APPLICATION OF MODERN CONTROL TECHNIQUES TO A DISTILLATION COLUMN

Volume 2

TAOFEEK OLADIRAN FOLAMI (BSc, MSc)

Doctor of Philosophy

THE UNIVERSITY OF ASTON IN BIRMINGHAM

April 1989

THE UNIVERSITY OF ASTON IN BIRMINGHAM

APPLICATION OF MODERN CONTROL TECHNIQUES TO A DISTILLATION COLUMN

Taofeek Oladiran Folami

PhD 1989

SUMMARY

Modern control techniques have been applied to a distillation column. Three control techniques were selected for evaluation. These are; a decoupling and disturbance rejection control scheme; an estimator aided control techniques using a Kalman filter; and an implicit generalised minimum variance self tuning control. A 10 tray pilot scale binary distillation column, interfaced with a microcomputer, was used for investigation of the process control techniques. A non-linear model of the column was developed. The reliability of this model was demonstrated. The model was therefore used for the design, analysis and screening of control systems for the pilot plant distillation column.

The results of extensive simulations on linearised state variable models of the column simulator demonstrate that the decoupling and disturbance rejection controller works in the presence of load disturbances and setpoint changes. The proper choice of the values of a diagonal matrix in the precompensator of the controller required for accurate setpoint tracking has also been shown. By analogy with PI control, integral and derivative modes have been introduced into the controller to equip it with the ability to remove offsets. Simulation results demonstrate that the sensitivity of the controller to non-linear effects makes the controller inoperable on the column simulator, as well as on the pilot plant. Therefore, the use of an adaptive form of the controller is necessary to compensate for the non-linear effects and other model errors for on-line application to be practical on the pilot plant.

On-line implementation of the Kalman filter algorithm using a linear state variable model of the column simulator as the filter model, was not possible because of the large memory requirement of the software, long execution time and the inability to produce satisfactory estimates of all the tray compositions.

Simulated and experimental studies for both single temperature control and dual temperature control of the distillation column, demonstrated that self tuning control can provide tighter control of the products of distillation columns than PI control.

An algorithm, called the Simplified Correction (SPC) method, has been implemented to prevent the parameters of a self tuning controller from reaching unsatisfactory values when the closed loop system is not sufficiently excited. Simulations show that the SPC can provide significant improvements even when only a subset of the controller parameters are prevented from attaining bad values.

The findings in this work verify the degrading effects that model errors have on controller performance. Areas for future work have been suggested in the case of the on-line implementation of the control schemes selected in this work.

Key Words: Self tuning control, Decoupling and Disturbance Rejection control, Parameter Correction, Simplified parameter Correction, Distillation Column.

This thesis is dedicated to my mother, Rashidat Olayinka Folami, to my late brother, Salmon Ayodeji Folami, and to the rest of my family; Sabitu Olaore Folami, Muyibat Wonuola Folami, Alhaja Nofisat Idewu, Sabitu Olayinka, Abubakar Oladotun, Uthman Oluwakemi, Hakeem Oludare, Kamal Ademola, Sarata, Yewande, Bolakale, Rotimi

Acknowledgements

I wish to thank Dr J.P. Fletcher for his supervision of this research work, and for his patience and valuable suggestions on this thesis.

I wish to Dr A.P.H. Jordan for his assistance and encouragement during the course of the research work.

I would also like to thank

Dr B. Gay for allowing me to use the computing facilities at the Depertment of Computer Science α Applied Mathematics;

The University of Aston in Birmingham for financing me and for providing all necessary equipment;

Diane Stretton for her typing of part of this thesis;

Phillipa Forde for her help in preparation of this thesis;

My friends, Numfor Ajongwen, Sama Nwana, Mohammed Sarki and all my colleagues in the Department of Chemical Eingineering and the Department of Computer Science and Applied Mathematics for their encouragement throughout the course of this work.

Finally, I should thank my family for their support throughout my education.

