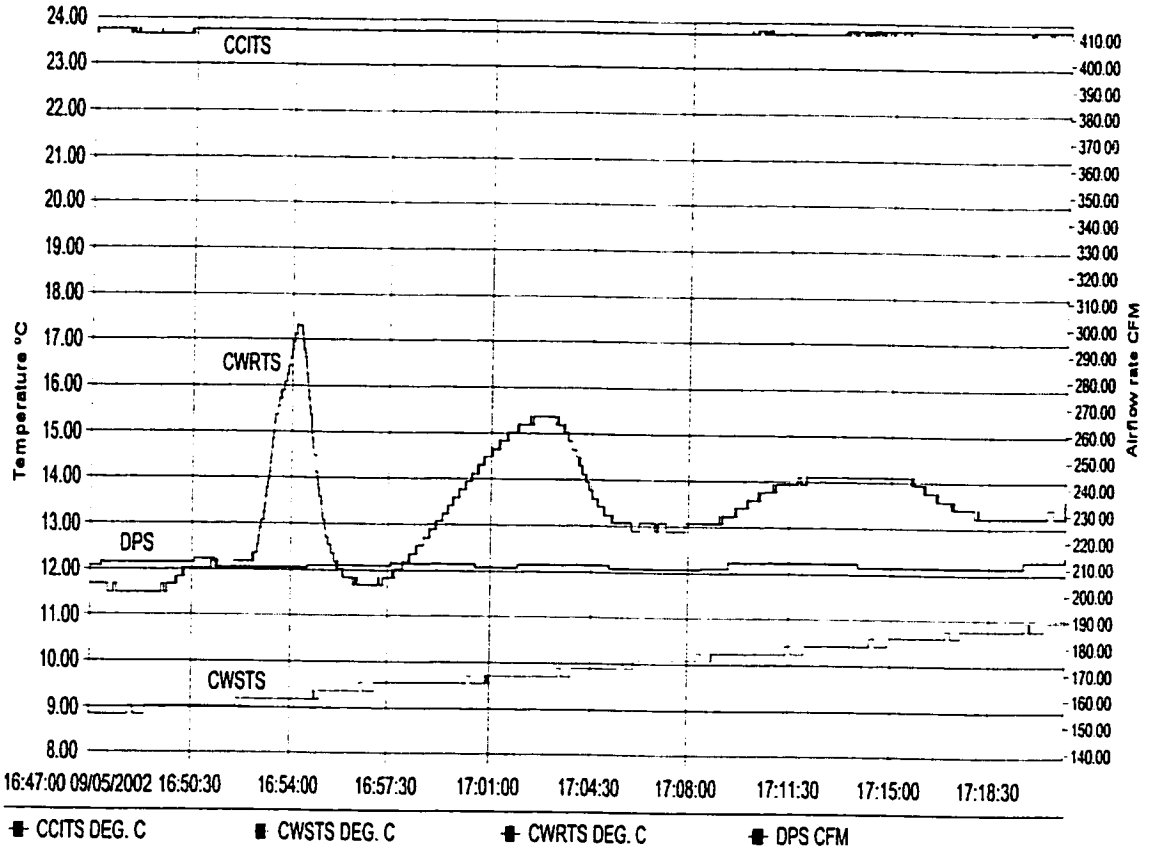

Figure 5.28 **Disturbances in constant PI control with damping response shown in Figure 5.27**

From Figures 5.26 and 5.28, it is noted that the main difference between Case A and B was in the chilled water supply temperature (CWSTS). The chilled water supply temperature was decreasing in Case A, which made the constant PI control response oscillatory (Figure 5.25). On the other hand CWSTS was increasing in Case B, which made the constant PI control response convergent (Figure 5.27).

Figure 5.29 shows how a change in total airflow rate (DPS) from 425.0 ± 5.0 CFM to 300.0 ± 5.0 CFM at 11:39:00 made a stable constant PI control become oscillatory as soon as the setpoint was increased from 11 to 13 °C at 11:51:00 hours. This shows the limitation of constant gain PI control.

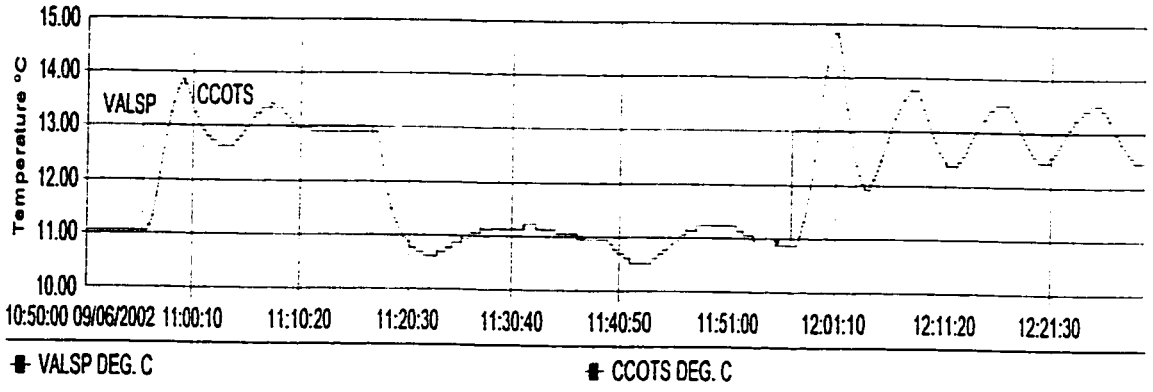

Figure 5.29 Experimental result for constant PI control

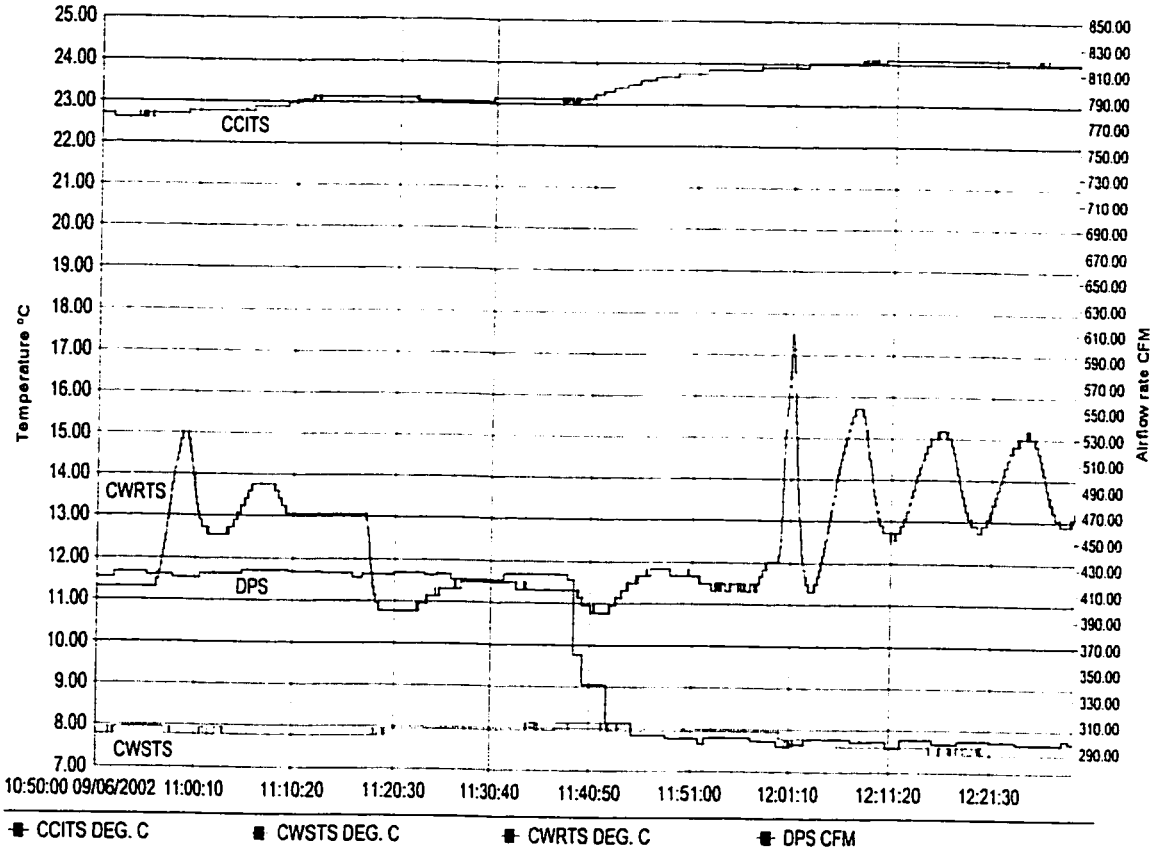

Figure 5.30 Disturbances for the constant PI control

To show the self adaptive regulation properties of CMBA PI control in contrast to constant PI control, the following case was considered.

The discharge air temperature system was controlled by constant PI control from 11:55:00 to 12:36:00 and from 13:43:00 to 14:45:00. CMBA PI control was activated from 12:36:00 to 13:43:00 and from 14:45:00 to 15:35:00. The results are shown in Figures 5.31 and 5.32.

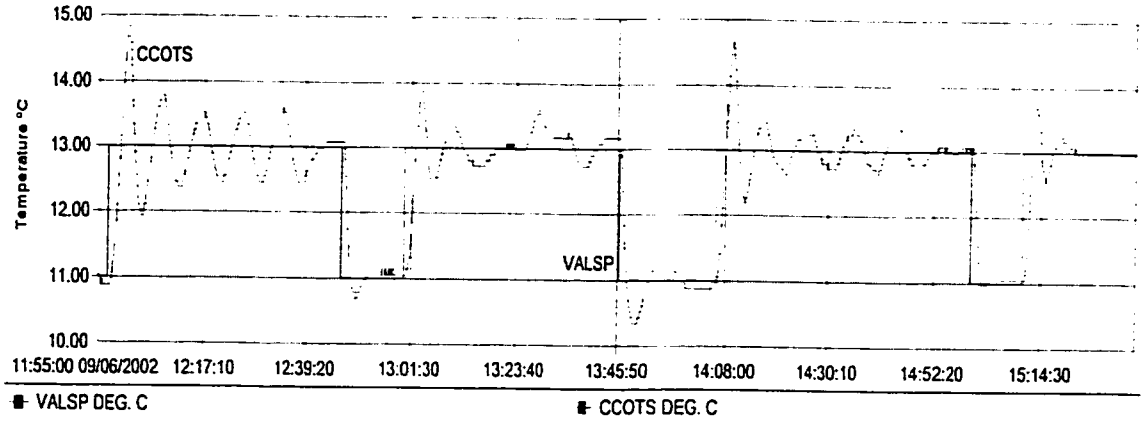

Figure 5.31 Experimental result for constant PI control and CMBA PI control

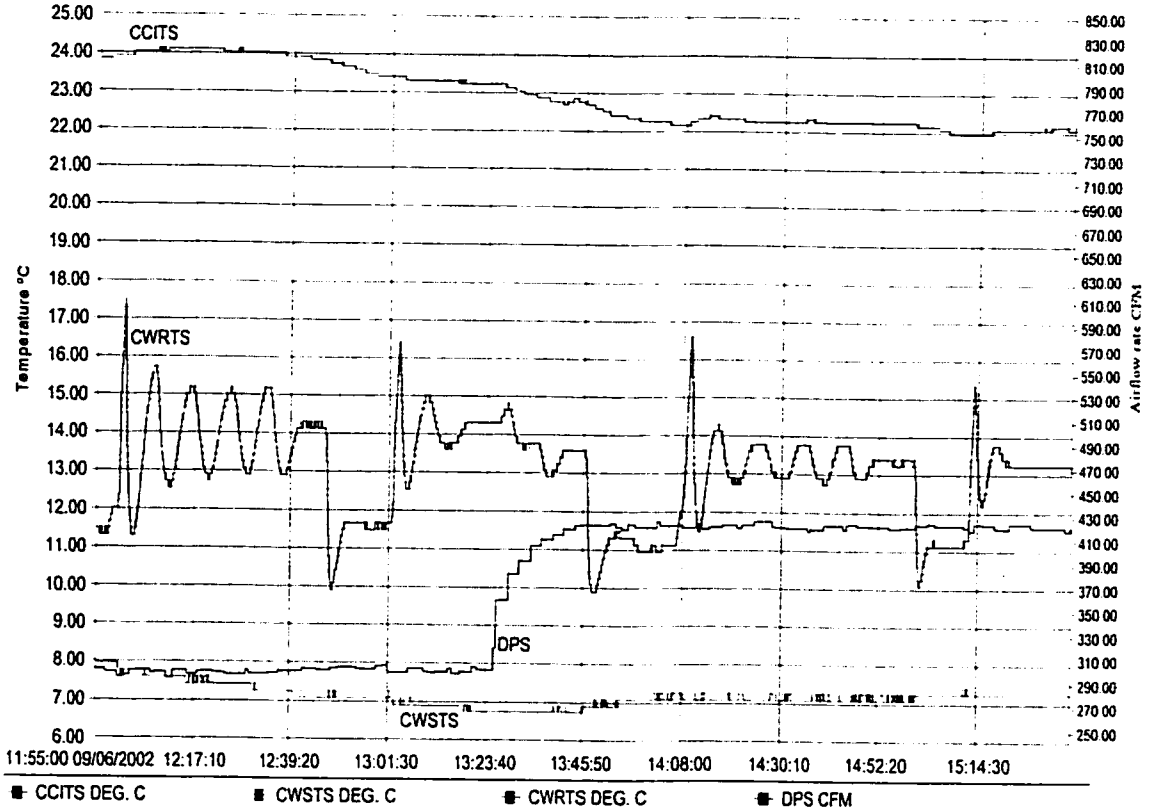

Figure 5.32 Disturbances for constant PI control and CMBA PI control

From Figures 5.31 and 5.32, we see that the discharge air temperature oscillates (with constant PI control) under both lower and higher cooling loads and airflow rates (298.0 ± 7.0 CFM and 420.0 ± 5.0 CFM in Figure 5.32) when the setpoint was increased. Under these oscillatory conditions the CMBA PI control was activated to take over the control action between 12:36:00 – 13:43:00 and between 14:45:00 – 15:35:00. It is apparent that the output responses converged rapidly under CMBA PI control showing its effectiveness in handling different operating conditions encountered in HVAC systems.