List of Contents
Volume 1

List of Contents
Volume 2

List of Figures
Volume 1

List of Figures
Volume 2

List of Tables
Volume 1

List of Tables
Volume 2

List of Plates
Volume 1

This page is left intentionally blank

CHAPTER EIGHT

The design of self tuning controllers for the distillation column

8.1 Design of the controllers

In this work implicit single input single output (SISO) and multiple input multiple output (MIMO) self tuning controllers based on the design of Clarke and Gawthrop (157) (see section 2.9.4) have been developed for the distillation column. This chapter gives the details of these self tuning controller designs as well as a description of the computer programs written for their implementation by simulation and real-time application.

The distillation column is represented as a process with two inputs and two outputs as shown in Figure 8.1 below.

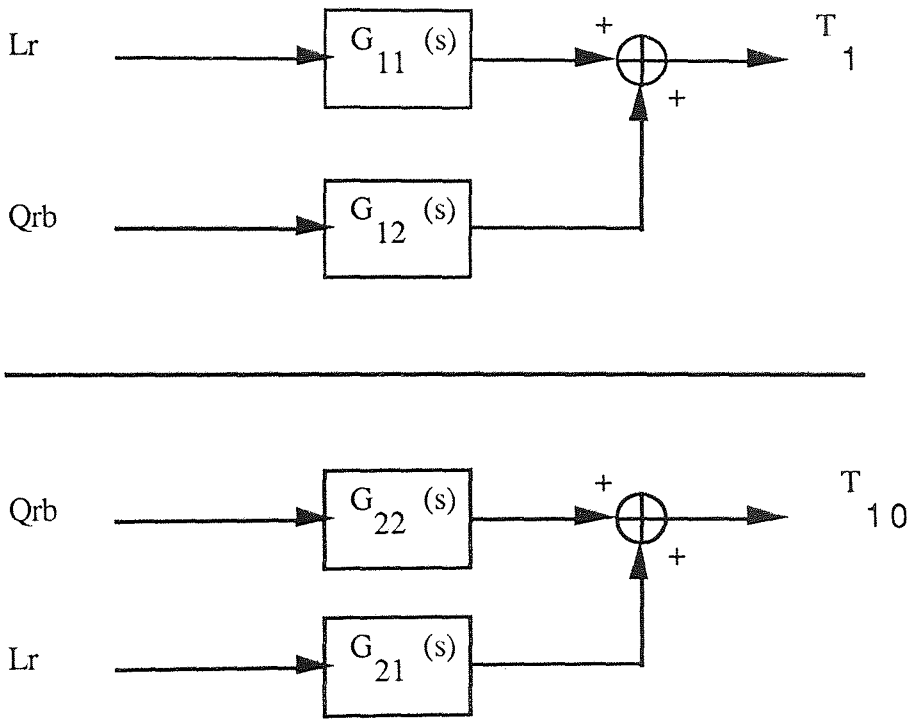


Figure 8.1 Block diagram representation of the distillation column.

The input-output relationships, $\mathbf{y}(s) = \mathbf{G}(s)\mathbf{u}(s)$, representing this diagram are

$$T_1(s) = G_{11}(s)Lr(s) + G_{12}(s)Qrb(s) \qquad 8.1a$$

$$T_{10}(s) = G_{21}(s)Lr(s) + G_{22}(s)Qrb(s) \qquad 8.1b$$

indicating that a change in either input Lr or Qrb will affect both T_1 and T_{10}. The G_{ij}'s are first order transfer functions relating the two outputs and the two inputs. Thus, $\mathbf{y}^T = (T_1, T_{10})$ and $\mathbf{u}^T = (Lr, Qrb)$.

8.1.1 Model structure for SISO case

As mentioned in Chapter 2 Section 2.9.2, for a SISO case, the system is modelled by the linear discrete time equation

$$A(z^{-1})\, y(t) = B(z^{-1})\, u(t-k) + C(z^{-1})\xi(t) + d(t) \qquad 8.2$$

where A, B and C are polynomials in z domain,

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_n z^{-n}$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + \ldots + b_m z^{-m}$$

$$C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \ldots + c_n z^{-n}$$

Here, the k is an integer representing the process time delay in terms of the number of sampling intervals, $\xi(t)$ is the random disturbance inputs such as noise and measurement errors, d(t) is the constant offset term to account for unmeasured load disturbances, inaccurate initial values and noise with a non-zero mean. The n and m are positive integers representing the orders of the A and B polynomials, respectively.

The assumptions made in the controller design are, n = 1, m = 0 k = 1 and $C(z^{-1}) = 1$. The SISO model then becomes

$$y(t) = -a_1 y(t-1) + b_0 u(t-1) + \xi(t) + d(t) \qquad 8.3$$

8.1.2 Model structure for MIMO case

In this work a 2 input 2 output MIMO system was considered

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

8.4a

The **u**, **y**, **d**, and ξ are all vectors; $\mathbf{y}(t) = (y_1(t), y_2(t))^T$, $\mathbf{u}(t) = (\mathbf{u}_1(t), \mathbf{u}_2(t))^T$, $\xi(t) = (\xi_1(t), \xi_2(t))^T$, $\mathbf{d}(t) = (\mathbf{d}_1(t), \mathbf{d}_2(t))^T$. The **P**-canonical form was considered in this work so that the **A** is diagonal. As discussed in Section 2.9.12, using the **P** -canonical form allows the MIMO system to be formulated as multiple input single output (MISO) sub-systems so that each sub-system can be treated independently. Thus

$$\mathbf{A} = \mathrm{diag}(\mathbf{A}_1, \mathbf{A}_2)$$

$$\mathbf{B} = \begin{bmatrix} B_{11} & z^{k_{11} - k_{12}} B_{12} \\ z^{k_{22} - k_{21}} B_{21} & B_{22} \end{bmatrix}$$

$$\mathbf{C} = \mathrm{diag}(\mathbf{C}_1, \mathbf{C}_2)$$

k_{ii} is the delay between $\mathbf{y}_i$ and $\mathbf{u}_i$ and k_{ij} is the delay between $\mathbf{y}_i$ to $\mathbf{u}_j$; it is assumed that $k_{ij} > k_{ii} > 1$. Equation 8.4 is decomposed into 2 MISO subsystems, described by

$$\mathbf{A}_i y_i(t) = \mathbf{B}_{ii}\mathbf{u}_i(t-k_{ii}) + \mathbf{B}_{ij}\mathbf{u}_j(t-k_{ij}) + \mathbf{C}_i\xi_i(t) + \mathbf{d}_i(t) \qquad 8.4b$$

Similar to the SISO case, deg ($\mathbf{A}$) = 1, deg ($\mathbf{B}$) = 0 and $\mathbf{C}(z^{-1}) = I$ in the the design of the MIMO STC's. Three cases were considered as examples. The first was where the time delays between the inputs **u** and each output in **y** are the same. For the assumption of unity delay between each input and output pair, the delay matrix D_k is

$$\mathbf{D}_k = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad 8.5$$

The second example is when the time delays are different, but the delays on the diagonal on each row of D_k are the smallest. For example,

$$\mathbf{D}_k = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \qquad 8.6$$

This means that the delay between output $\mathbf{y}_1(t)$ and changes in input $\mathbf{u}_2(t)$ and between output $\mathbf{y}_2(t)$ and changes in input $\mathbf{u}_1(t)$ are 2 sample intervals.

In these two cases the effects on each input, $u_i(t)$, on both outputs are included in the model. The third case is where the effects of an input $u_i(t)$ only affects output $y_i(t)$, in effects uncoupling the loops.

The corresponding MIMO system models for the cases discussed above are as follows:

1) For the delay matrix in Equation 8.4 the model is

$$y_1(t) = -a_1^{11} y_1(t-1) + b_0^{11} u_1(t-1) + b_0^{12} u_2(t-1) + \xi_1(t) + d_1(t)$$
$$y_2(t) = -a_1^{22} y_2(t-1) + b_0^{21} u_1(t-1) + b_0^{22} u_2(t-1) + \xi_2(t) + d_2(t) \quad 8.7$$