The above experimental results illustrate that a constant PI control is not suitable for discharge air temperature systems because of its weak adaptability to disturbance changes. The CMBA PI control is well suited for discharge air temperature systems because it is able to adapt to several different operating conditions effectively.

5.4 H_∞ Adaptive PI Control System

In this section, the implementation of the H_∞ adaptive PI control as an application of EMCS is presented, and then the experimental results for different environmental conditions are shown. The experimental results indicate that the H_∞ adaptive PI control have acceptable performance for discharge air temperature control in HVAC systems. A comparison with constant PI control is also presented.

5.4.1 Interface Modules for H_∞ Adaptive PI Control System

The interface modules are designed to monitor and change the variables of interest in the H_∞ adaptive PI control algorithm. In all there are four modules: H_∞

Adaptive PI Control (called HIAPI Control), an Identifier, a PID Tuner and a PID module.

The H_∞ Adaptive PI Control module (Figure 5.33) is designed as the main interface module to show the control system structure and display the real values for the setpoint – desired discharge air temperature (VALSP) and the system output – discharge air temperature (CCOTS). We can also click on any one of the blocks with an arrow to go to the related sub-interface module.

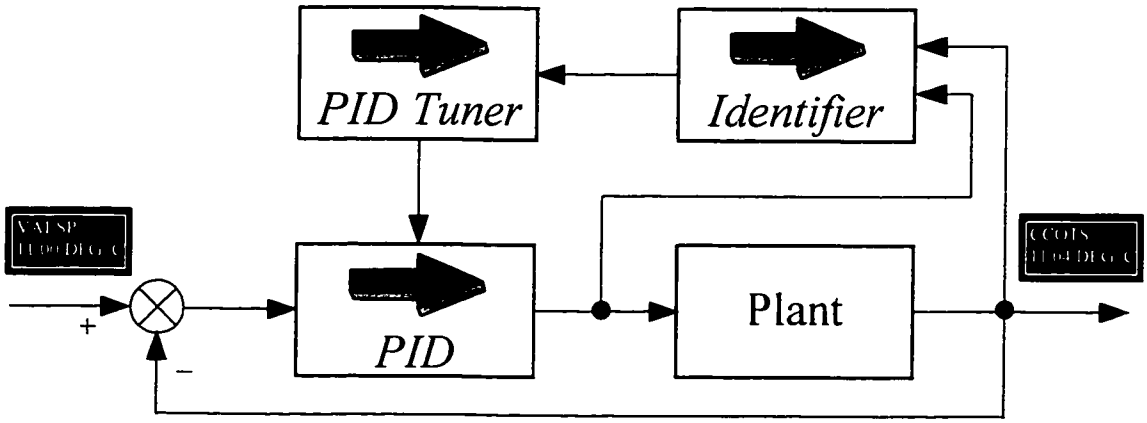

Figure 5.33 Main Interface module of HIAPI Control

The interface module for the Identifier (Figure 5.34) is designed to monitor the identification performance for the plant by displaying the related inputs and outputs of the identification program. The inputs of the identification program are the controller output – VALOP (as the input of the FOPDT model), the discharge air temperature (as the output of the FOPDT model), the entering air temperature – CCITS (as the output of the FOPDT model when the input of the FOPDT model equals zero), and the dead time of the FOPDT model (number of samples – LOfModel). The outputs of the module are the dead time of the FOPDT model expressed as the number of samples, the parameters aOfModel ($\hat{a}$ of the FOPDT model) and bOfModel ($\hat{b}$ of the FOPDT model), and the

status of the identification shown as IdentFlag ON (updating) or OFF (waiting/collecting data).

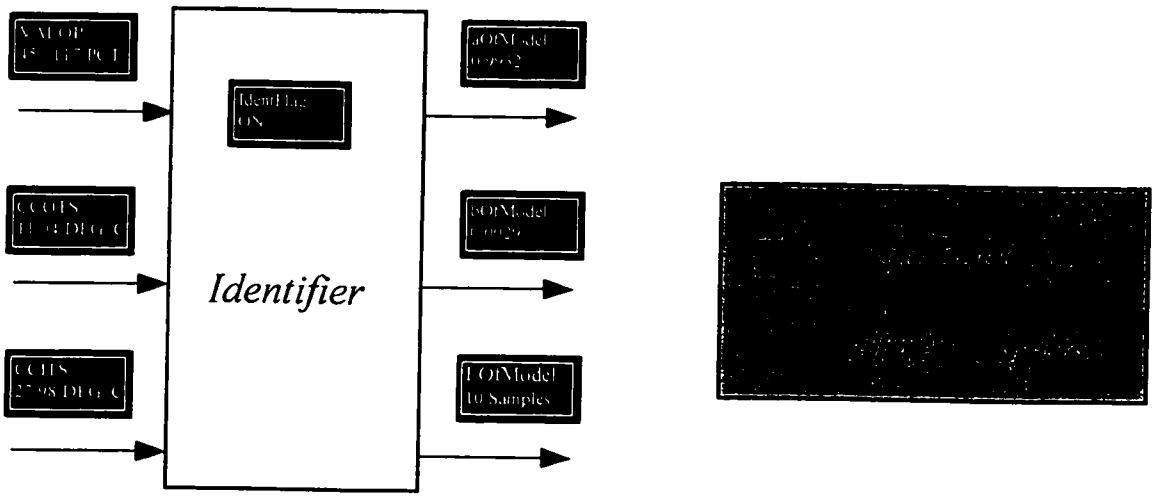

Figure 5.34 Interface module for the Identifier

The interface module of the PID Tuner (Figure 5.35) is designed to monitor the PI parameters using the H_∞ PI tuning rules. Also the related inputs and outputs of the PI Tuner are displayed. The inputs of the PID Tuner are the parameters of the FOPDT model and the program control signal (UpdatingControl). The outputs are the computed PID parameters: VALKPEM, VALKIEM and VALKDEM.

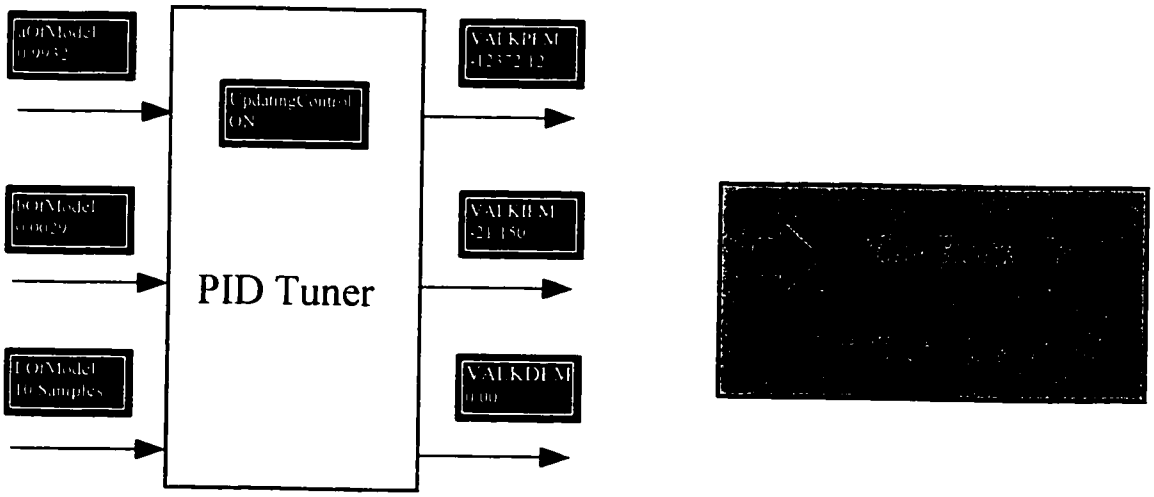

Figure 5.35 Interface module for the PID Tuner

The interface module for the PID (Figure 5.36) is designed to show the updated PID parameters (VALKP, VALPI and VALPD) and display the input and output of the PID controller. The input (VALE) is the difference between VALSP and CCOTS. The output is VALOP. Each of the three sub-interface modules has a block that can be clicked as to go back to the main interface module of H_∞ Adaptive PI Control (HIAPI Control).

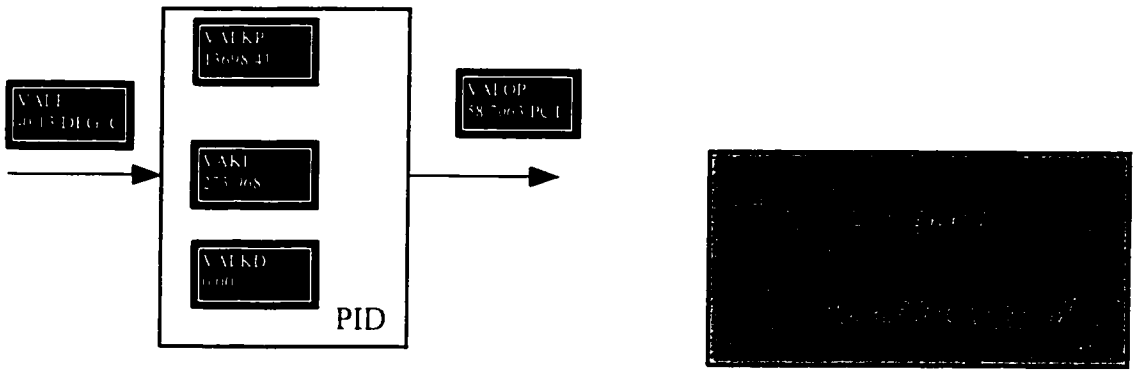

Figure 5.36 Interface module for the PID parameters

5.4.2 Effects of Setpoint Changes and Dead-time

In this sub-section, the experimental results of the H_∞ adaptive PI control performance for setpoint changes with different dead-time are presented and comparisons are made with constant PI responses.

Figures 5.37 – 5.39 show the H_∞ adaptive PI control responses for setpoint changes with three different choices of the length of dead-time referred to as Test case 1 through 3, listed in Table 5.5.