This model is a set of MISO sub-systems.This model will be called model MD1

2) For the delay matrix in Equation 8.5 the MIMO model is

$$y_1(t) = -a_1^{11} y_1(t-1) + b_0^{11} u_1(t-1) + b_1^{12} u_2(t-2) + \xi_1(t) + d_1(t)$$
$$y_2(t) = -a_1^{22} y_2(t-1) + b_1^{21} u_1(t-2) + b_0^{22} u_2(t-1) + \xi_2(t) + d_2(t) \quad 8.8$$

This model is also a set of MISO sub-systems and will be called model MD2

3) For third case where the effects of u_1 on y_2 and u_2 on y_1 are ignored, two separate or independent single variable models result. For unity delays between each u_i and y_i pair, these models are

$$y_1(t) = -a_1^{11} y_1(t-1) + b_0^{11} u_1(t-1) + \xi_1(t) + d_1(t)$$
$$y_2(t) = -a_1^{22} y_2(t-1) + b_0^{22} u_2(t-1) + \xi_2(t) + d_2(t) \quad 8.9$$

This model will be called model MD3.

8.2 Control law synthesis

8.2.1 The SISO case

The design procedure for the SISO self tuning controller used in this work has been done in Section 2.9.4. In summary, the objective function that is minimised the given by:

$$J_1 = \left(\left(P(z^{-1})\, y(t-k) - R(z^{-1})\, w(t) \right)^2 + [Q'(z^{-1}) u(t)]^2 \right) \quad 8.10$$

where w(t) is the set point and the $\mathbf{P}(z^{-1})$, $\mathbf{Q}'(z^{-1})$ and $\mathbf{R}(z^{-1})$ are user specified transfer functions of the form

$$\mathbf{P}(z^{-1}) = \mathbf{P}_N(z^{-1}) / \mathbf{P}_D(z^{-1}) \qquad 8.11.$$

The $\mathbf{P}(z^{-1})$, $\mathbf{Q}'(z^{-1})$ and $\mathbf{R}(z^{-1})$ are the controller parameters that are pre-specified to give the desired closed loop properties. To recap, the **P** is the output weighting which provides the controller with model following features; the **Q'** penalises control action to reduce excessive control activity; **R** is the set point filter that has the effect of reducing over shoots of the output variables after setpoint changes. The condition $\mathbf{P}(1) = \mathbf{R}(1)$ must be satisfied to avoid offsets.

The self tuning controller is based on an optimal k-step ahead prediction of the weighted output $\mathbf{P}(z^{-1})y^*(t + k)$ where k is the process delay. The predictor equation has the form

$$\mathbf{P}(z^{-1})\, y^*(t + k) = \mathbf{F}(z^{-1})y'(t) + \mathbf{G}(z^{-1})u(t) + \xi(t) + d \qquad 8.12$$

where $y'(t) = y(t)/\mathbf{P}_D$. The superscript * indicates prediction. The control law that minimises Equation 8.10 w.r.t u(t) is

$$\mathbf{P}(z^{-1})\, y^*(t + k) - \mathbf{R}w(t) + \mathbf{Q}u(t) = 0 \qquad 8.13$$

where $\mathbf{Q} = \mathbf{Q}'\mathbf{Q}'(0)/\mathbf{G}(0)$. Substituting Equation 8.12 into Equation 8.13 the positional controller equation

$$\mathbf{F}\, y'(t) + \mathbf{G}\, u(t) + d - [\mathbf{R}\, w(t) - \mathbf{Q}\, u(t)\,] = 0 \qquad 8.14$$

from which the controller output can be calculated as

$$u(t) = [\mathbf{G} + \mathbf{Q}]^{-1}\, [- \mathbf{F}\, y'(t) + \mathbf{R}\, w(t) - d] \qquad 8.15$$

As explained in Section 2.9.5 using scalar **Q** ($\mathbf{Q} = \lambda$) in Equation 8.15, introduces offset commonly known as lamda offset. If the removal of lamda offset is required, Equation 8.14 becomes

$$\mathbf{F}\, y'(t) + \mathbf{G}\, u(t) + d - [\mathbf{R}\, w(t) - \mathbf{Q}\Delta_1 u(t)\,] = 0 \qquad 8.16$$

and Equation 8.15 becomes

$$u(t) = [\mathbf{G} + \mathbf{Q}]^{-1}\, [- \mathbf{F}\, y'(t) + \mathbf{R}\, w(t) + \mathbf{Q}u(t\text{-}1) - d] \qquad 8.17$$