Test case	Date	Time	Operating conditions	Identification and control parameters	Figure
1	June 4, 2002	14:04:14	Setpoint: 13 to 11 °C; 11 to 13 °C CFM: 268 ± 3 Inlet air temp. : 23 ± 0.3 °C Inlet water temp. : 5.8 ± 0.3 °C	dead-time: 7 Samples (28 Sec) $K_{pd\max} = -4000$ $K_{pd\min} = -24000$	5.37
2	June 4, 2002	14:55:50	Setpoint: 13 to 11 °C; 11 to 13 °C CFM: 268 ± 3 Inlet air temp. : 27.5 ± 0.3 °C Inlet water temp. : 6.3 ± 0.3 °C	dead-time: 10 Samples (40 Sec) $K_{pd\max} = -4000$ $K_{pd\min} = -24000$	5.38
3	June 6, 2002	11:05:32	Setpoint: 13 to 11 °C; 11 to 13 °C CFM: 360 ± 3 Inlet air temp. : 24.3 ± 0.25 °C Inlet water temp. : 7.2 ± 0.25 °C	dead-time: 9 Samples (36 Sec) $K_{pd\max} = -4000$ $K_{pd\min} = -25000$	5.39 5.40 5.41
4	June 6, 2002	a) 17:00:00 b) 17:21:03 c) 17:37:00	Setpoint: 13 to 11 °C; 11 to 13 °C; 13 to 11 °C CFM: 213 ± 6 Inlet air temp. : 22.8 ± 0.2 °C Inlet water temp. : 6.8 ± 0.6 °C	a) constant PI: $K_p = 6000$ $K_i = 120$ b) constant PI: $K_p = 4000$ $K_i = 80$ c) dead-time: 9 Samples $K_{pd\max} = -1000$ $K_{pd\min} = -25000$	5.42 5.43 5.44
5	Sep. 10, 2002	a) 16:40:00 b) 16:55:00	Setpoint: 11 to 13 °C; 13 to 11 °C; 11 to 13 °C CFM: 210 ± 5 Inlet air temp. : 25 ± 0.5 °C Inlet water temp. : 6.5 ± 0.5 °C	a) constant PI: $K_p = 6000$ $K_i = 120$ b) dead-time: 9 Samples $K_{pd\max} = -4000$ $K_{pd\min} = -24000$	5.45 5.46 5.47

Table 5.5 HIAPI Control: Setpoints and operating conditions

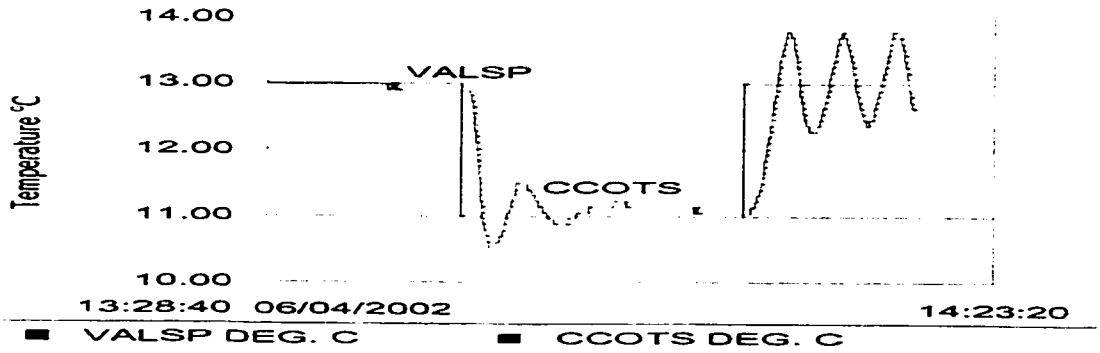

Figure 5.37 **Output response of the HIAPI control with dead-time equal to 7 samples (case 1)**

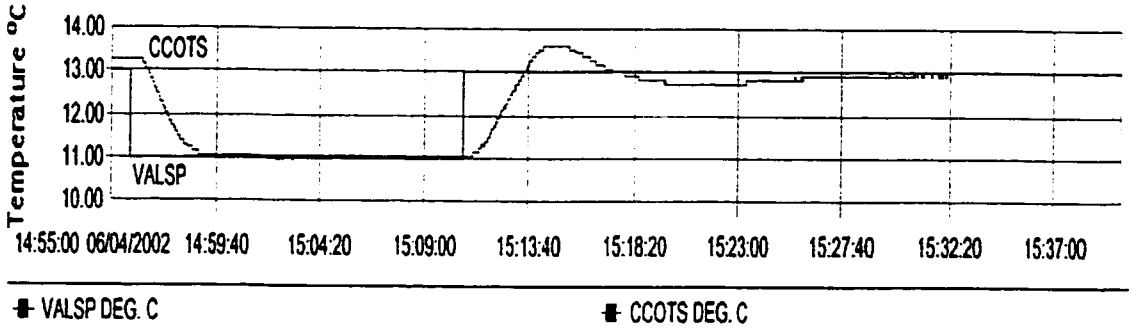

Figure 5.38 **Output response of the HIAPI control with dead-time equal to 10 samples (case 2)**

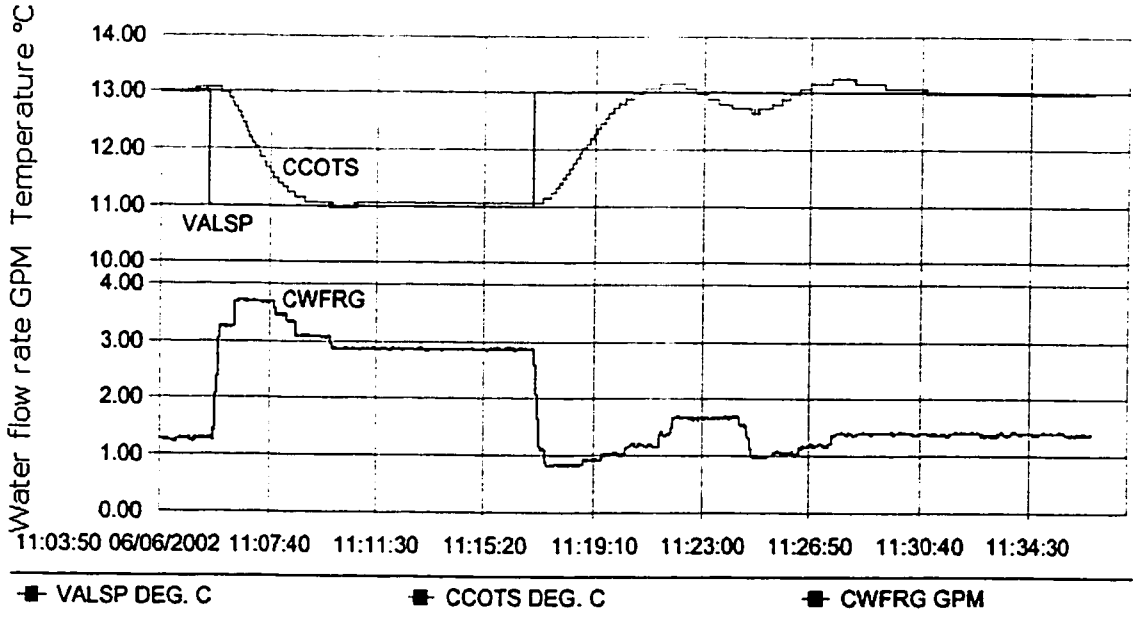

Figure 5.39 **Output response of the HIAPI control with dead-time equal to 9 samples (case 3)**

In Figure 5.37, we see that the response with a dead-time of 7 samples (28 sec) is oscillatory especially when the setpoint is increased from 11 to 13 °C. The main reason is that a dead-time of 28 sec for the FOPDT model was too small. From the open-loop responses, it was estimated that a dead-time of 36 to 40 sec may be suitable. Therefore, tests were conducted with dead-times of 40 sec (Figure 5.38, Test case 2) and 36 sec (Figure 5.39, Test case 3).

The smooth experimental responses shown in Figure 5.38 and 5.39 indicate that a dead-time delay of 10 samples (40 sec) and/or 9 samples (36 sec) is appropriate in describing the DAT system as an FOPDT model. The evolution of chilled water flow rate responses obtained from Test case 3 is also shown in Figure 5.39. Furthermore, how the FOPDT model parameters a and b varied during the test are shown in Figure 5.40. And the evolution of the updated PI gains from the H_∞ PI tuning rules is shown in Figure 5.41. It is interesting to note that even through the model parameters and PI gains are changing rapidly, the output responses remain smooth and stable with good setpoint tracking.

Shown in Figures 5.42 to 5.44 (Test case 4) and Figures 5.45 to 5.47 (Test case 5) are two sets of comparisons of the responses obtained from the constant PID control and HIAPI control. In the first set (Figures 5.42 to 5.44), the DAT system was operated with constant PI control. The CCOTS response as shown in Figure 5.42 is oscillatory at about 17:37:00 the HIAPI was activated resulting in a stable response reaching steady state in about 5 minutes. The evolution of PI parameters and operating conditions during this test (case 4) are shown in Figures 5.43 to 5.44.

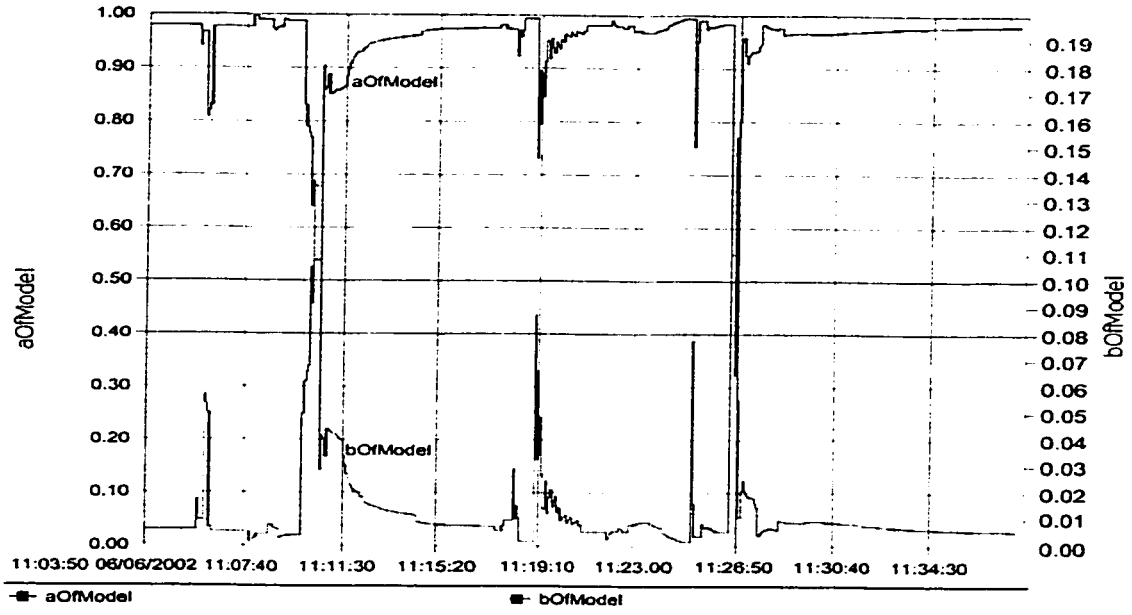

Figure 5.40 The evolution of FOPDT model parameters: a and b

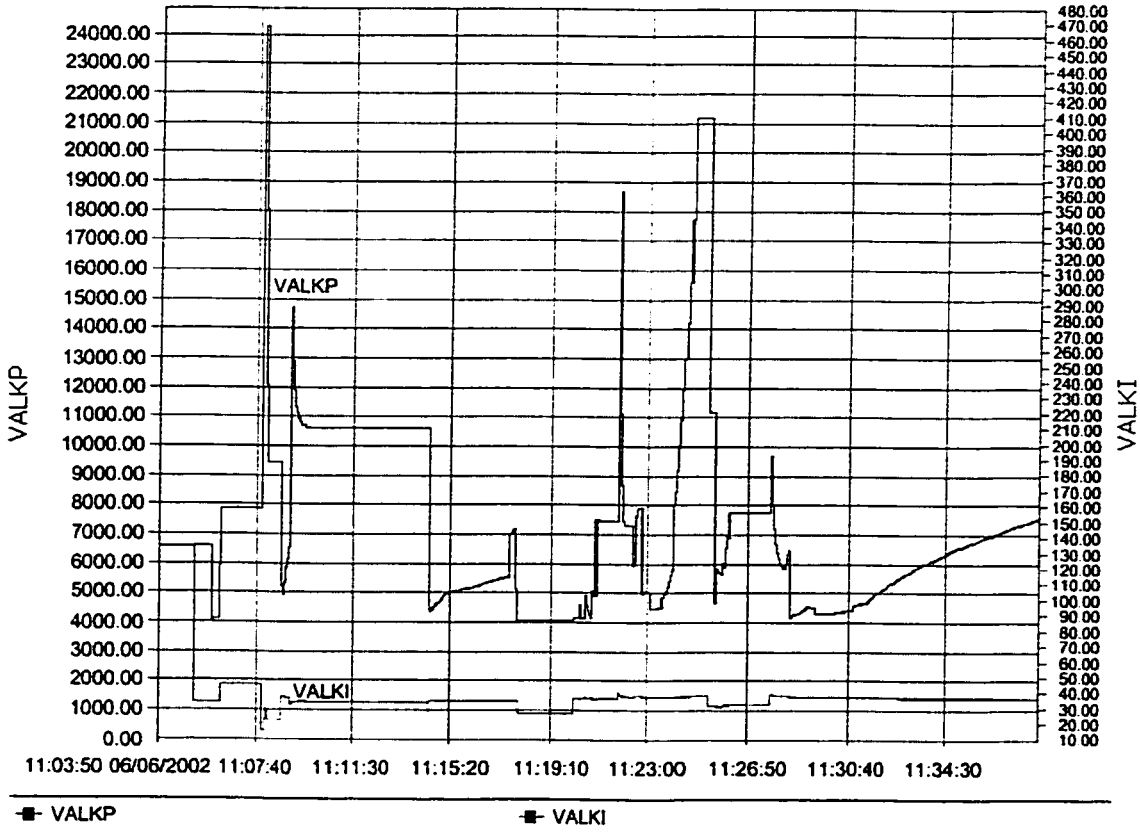

Figure 5.41 Evolution of the updated PI parameters for the HIAPI control

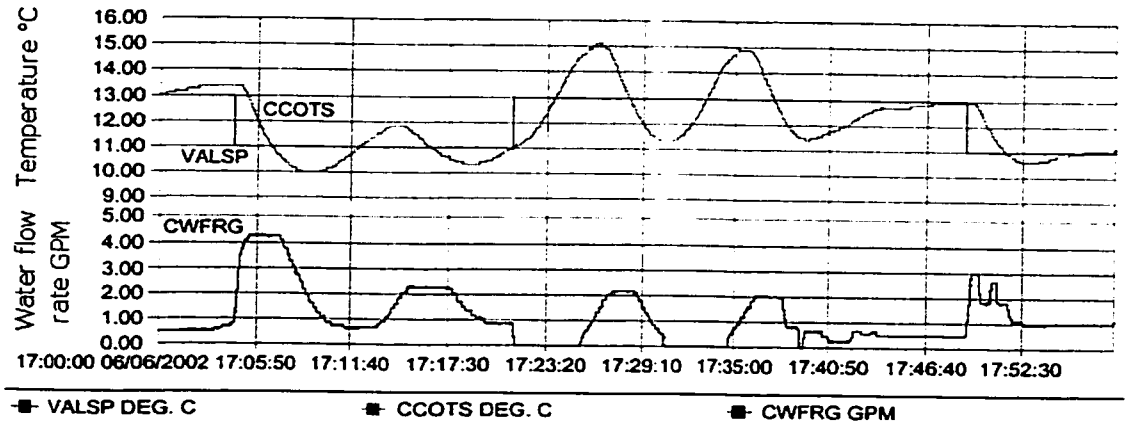

Figure 5.42 Comparison of the output responses for the PI and HIAPI control schemes (case 4)

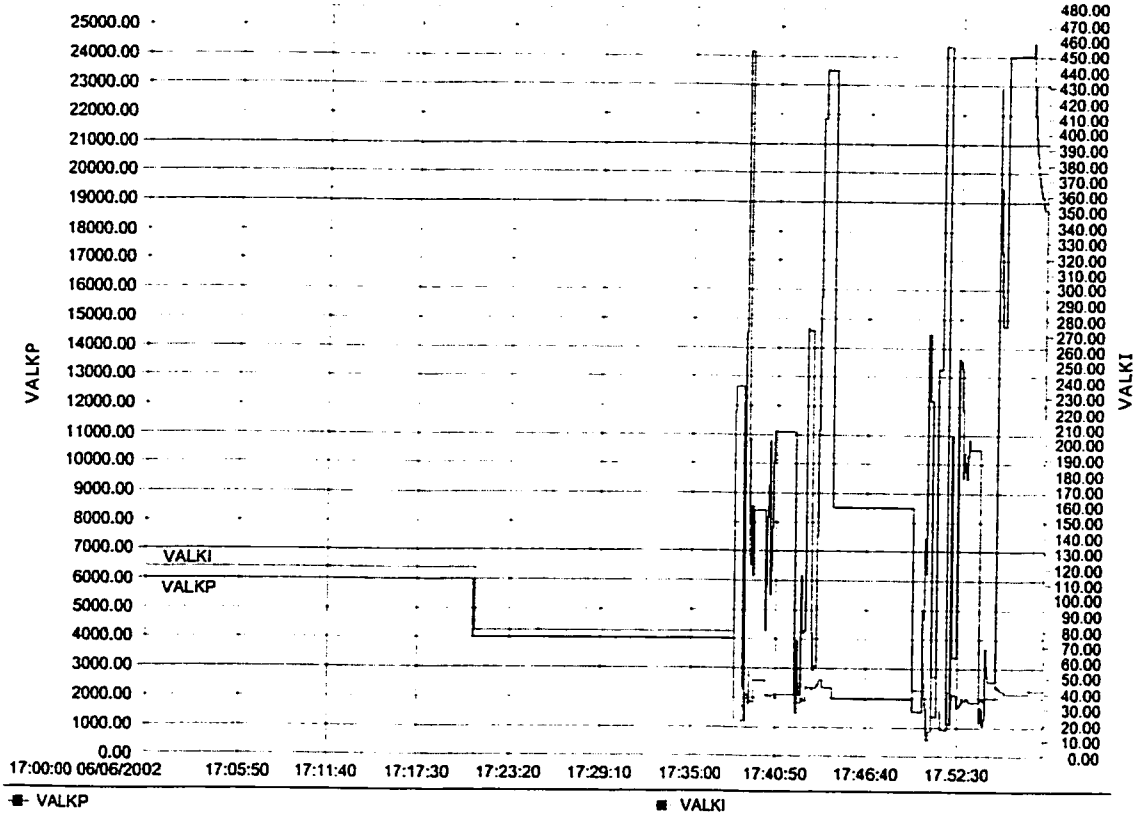

Figure 5.43 Comparison of PI parameters for the PI and HIAPI control schemes (case 4)

The response shown in Figure 5.42 and the PI parameter changes shown in Figure 5.43 indicate that using H_∞ adaptive PI control it is very easy to reach steady-state by self-tuning even though the initial state of the discharge air system is oscillating.

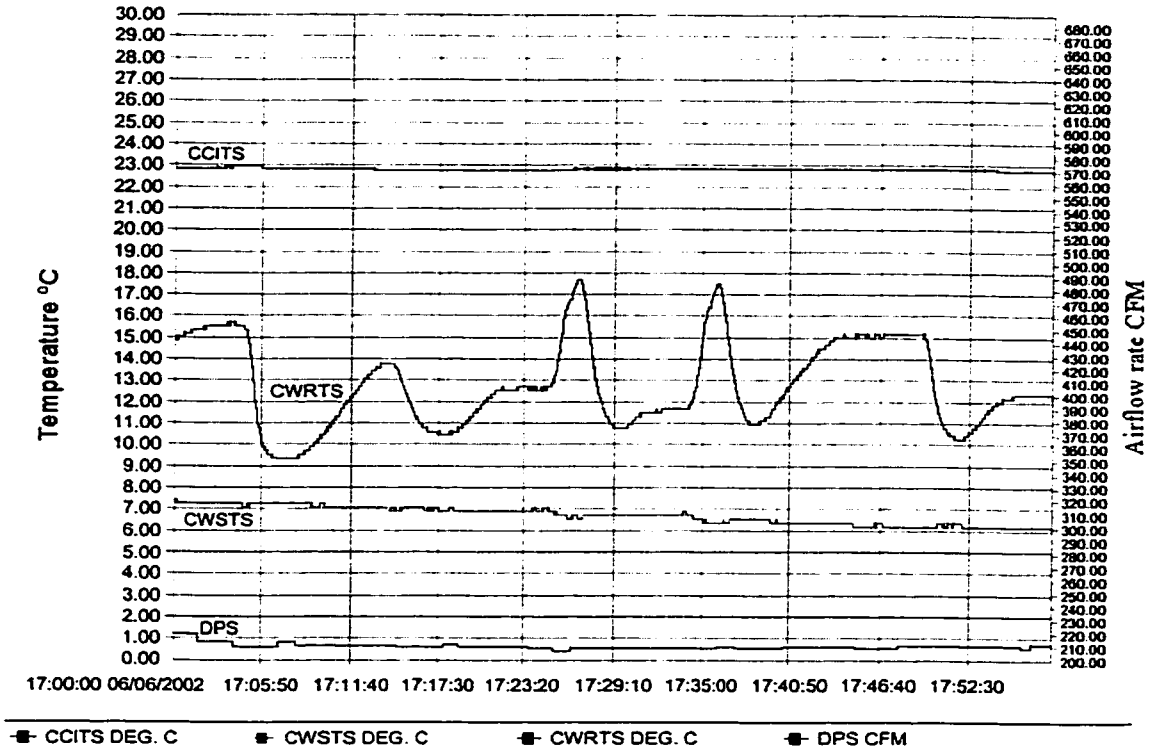

Figure 5.44 Disturbance profiles for the PI and HIAPI control schemes (case 4)

The second set of results shown in Figure 5.45 to 5.47 show a similar advantage for HIAPI control over constant PI control. As shown in Figure 5.45 the HIAPI control was activated at 16:55:00 hours resulting in a stable and smooth response.

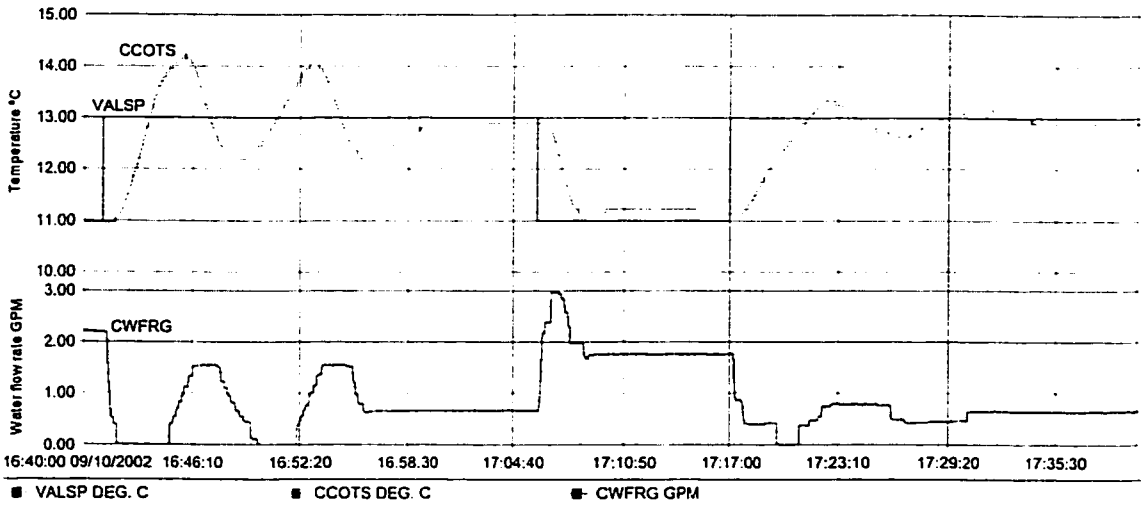

Figure 5.45 Comparison of output responses for the PI and HIAPI control schemes (case 5)

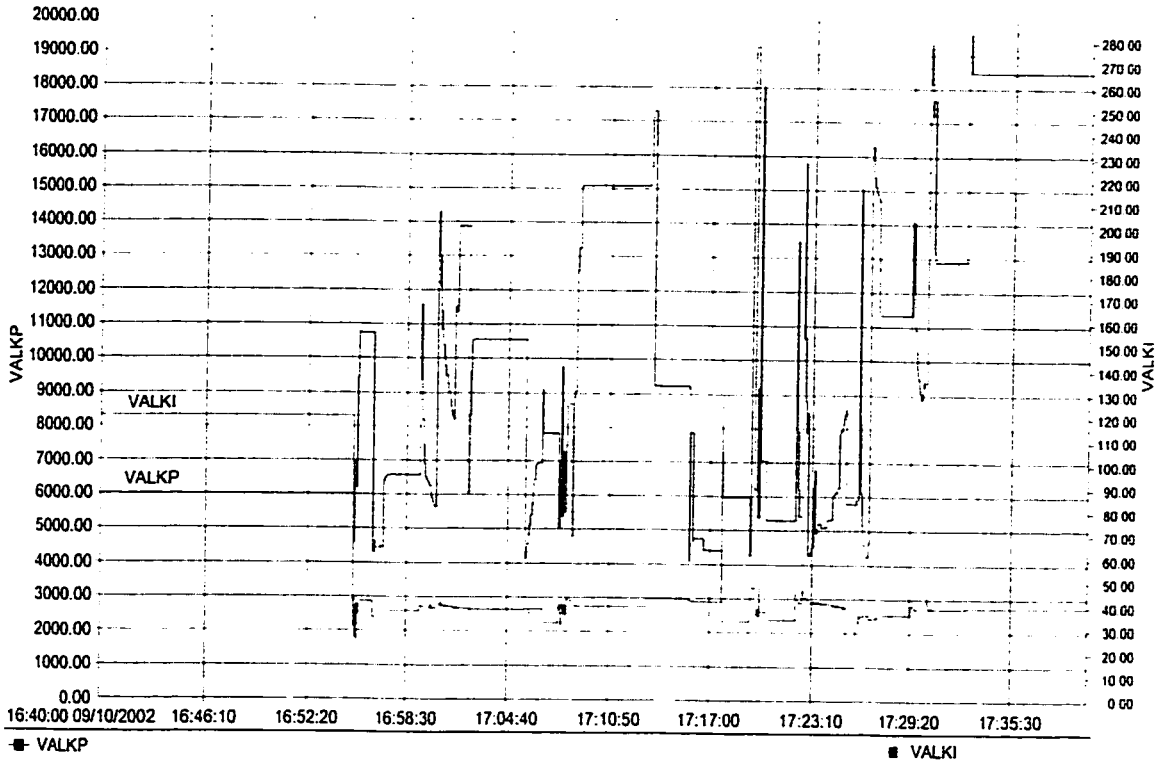

Figure 5.46 PI parameters for the PI and HIAPI control schemes (case 5)