The k - incremental version of Clarke et al. (21) was obtained as presented in Section 2.9.9 (equations 2.97 - 2.100). The k - incremental predictor model is

$$\mathbf{P}(z^{-1})\Delta_k y^*(t+k) = \mathbf{F}\Delta_k y'(t) + \mathbf{G}\Delta_k u(t) \qquad 2.98$$

which, when re-arranged, gives

$$\mathbf{P}(z^{-1})y^*(t+k) = \mathbf{P}(z^{-1})y^*(t) + \mathbf{F}\Delta_k y'(t) + \mathbf{G}\Delta_k u(t) \qquad 2.99$$

Since the control law is the same as that used by the positional form, the controller output is computed as

$$\Delta_k u(t) = [\ \mathbf{G} + \mathbf{Q}]^{-1}\ [-\mathbf{P}(z^{-1})y^*(t) - \mathbf{F}\Delta_k\ y'(t) + \mathbf{R}\ w(t)] \qquad 2.100$$

which is

$$\Delta_k u(t) = [\ \mathbf{G} + \mathbf{Q}]^{-1}\ [-\mathbf{P}(z^{-1})y\ (t) + ep(t) - \mathbf{F}\Delta_k\ y'(t) + \mathbf{R}\ w(t)]$$

8.18

In practice $\mathbf{P}(z^{-1})y^*(t)$ is replaced by $\mathbf{P}(z^{-1})y(t)$ and a proxied estimation error to enhance accuracy, but simply letting $\mathbf{P}(z^{-1})y^*(t) = \mathbf{P}(z^{-1})y(t)$ is known to yield good results (Tham et al. (213)). In this case

$$\Delta_k u(t) = [\ \mathbf{G} + \mathbf{Q}]^{-1}\ [-\mathbf{P}(z^{-1})y\ (t) - \mathbf{F}\Delta_k\ y'(t) + \mathbf{R}\ w(t)] \qquad 8.19$$

Note that replacing $\mathbf{P}(z^{-1})y^*(t)$ by $\mathbf{P}(z^{-1})y(t)$ in Equation 2.99 and expanding gives

$$\mathbf{P}(z^{-1})y^*(t+k) = \mathbf{P}(z^{-1})y(t) + \mathbf{F}y'(t) + \mathbf{G}u(t) - \mathbf{P}(z^{-1})y^*(t)$$

8.20

since $\mathbf{P}y^*(t) = \mathbf{F}y'(t-k) + \mathbf{G}u(t-k)$. Thus, substituting Equation 8.20 into Equation 8.13 and re-arranging, gives the controller output as

$$u(t) = [\ \mathbf{G} + \mathbf{Q}]^{-1}\ [-\mathbf{F}(y'(t) + \mathbf{R}w(t) - \{\mathbf{P}(z^{-1})y(t) - \mathbf{P}(z^{-1})y^*(t)\}]$$

8.21

The terms within {.} in Equation 8.21 represents the prediction error, ep(t), and contains information about bias or disturbance effects affecting the system. The offset rejection capabilities of the k - incremental control law can therefore be attributed to the fact that the control law is capable of providing an implicit estimate of the bias term (or DC levels) (Tham et al. (213)). Shown in the form of Equation 8.21, the k - incremental control law is equivalent to the positional control law in Equation 8.15 with the terms in {.} equivalent to the $\mathbf{d}$ used to explicitly estimate the DC levels in the positional form. The advantage of the k - incremental algorithm over the positional form is that data for parameter estimation is naturally conditioned to have zero mean due to the differencing operation.

It is necessary to compute the coefficients of $\mathbf{P}(z^{-1})$ and $\mathbf{R}(z^{-1})$. If a first order reference model, $\mathbf{P}(z^{-1})$, and a first order setpoint filter, $\mathbf{R}(z^{-1})$, then

$$\mathbf{P}(z^{-1}) = (1 - \mathbf{p}\, z^{-1})/(1 - \mathbf{p}) \qquad 8.22a$$

$$\mathbf{R}(z^{-1}) = 1\text{-}\mathbf{r} \,/\, (1\text{-}\mathbf{r}z^{-1}) \qquad 8.22b$$

The $\mathbf{p}$ and $\mathbf{r}$ are determined by the time constant of the reference model and the setpoint filter, respectively. If the time constant of the reference model is $\tau_{\mathbf{P}(z)}$ and the time constant of the setpoint filter is $\tau_{\mathbf{R}(z)}$, then

$$\mathbf{p} = e^{-(\Delta T/\tau_{\mathbf{P}(z)})} \qquad 8.23$$

$$\mathbf{r} = e^{-(\Delta T/\tau_{\mathbf{R}(z)})} \qquad 8.24$$

where ΔT is the control interval.

8.2.2 The MIMO case

For each MIMO model, the corresponding self tuning controllers were obtained by treating each MISO model, independently. The design procedure for SISO case was then followed accordingly. For the MIMO case, the equation corresponding to Equation 8.13 is

$$\mathbf{P}(z^{-1}) = \mathbf{E}(z^{-1})\, \mathbf{A}(z^{-1}) + z^{-k}\, \mathbf{F}(z^{-1}) \qquad 8.25$$

The the corresponding $\mathbf{G}(z^{-1})$, $\mathbf{F}(z^{-1})$ and $\mathbf{E}(z^{-1})$ are given by

$$\mathbf{G}(z^{-1}) = \mathbf{E}(z^{-1})\, \mathbf{B}(z^{-1}) = \mathbf{G}_0 + \mathbf{G}_1 z^{-1} + ... + \mathbf{G}_{n+k-1}\, z^{-(n+k-1)}$$

$$\mathbf{E}(z^{-1}) = I + \mathbf{E}_1\, z^{-1} + ...\ \mathbf{E}_{k-1}\, z^{k-1}$$

$$\mathbf{F}(z^{-1}) = \mathbf{F}_0 + \mathbf{F}_1\, z^{-1} + ...\ \mathbf{F}_{n-1}\, z^{n-1}$$

$$\mathbf{d} = \mathbf{E}(1)\mathbf{d}.$$

The $\mathbf{G}_i$ and $\mathbf{E}_i$ have the form of $\mathbf{B}_i$ in Equation 8.3. The $\mathbf{F}_i$ is a diagonal polynomial matrix since the $\mathbf{P}$-canonical form was used. The corresponding weighting matices $\mathbf{P}(z^{-1})$, $\mathbf{Q}(z^{-1})$ and $\mathbf{R}(z^{-1})$ are diagonal polynomial matrices

$$\mathbf{P}(z^{-1}) = \text{diag}(\mathbf{P}_1(z^{-1}), \mathbf{P}_2(z^{-1})) \qquad 8.26$$

From here on the z^{-1} will be dropped from these polynomial matrices for clarity.

8.2.3 The parameter vectors, the data vectors and the control laws

From here on the controller designs corresponding to the models will be referred to as:

(i) SV-STC; a single variable self tuning controller based on the SISO model;

(ii) MD1 - STC; a multiple loop controller based on model MD1

(iii) MD2 - STC; a multiple loop controller based on model MD2

(iv) MD3 - STC; a multiple loop controller based on model MD3

Both MD1-STC and MD2-STC have interaction compensation, i.e. decoupling, since the effects of $\mathbf{u}_2$ on $\mathbf{y}_1$ and $\mathbf{u}_1$ on $\mathbf{y}_2$ have been incorporated into the model assumed in their designs. The MD3-STC does not include interaction compensation; it consists of two independent SV-STCs.