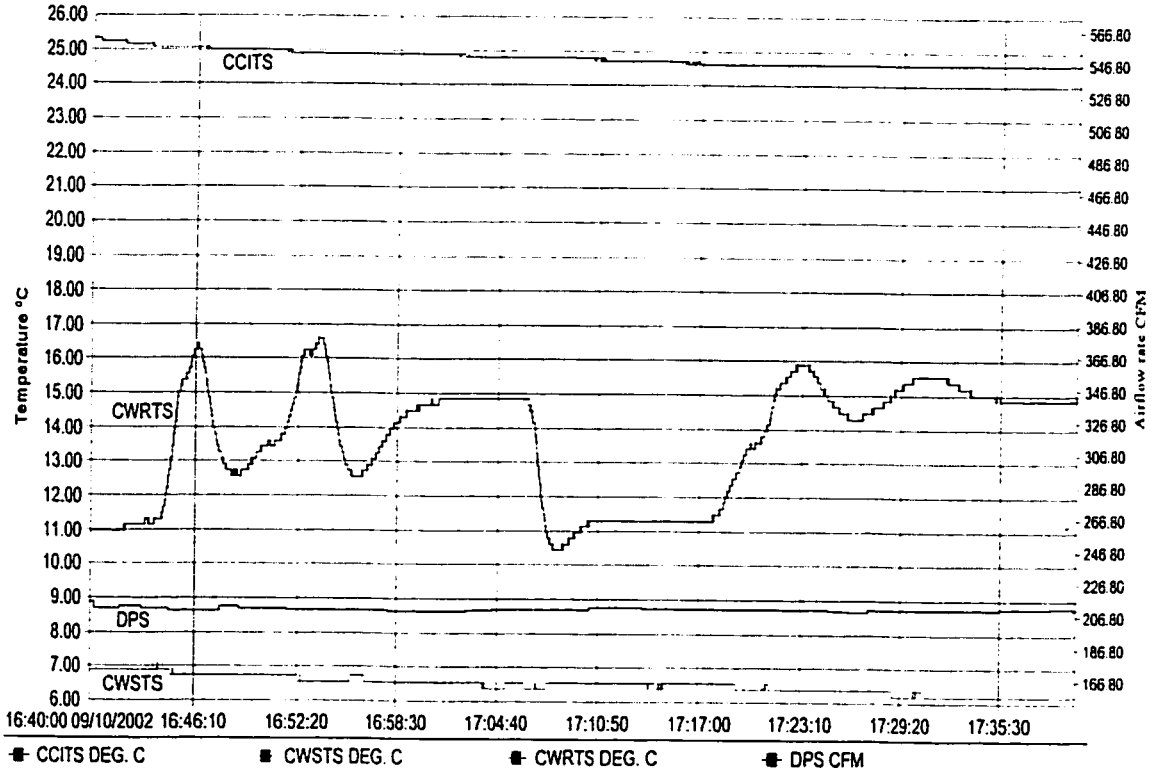

Figure 5.47 Disturbance profiles for the PI and HIAPI control schemes (case 5)

5.4.3 Effects of Disturbances

In this section, the results of H_∞ adaptive PI control performance subject to disturbance effects are presented. The disturbances are the same as in the CMBA PI control: namely changes in entering air temperature (CCITS), chilled water supply temperature (CWSTS), and airflow rate (DPS). Figures 5.48 – 5.57 show the results for the H_∞ adaptive PI control system subject to four different levels of disturbances and operating conditions. These are referred to as Test cases 6 through 9 and are listed in Table 5.6.

Test case	Date	Time	Operating conditions	Identification and control parameters	Figure
6	June 6, 2002	12:19:59	Setpoint: 13 to 11 °C; 11 to 13 °C CFM: 422 ± 5 Inlet air temp. : 23.8 to 22.3 °C Inlet water temp. : 7.5 ± 0.3 °C	dead-time: 9 Samples (36 Sec) $K_{pd\max} = -500$ $K_{pd\min} = -25000$	5.48 5.49
7	June 6, 2002	14:08:40	Setpoint: 13 °C CFM: 422 ± 5 Inlet air temp. : 267.5 ± 0.2 °C Inlet water temp. : 7.1 to 8 °C	Same as in case 6	5.50 5.51
8	June 5, 2002	18:28:32	Setpoint: 11 to 13 °C CFM: 210 to 424 Inlet air temp. : 27.5 ± 0.8 °C Inlet water temp. : 8.5 ± 0.5 °C	dead-time: 9 Samples (36 Sec) $K_{pd\max} = -4000$ $K_{pd\min} = -25000$	5.52 5.53
9	June 6, 2002	13:20:36	Setpoint: 13 °C CFM: 425 to 210 Inlet air temp. : 22.2 to27.2 °C Inlet water temp. : 7.7 to 6.8 °C	Same as in case 6	5.54 5.55 5.56 5.57

Table 5.6 HIPIA Control: Disturbances and operating conditions

Figure 5.48 shows the effect of decreasing the entering air temperature (case 6) on the output response for the H_∞ adaptive PI control system. The H_∞ adaptive PI control

performs very well – lower overshoot (less than 0.3 °C) and good tracking speed (the settling times with 0.5 °C allowable tolerance were less than 2.5 minutes). The changes in the entering air temperature are shown in Figure 5.49.

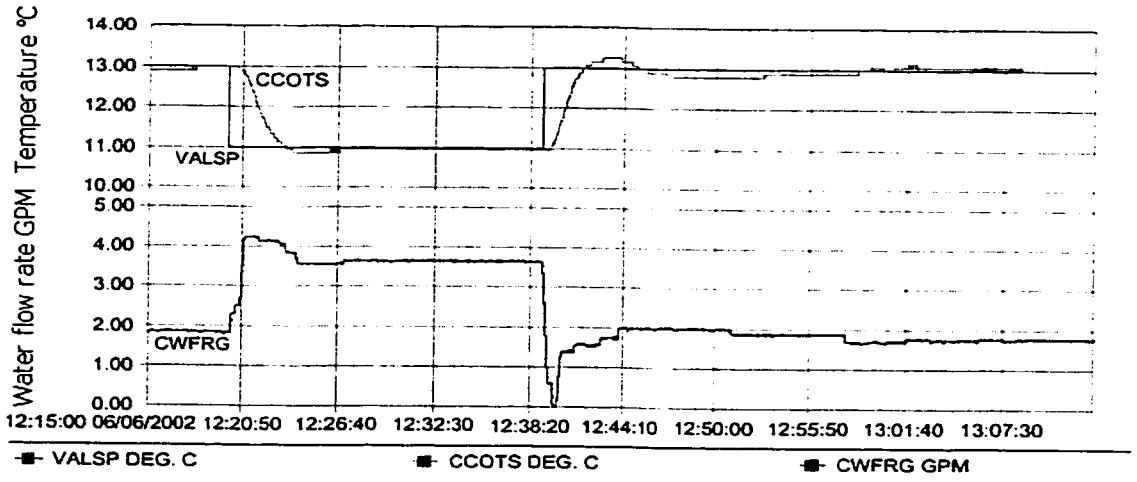

Figure 5.48 Effect of entering air temperature changes on the HIAPI control

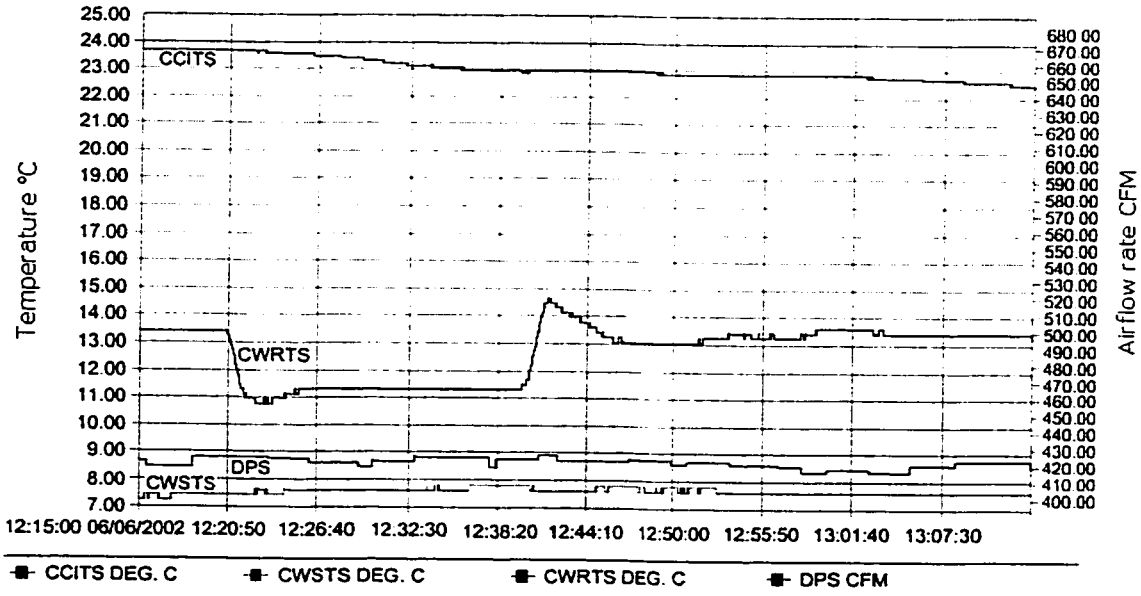

Figure 5.49 Entering air temperature profile in the HIAPI control

Figure 5.50 shows that the effect of change in the chilled water supply temperature (Figure 5.51) (case 7) is also small. The largest difference of the discharge air temperature and the desired air temperature was less than 0.25 °C. Figure 5.52 shows

that the effect of airflow rate changes (case 8) were large but acceptable and the output responses from the H_∞ adaptive PI control system reach stead-state rapidly. The largest difference between the discharge air temperature and the desired air temperature was less than 0.8 °C. Figure 5.54 shows the effect of multiple disturbances (case 9) on the output response of the H_∞ adaptive PI control system. The largest difference from the setpoint was less than 0.5 °C.

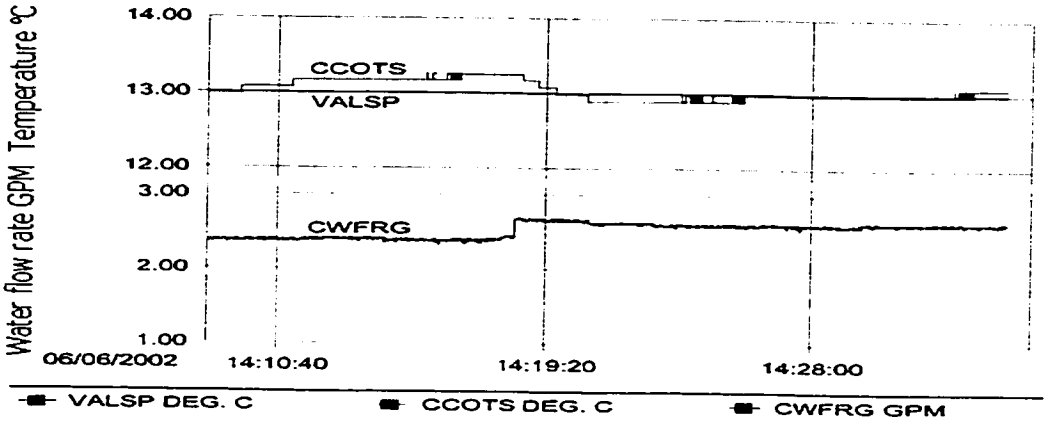

Figure 5.50 **Effect of chilled water supply temperature changes on the HIAPI control**

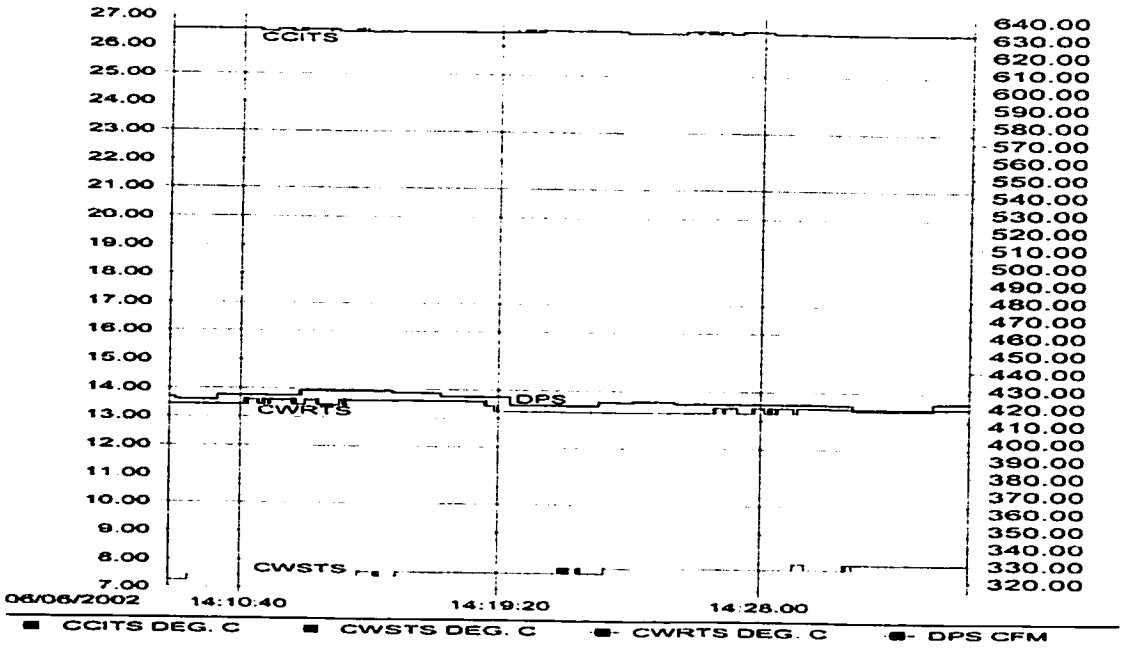

Figure 5.51 **Chilled water supply temperature profile in the HIAPI control**

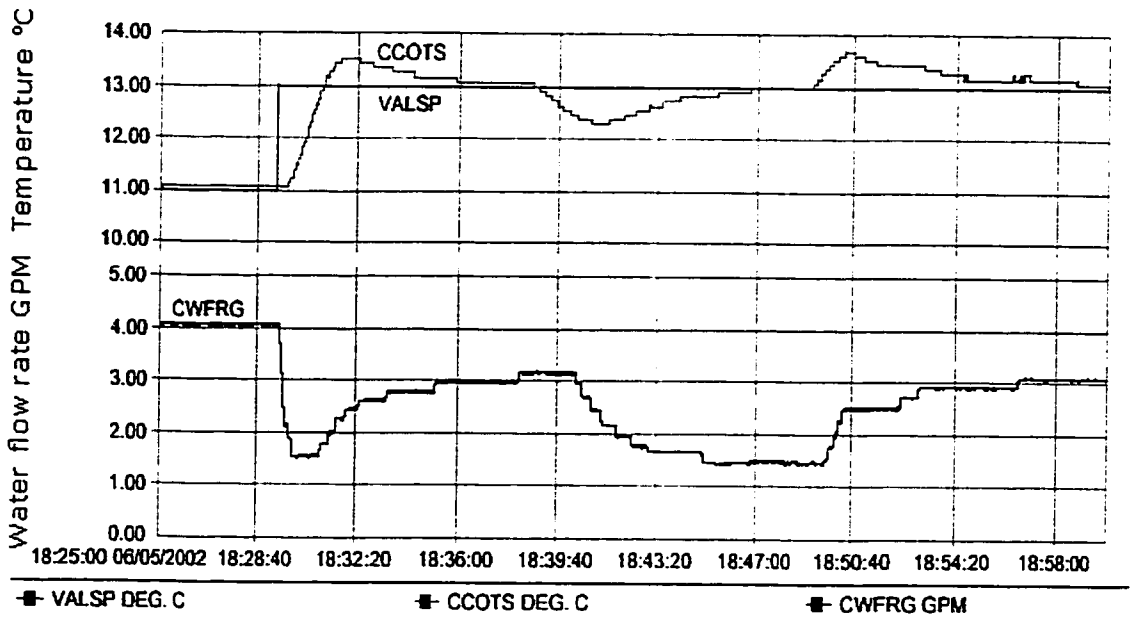

Figure 5.52 **Effect of airflow rate changes on the HIAPI control**

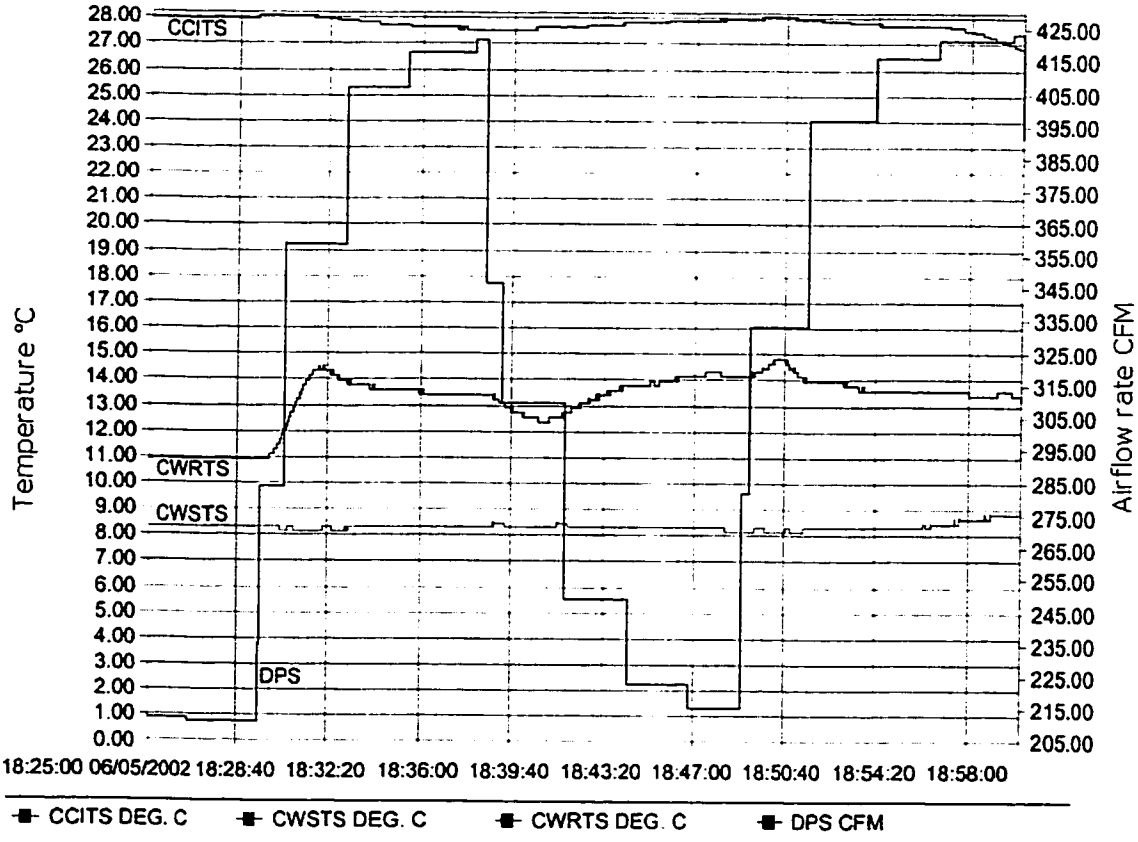

Figure 5.53 **Airflow rate changes in the HIAPI control**

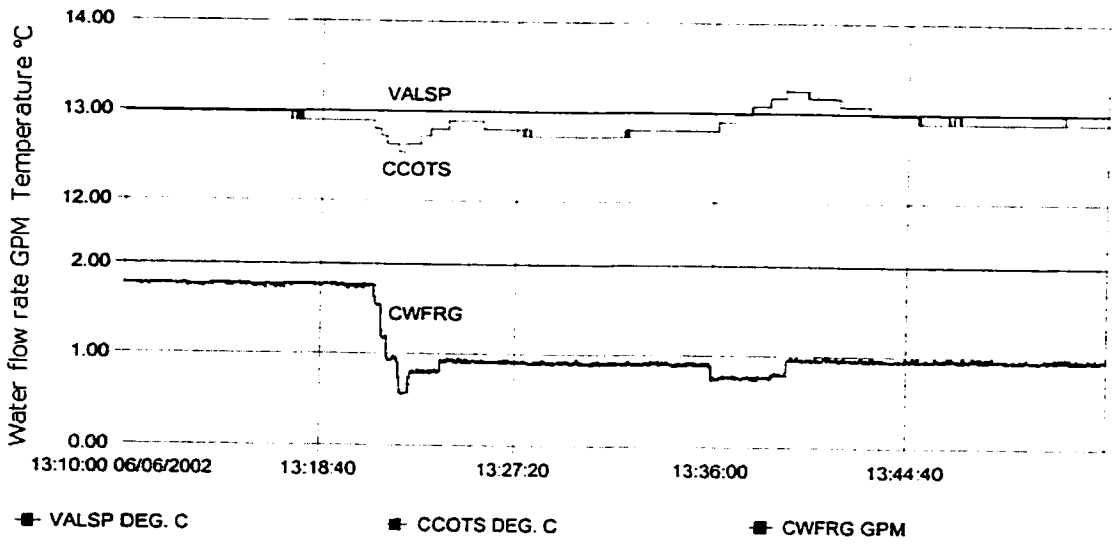

Figure 5.54 **Effect of multiple disturbances on the HIAPI control**

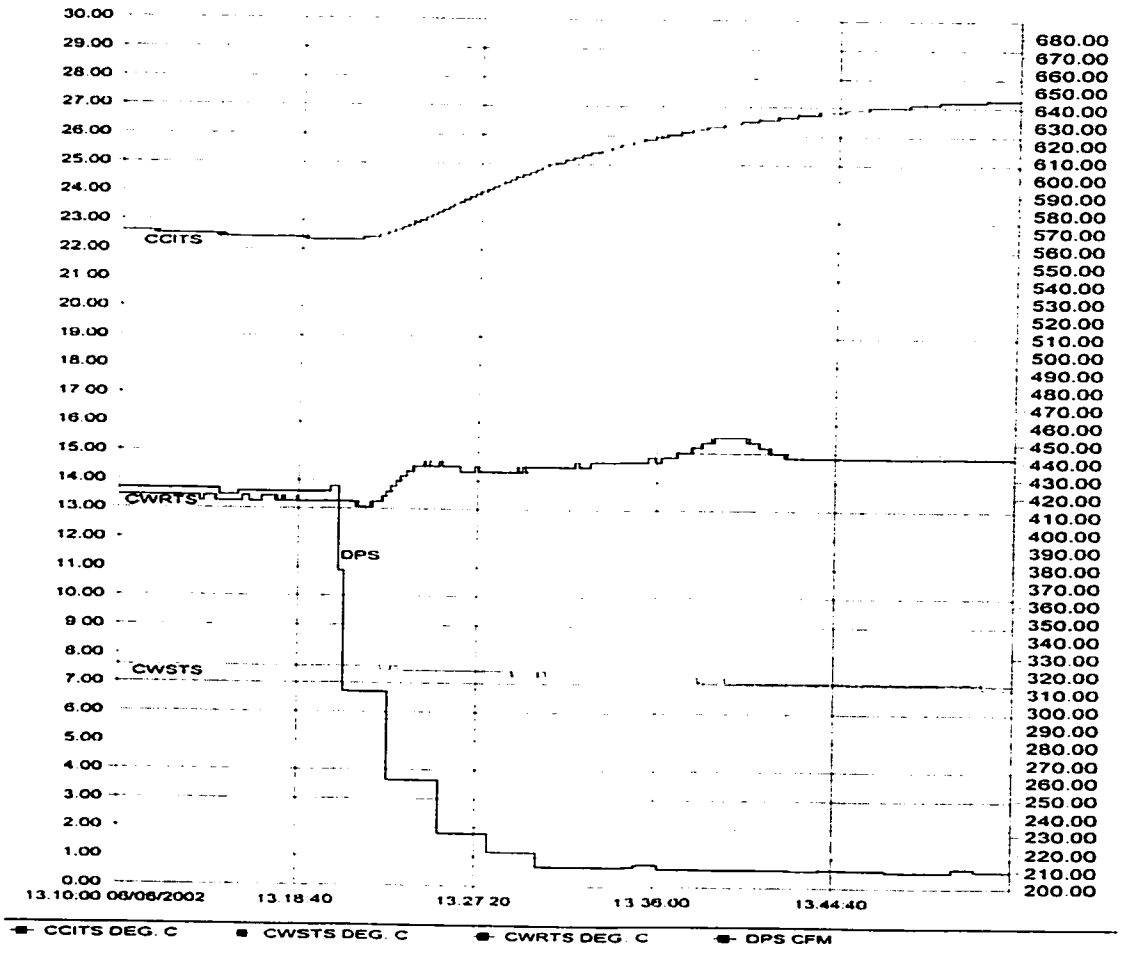

Figure 5.55 **Multiple disturbances acting on the HIAPI control**

The identified parameters (a and b of FOPDT model) and the PI parameter evolution under the effect of multiple disturbances are shown in Figure 5.56 and Figure 5.57. These results show that the HIAPI control is robust and is able to reject the effect of multiple disturbances and setpoint changes efficiently.

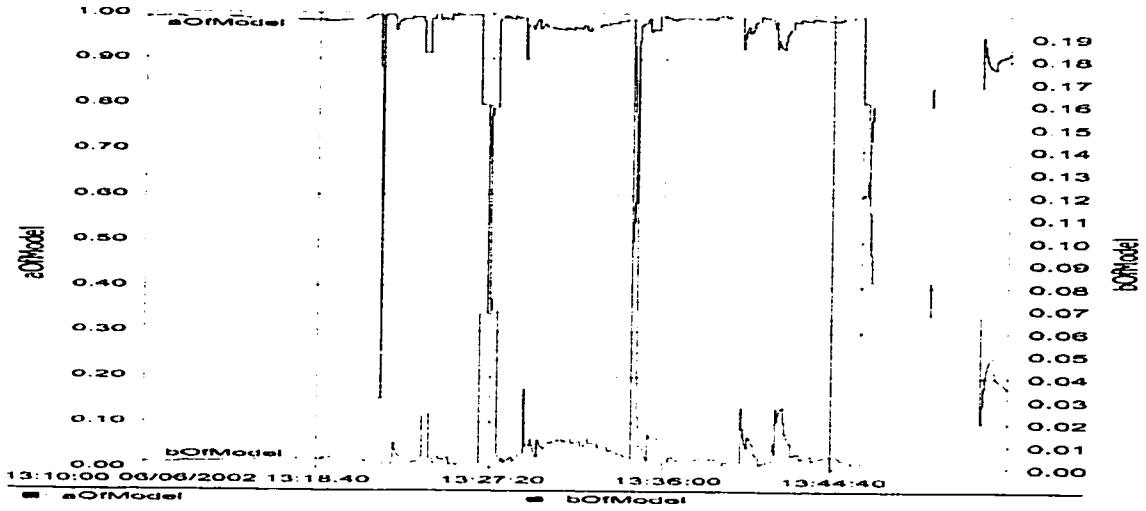

Figure 5.56 The evolution of model parameters in the HIAPI control subject to multiple disturbances

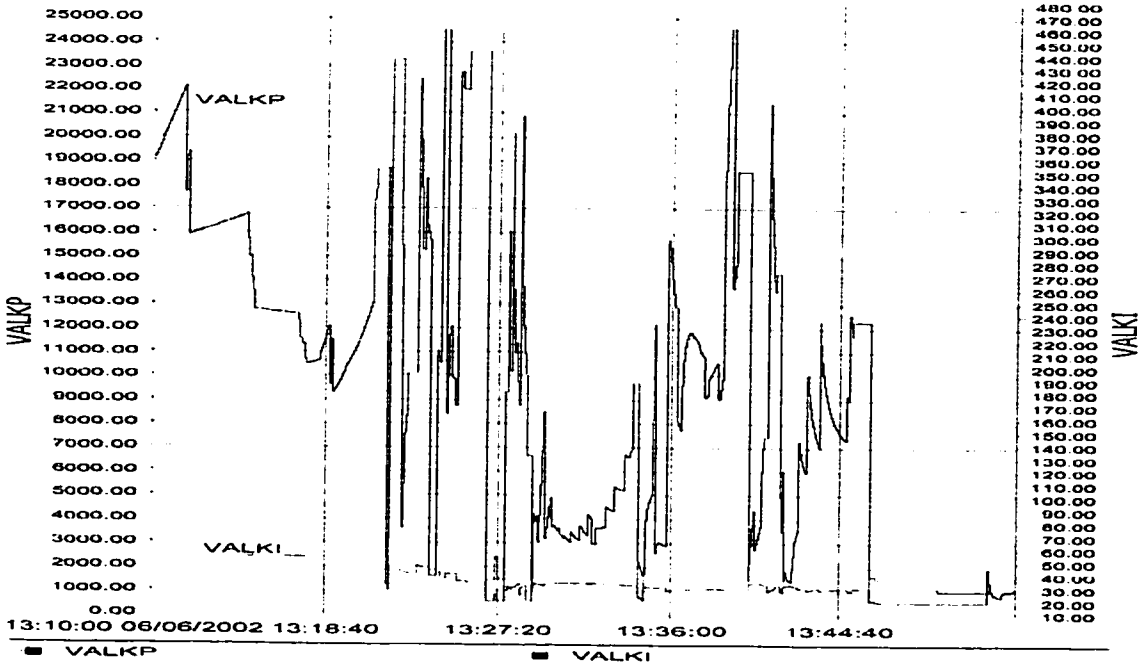

Figure 5.57 The evolution of PI parameters in the HIAPI control subject to multiple disturbances

5.5 Summary

Two real-time adaptive PI control algorithms (CMBA PI control and HIAPI control) were tested in an experimental test facility. The test included changes in setpoints, effect of multiple disturbances and control loop interactions. Results from both adaptive tuning algorithms show good temperature tracking, fast response under a wide range of operating conditions. Being robust and adaptive, both methods are well suited for HVAC control applications. Of particular importance is the simplicity of CMBA PI control. It has a simple structure and is easy to implement. It is a "model-less" adaptive control strategy with good tracking and regulation properties and is therefore well suited for HVAC control applications.

6. Conclusions and Recommendations

6.1 Conclusions

The DAT system was modelled as a second-order system, a first-order plus dead-time (FOPDT) system using available experimental input-output data sets. Also, an online recursive least squares (RLS) algorithm was developed to estimate the FOPDT system parameters. The specific conclusions from this contribution can be summarised as follows:

1. Although the dynamics of the DAT loop are known to be distributive and nonlinear, simulation results show that an FOPDT model can capture the DAT system dynamics as well as a second-order model. For online implementation the FOPDT model was found to be fast and efficient and to give acceptable responses within ±2% of the measured data.
2. From the open-loop results, it was estimated that the dead time (τ) of the DAT loop is between 36 – 40s.
3. During online implementation, τ was kept constant. As a result, only two parameters (a and b) were updated online. This reduced the number of computations and helped to free up processor memory. This is significant even when using the latest processors available in building EMC systems.
4. It was found that the use of two preceding values of the parameters θ and $\hat{\theta}$ improved convergence of the parameters a and b compared to single preceding value of θ and $\hat{\theta}$.

5. Simulation results show that the converged values of the parameters a and b ranged between $a = 0.9$ to 0.98 and $b = 0.11$ to 0.17. However, during initial few samples, following a disturbance, the parameters a and b could change between 0 to 1 somewhat rapidly before converging to their respective final values.

Three real-time adaptive PI tuning control strategies were developed. These are (i) model following adaptive (MFA) PI control, (ii) control-model based adaptive (CMBA) PI control and (iii) H_∞ adaptive PI (HIAPI) control. Simulation runs were made to compare their performance with constant gain PI control, LQR based adaptive PI control and Ziegler-Nichols based adaptive PI control. Some specific conclusions from these simulations are summarized below.

6. The output responses of MFA PI and CMBA PI controls can be optimized somewhat by choosing a proper range of $A_{(c)m}$, $B_{(c)m}$ and $\varsigma_{(m)}$ parameters.

7. The following values of $A_{(c)m}$, $B_{(c)m}$ and $\varsigma_{(m)}$ were found to give good output responses

$$A_{(c)m} = 1.0$$

$$B_{(c)m} = 0.025$$

$$\varsigma_{(m)} = 0.01 \text{ to } 0.08$$

Two to three trials were sufficient to obtain good values of $A_{(c)m}$, $B_{(c)m}$ and $\varsigma_{(m)}$.

8. Results show that both MFA and CMBA PI controls reject the effect of disturbances and changes in system parameters effectively. The same changes resulted in oscillatory response from the constant gain PI controller.

9. Using stability analysis it was found that the maximum range of the controller gain should be less than 0.525. The minimum range was arbitrarily set as 1% of the maximum gain.

10. The HIAPI control was found to give responses close to those that can be achieved with LQR based optimal adaptive PI control under similar operating conditions.

11. It was found that HIAPI control could effectively reject noise which was simulated as white noise added to the output response. In each case, the HIAPI was able to attenuate the effect of cyclic disturbances.

Of the three adaptive control strategies, two (CMBA PI and HIAPI controls) were chosen for online implementation on an existing two-zone HVAC – EMC system. The following conclusions are based on experimental results.

CMBA PI Control

12. The CMBA PI control shows good tracking ability to setpoint changes. A ±2 °C change in setpoint was achieved in 4 – 8 minutes (settling time) with overshoot ranging between 0.2 to 0.5 °C. The optimum range of control parameters were found to be

$$A_{cm} = 1.0$$

$$B_{cm} = 0.2 - 0.5$$

$$\varsigma = 100 - 360.$$

13. The effect of multiple disturbances was investigated by changing the setpoint by ±5 °C, changing inlet air temperature from 24 to 27 °C and inlet water temperature by ±2 °C and a roughly 50% change in airflow rate simultaneously.

The results show that CMBA PI control is able to reject the effect of these disturbances. The maximum overshoot during this process was ±0.3 °C.

14. The effect of control loop interactions were examined by activating two additional constant gain PI control loops in the test facility. Results show that the airflow control loop has significant impact on the DAT loop. In spite of this interaction, the CMBA PI control was effective in tracking the setpoint. The maximum overshoot during this process was +0.9 °C and the settling time was between 13 – 19 minutes.

15. The results show that even with good PI gains, a constant gain PI controller will give oscillatory output responses when subjected to change in setpoints or disturbances. On the other hand, the CMBA PI controller not only effectively rejects the same disturbances but can also stabilize an oscillatory system smoothly and effectively.

16. The CMBA PI tuning strategy continuously updates the gains in response to changes in operating conditions. During the tests, the highest and lowest gains recorded were 15000 and 4000.

17. Being simple to implement and robust to changes in operating conditions, the CMBA PI tuning strategy is well suited for HVAC systems.

HIAPI Control

The HIAPI control strategy was likewise tested for changes in setpoints, and disturbances. Some important conclusions are summarized here.

18. To achieve good output responses, a dead-time of 36 sec (9 samples) was the most effective for use in the online RLS algorithm.

19. As opposed to simulation results, the experimental results for the range of model parameters a and b (during good tracking) were

    $a = 0.65$ to 0.99

    $b = 0.05$ to 0.68.

20. The HIAPI control shows good tracking to changes in setpoints (±2 °C). The overshoot was less than 0.5 °C and settling time was between 5 – 10 minutes.
21. The change in PI gains with HIAPI control was more rapid and much higher in range compared to CMBA PI control. The highest and lowest controller gains recorded with HIAPI algorithm were 24500 and 1000 respectively.
22. The HIAPI control was able to reject the effect of multiple disturbances. The maximum overshoot during such tests was 0.8 °C with settling time of 5 – 15 minutes. As such the HIAPI control is robust to changes in operating conditions and effect of disturbances.
23. The HIAPI control was able to damp out oscillatory control loop responses effectively and in a short period of time (in about 8 minutes).
24. Much of the rapid variance which was observed in HIAPI control gains stems from the RLS algorithm. Therefore, the accuracy and stability with which the model parameters a and b are updated in the RLS algorithm play a crucial part in model-based adaptive control methods such as the HIAPI technique.
25. In spite of the higher computational effort and memory requirements, it was feasible to implement the HIAPI tuning strategy on the existing EMCS.

Thus, along with CMBA PI control, the HIAPI control has definite advantages in regulating and tracking the HVAC processes.

6.2 Recommendations

The results described in this thesis are useful for adaptive tuning of local loop controllers in HVAC systems. In order to extend this method to buildings with a large number of control loops, the following avenues of research need to be explored.

1) Develop online tuning strategies for multiple loop HVAC control processes in which all control loops can be simultaneously tuned.
2) Develop improved and implementable identification methods for HVAC processes.
3) An optimal set of parameters (B_{cm} and ς) in CMBA PI control design may exist, so it is important to develop a design strategy for choosing the best set of parameters.

References

[1] M.S. Imbabi, "Computer validation of scale model tests for building energy simulation," *International Journal of Energy Research*, Vol. 14, pp. 727-736, 1990.

[2] F.C. McQuiston, and J.D. Parker, *Heating, Ventilating and Air Conditioning: Analysis and Design*, 3d ed. John Wiley & Sons, Inc. 1988.

[3] K.J. Åström, "Theory and Applications of Adaptive Control—A Survey," *Automatica*, Vol. 19, No. 5, pp. 471-486, 1983.

[4] S.G. Brandt, "Adaptive Control Implementation Issues," *ASHRAE Transactions*, Vol. 92, Part 2B, pp. 211-219, 1986.

[5] K.I. Krakow, S. Lin, and Z.S. Zeng, "Temperature and humidity control during cooling and dehumidifying by compressor and evaporator fan speed variation," *ASHRAE Transactions*, Vol. 101, Part 1, pp292-pp304, 1995.

[6] K.I. Krakow, S. Lin, and Z.S. Zeng, "Analytical determination of PID coefficients for temperature and humidity control during cooling and dehumidifying by compressor and evaporator fan speed variation," *ASHRAE Transactions*, Vol. 101, Part 1, pp343-pp354, 1995.

[7] K.I. Krakow, S. Lin, "PI control of fan speed to maintain constant fan discharge pressure," *ASHRAE Transactions*, Vol. 101, Part 2, pp398-pp407, 1995.

[8] G.M. Hussein, "PI control of air temperature and humidity with a chilled water system," *Master's thesis, Concordia University, Montreal, Quebec, Canada*, 1996.

[9] M. Corless and G. Leitmann, "Adaptive Control for Uncertain Dynamical Systems," in: A. Blaquiere and G. Leitmann, (eds.), *Dynamical Systems and Microphysics: Control Theory and Mechanics*, Academic press, Orlando, 1984.

[10] J.F. Kreider, and J.S. Haberl, "Predicting hourly building energy use: The great energy predictor shootout – overview and discussion of results," *ASHRAE Transactions*, Vol. 100, Part 2, pp.1104-1118, 1994.

[11] D.W. Clarke, C. Mohtadi, and P.S. Tuffs, "Generalized Predictive Control – Part I. The Basic Algorithm; Part II. Extensions and Interpretations," *Automatica*, Vol. 23, No. 2, pp.137-160, 1987.

[12] A.L. Dexter, "Self-tuning control algorithm for single-chip microcomputer implementation," *Proc. IEE*, Vol. 130, No. 5, pp. 255-260, 1983.

[13] M. Zaheer-uddin, "Design and commissioning of a two-zone variable air volume HVAC test facility," *Internal Report, Department of Building, Civil and Environmental Engineering, Concordia University*, 1999.

[14] G.F. Franklin and J.D. Powell, *Digital Control of Dynamic Systems*, Addison-Wesley, Reading, MA, 1980.

[15] G.R. Zheng, and M. Zaheer-uddin, "Discharge air system: modelling and optimal control," *International Journal of Energy Research*, Vol. 23, pp.727-738, 1999.

[16] G. Shavit, and S.G. Brandt, "The Dynamic Performance of Discharge Air-Temperature System with a P-I controller," *ASHRAE Transactions*, Vol.88, Part2, pp.826-838, 1982.

[17] G. Shavit, "Tuning of PID DDC Controllers," *American Control Conference*, Vol.3, pp.3065-3069, 1994.

[18] J.E. Seem, "A Pattern Recognition Adaptive Controller," *IFAC 13th Triennial World Congress, San Francisco*, pp.121-126, 1996.

[19] G. Singh. "Adaptive Control of a Multizone Fan-Coil Heating System," *Master's thesis, Concordia University, Montreal, Quebec, Canada*, 1993.

[20] I. Jette, "PI-Control in Dual Duct Systems: A Study on Manual Tuning and Control Loop Interaction," *Master's thesis, Concordia University, Montreal, Quebec, Canada*, 1997.

[21] M. Kasahara, T. Yamazaki, Y. Kuzuu, Y. Hashimoto, K. Kamimura, T. Matsuba, and S. Kurosu, "Stability Analysis and Tuning of PID Controller in VAV Systems," *ASHRAE Transactions*, Vol. 107, Part 1, pp. 285-296, 2001.

[22] Q.G. Wang, T.H. Lee, H.W. Fung, Q. Bi, and Y. Zhang, "PID Tuning for Improved Performance," *IEEE Transactions on Control Systems Technology*, Vol. 7, No. 4, pp.457-465, July 1999.

[23] M. Kasahara, T. Matsuba, Y. Kuzuu, T. Yamazaki, Y. Hashimoto, K. Kamimura, and S. Kurosu, "Design and Tuning of Robust PID Controller for HVAC Systems," *ASHRAE Transactions*, Vol. 105, Part 2, pp. 154-166, 1999.

[24] K.I. Krakow, "Sampling Interval Magnitude Relation to Digital PI Control System Performance," *ASHRAE Transactions*, Vol. 104, Part 2, pp. 246-256, 1998.

[25] J.E. Seem, "Implementation of a New Pattern Recognition Adaptive Controller Developed Through Optimization," *ASHRAE Transactions*, Vol. 103, Part 1, pp. 494-506, 1997.

[26] C.G. Nesler, "Automated Controller Tuning For HVAC Applications," *ASHRAE Transactions*, Part 2B, Vol. 92, pp. 189-201, 1986.

[27] C.G. Nesler, "Adaptive Control of Thermal Processes in Buildings," *IEEE Control Systems Magazine*, Vol. 6, No. 4, pp. 9-13, 1986.

[28] K.J. Åström, T. Hagglund, and A. Wallenborg, "Automatic Tuning of Digital Controllers with Applications to HVAC Plants," *Automatica*, Vol. 29, No. 5, pp. 1333-1343, 1993

[29] Y.H. Chen, K.M. Lee, and W.J. Wepfer, "Adaptive Robust Control Scheme Applied to a Single-zone HVAC System," *Proceedings of the 1990 American Control Conference*, Vol. 2, pp. 1076-1081, May 1990.

[30] O. Ahmed, J.W. Mitchell, and S.A. Klein, "Feedforward-Feedback Controller Using General Regression Neural Network (GRNN) for Laboratory HVAC System: Part I – Pressure Control," *ASHRAE Transactions*, Vol. 104, Part 2, pp. 613-625, 1998.

[31] O. Ahmed, J.W. Mitchell, and S.A. Klein, "Feedforward-Feedback Controller Using General Regression Neural Network (GRNN) for Laboratory HVAC System: Part II – Temperature Control – Cooling," *ASHRAE Transactions*, Vol. 104, Part 2, pp. 626-634, 1998.

[32] O. Ahmed, J.W. Mitchell, and S.A. Klein, "Feedforward-Feedback Controller Using General Regression Neural Network (GRNN) for Laboratory HVAC System: Part III – Temperature Control – Heating," *ASHRAE Transactions*, Vol. 104, Part 2, pp. 635-643, 1998.

[33] E. Jeannette, K. Assawamartbunlue, P.S. Curtiss and J.F. Kreider, "Experimental Results of a Predictive Neural Network HVAC Controller," *ASHRAE Transactions*, Vol. 104, Part 2, pp. 192-197, 1998.

[34] A.L. Dexter, and P. Haves, "A Robust Self-Tuning Predictive Controller for HVAC Applications," *ASHRAE Transactions*, Vol. 95, Part. 2, pp. 431-438, 1989.

[35] M. Kasahara, T. Matsuba, Y. Hashimoto, I. Murasawa, A. Kimbara, K. Kamimura, and S. Kurosu, "Optimal Preview Control for HVAC System," *ASHRAE Transactions*, Vol. 104, Part 1A, pp. 502-513, 1998.

[36] M. Zaheer-uddin, and G.R. Zheng, "Multistage Optimal Operating Strategies for HVAC Systems," *ASHRAE Transactions*, Vol. 107, Part 2, pp. 346-352, 2001.

[37] B. Arguello-Serrano and M. Velez-Reyes, "Nonliear Control of a Heating, Ventilating, and Air Conditioning System with Thermal load Estimation," *IEEE Transaction on Control Systems Technology*, Vol. 7, No. 1, pp.56-63, 1999.

[38] A.O. Wallenborg, "A New Self-tuning Controller for HVAC Systems," *ASHRAE Transactions*, Vol. 97, Part 1, pp. 19-25, 1991.

[39] W. Tan, L. Liu, and P.K.S. Tam, "PID tuning based on loop-shaping H_∞ control," *IEE Proc. Control Theory Appl.*, Vol. 145, No. 6, pp. 485-490, November 1998.

[40] K.J. Åström, and T. Hagglund, "Automatic Tuning of Simple Regulators with Specifications on Phase and Amplitude Margins," *Automatica*, Vol. 20, No. 5, pp. 645-651, 1984.

[41] W.K. Ho, C.C. Hang and L.S. Cao, "Tuning of PID Controllers Based on Gain and Phase Margin Specifications," *Automatica*, Vol.31, No. 3, pp. 497-502, 1995.

[42] Y. Nishikawa, N. Sannomiya, T. Ohta, and H. Tanaka, "A Method for Auto-tuning of PID Control Parameters," *Automatica*, Vol. 20, No. 3, pp. 321-332, 1984.

[43] S. Majhi, and D.P. Atherton, "Autotuning and controller design for processes with small time delays," *IEE Proc.-Control Theory Appl.*, Vol. 146, No.5, September 1999.

[44] V. Peterka, "Predictor-based Self-tuning Control," *Automatica*, Vol. 20, No. 1, pp. 39-50, 1984.

[45] K.J. Åström and B. Wittenmark, *Adaptive Control*, Addison-wesley publishing company, 1989.

[46] Landis and Staefa, "System 600 Apogee Modular Building Controller and Remote Building Controller Owners Manual (125 – 1992)" Rev. 5.

Appendix A

Ziegler-Nichols for FOPDT models in discrete-time systems

Assume that we have designed a PID controller for a first-order plus dead-time model of continuous system between u to y as shown in Figure 3.7 and the controller can be expressed as follow.

$$K(s) = -K_p(1 + \frac{1}{T_i s} + T_d s) \tag{A-1}$$

Using Tustin's method to obtain the discrete equivalent, we obtain the following expression of controller in the digital system for a plant can be approximately expressed as a first-order plus dead-time model.

$$K_T(z) = -K_p \frac{\left(1 + \frac{T}{2T_i} + \frac{2T_d}{T}\right)z^2 + \left(\frac{T}{T_i} - \frac{4T_d}{T}\right)z + \frac{T}{2T_i} + \frac{2T_d}{T} - 1}{(z-1)(z+1)} \tag{A-2}$$

Using $K_d(z)$ (4.1) to substitute for $K_T(z)$, we must have

$$\lim_{z\to 1}(z-1)K_d(z) = \lim_{z\to 1}(z-1)K_T(z) \quad \rightarrow \quad K_{id} = -\frac{K_p}{T_i} \tag{A-3}$$

for the steady state equivalent, and

$$-K_p\left(1 + \frac{T}{2T_i} + \frac{2T_d}{T}\right) = K_{pd} + \frac{K_{id}T}{2} \tag{A-4}$$

$$-K_p\left(\frac{T}{T_i} - \frac{4T_d}{T}\right) = K_{id}T \tag{A-5}$$

$$-K_p\left(\frac{T}{2T_i} + \frac{2T_d}{T} - 1\right) = \frac{K_{id}T}{2} - K_{pd} \tag{A-6}$$

→

$$T_d = 0 \text{ and } K_{pd} = -K_p \tag{A-7}$$

This means that the controller for the first-order plus dead-time models of continuous systems must be PI controller in using $K_d(z)$ (4.1) to substitute for $K_T(z)$ (A-2). Therefore,

$$K(s) = -K_p(1 + \frac{1}{T_i s}) \tag{A-8}$$

The PI parameters according to Ziegler-Nichols step response method for the first-order plus dead-time models of continuous systems can be expressed as follow [45].

$$K_p = \frac{0.9}{k_s \Theta} \tag{A-9}$$

and $$T_i = 3\tau \tag{A-10}$$

where $$\Theta = \frac{\tau}{T_s}. \tag{A-11}$$

From equations (3.3) to (3.5), (A-3), (A-7), and (A-9) to (A-11), we have

$$K_{pd} = \frac{0.9(1-a)}{bl\ln(a)} \tag{A-12}$$

$$K_{id} = \frac{K_{pd}}{3lT} \tag{A-13}$$

Appendix B

LQR optimal control for FOPDT models in discrete-time systems

Assume that the considered plant can be represented as in Figure 3.8. Then the feedback control system with a PI controller $K_d(z)$ (4.1) can be obtained as follows.

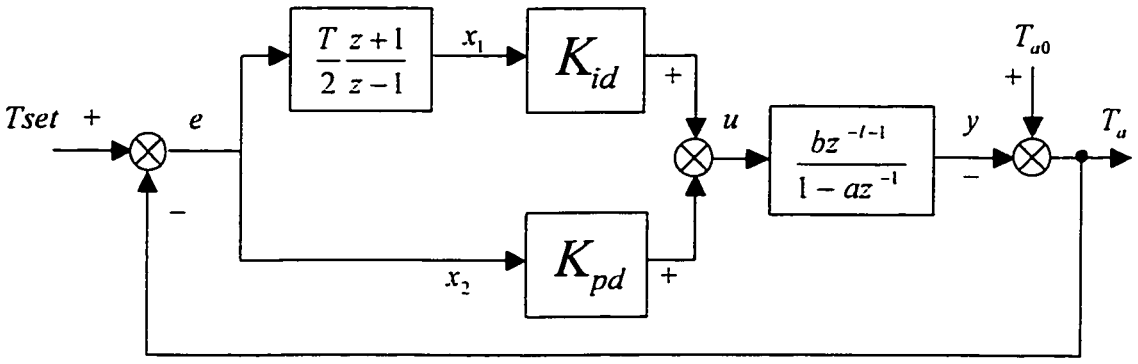

Figure B.1 State expression for PI controller $K_d(z)$

Let $K_f = [-K_{id} \quad -K_{pd}]$ (B-1)

we have $u = -K_f x$ (B-2)

where $x = [x_1 \quad x_2]^T$. (B-3)

The LQR optimal control problem becomes

$$\min_{K_f} J = \min_{K_f} \frac{1}{2}\sum_{k=0}^{N}\left[x^T[k]Qx[k] + u^T[k]Ru[k]\right] \tag{B-4}$$

where $x_2[k] = e[k]$, (B-5)

$$x_1[k] = x_1[k-1] + \frac{T}{2}\left(e[k] + e[k-1]\right),\ x_1[-1] = 0,\ e[-1] = 0 \tag{B-6}$$

$$e[k] = Tset[k] - T_a[k] \tag{B-7}$$

$$T_a[k] = T_{a0} - y[k] \tag{B-8}$$

$$y[k] = \begin{cases} 0 & k \le l \\ ay[k-1] + bu[k-l-1] & k > l \end{cases} \tag{B-9}$$

and $$u[k] = K_{pd} x_2[k] + K_{id} x_1[k]. \tag{B-10}$$