The parameter vectors, the data vectors and the control laws of these controller designs are given in the following. The positional forms using the "1 in the data vector" method for estimating d, is presented. In all cases $\mathbf{Q} = q = \lambda(1-z^{-1})$ for single loop case and $\mathbf{Q} = \text{diag}(q_1, q_2) = \text{diag}[\lambda_2(1-z^{-1}), \lambda_2(1-z^{-1})]$

1) **Single variable, STC , SV-STC**

The predictor model is

$$Py(t) = f_0 y_1(t-1) + g_0 u(t-1) + d \qquad 8.27$$

$$\theta = [f_0, g_0, d]^T$$

$$\emptyset = [y(t-1), u(t-1), 1]^T$$

$$u(t) = \frac{1}{(g_0 + q)}(-f_0 y(t) + Rw(t) - d) \qquad 8.28$$

2) **Multiple loop STC, MD1-STC**

The predictor model is

$$P_1 y_1(t) = f_0^{11} y_1(t-1) + g_0^{11} u_1(t-1) + g_0^{12} u_2(t-1) + d_1$$

$$P_2 y_2(t) = f_0^{22} y_2(t-1) + g_0^{21} u_1(t-1) + g_0^{22} u_2(t-1) + d_2 \qquad 8.29$$

$$\theta_1 = [f_0^{11}, g_0^{11}, g_0^{12}, d_1]^T$$

$$\theta_2 = [f_0^{22}, g_0^{21}, g_0^{22}, d_2]^T$$

$$\emptyset_1 = [y_1(t-1), u_1(t-1), u_2(t-1), 1]^T$$

$$\emptyset_2 = [y_2(t-1), u_1(t-1), u_2(t-1), 1]^T$$

$$f_0^{11} y_1(t) + g_0^{11} u_1(t) + g_0^{12} u_2(t) + d_1 - R_1 w_1(t) + q_1 u_1(t) = 0$$

$$f_0^{22} y_2(t) + g_0^{21} u_1(t) + g_0^{22} u_2(t) + d_2 - R_2 w_2(t) + q_2 u_2(t) = 0 \qquad 8.30$$

At every interval, the two equations in Equation 8.30 are solved simultaneously for $\mathbf{u}_1(t)$ and $\mathbf{u}_2(t)$. This can be done by formulating them in the form $A\mathbf{u}(t) = b$ and solving for **u** directly. The other approach is to formulate the equations in the form $f(\mathbf{u}(t)) = \mathbf{0}$ and solve for **u** by the Newton Raphson iterative procedure. This method was used in order to introduce generality. The tolerance limit for the iterative calculations were specified as 1×10^{-6}. Typically, convergence was achieved in less than 6 iterations for both simulation and real-time applications since the equations are linear at each interval.

3) Multiple loop STC, MD2-STC

The predictor model is

$$P_1 y_1(t) = f_0^{11} y_1(t-1) + g_0^{11} u_1(t-1) + g_1^{12} u_2(t-2) + d_1$$
$$P_2 y_2(t) = f_0^{22} y_2(t-1) + g_1^{21} u_1(t-2) + g_0^{22} u_2(t-1) + d_2$$

8.31

$$\theta_1 = [f_0^{11}, g_0^{11}, g_1^{12}, d_1]^T$$
$$\theta_2 = [f_0^{22}, g_1^{21}, g_0^{22}, d_2]^T$$

$$\varnothing_1 = [y_1(t-1), u_1(t-1), u_2(t-2), 1]^T$$

$$\varnothing_2 = [y_2(t-1), u_1(t-2), u_2(t-1), 1]^T$$

$$u_1(t) = \frac{1}{(g_0^{22} + q_1)} (-f_0^{11} y_1(t) - g_1^{12} u_2(t-1) - d_1 + R_1 w_1(t))$$

$$u_2(t) = \frac{1}{(g_0^{22} + q_2)} (-f_0^{22} y_2(t) - g_1^{21} u_1(t-1) - d_2 + R_2 w_2(t))$$

8.32

4) Multiple loop STC, MD3-STC

The predictor model is

$$P_1 y_1(t) = f_0^{11} y_1(t-1) + g_0^{11} u_1(t-1) + d_1$$
$$P_2 y_2(t) = f_0^{22} y_2(t-1) + g_0^{22} u_2(t-1) + d_2$$

8.33

$$\theta_1 = [f_0^{11}, g_0^{11}, \mathbf{d}_1]^T$$

$$\theta_2 = [f_0^{22}, g_0^{22}, \mathbf{d}_2]^T$$

$$\emptyset_1 = [y_1(t-1), \mathbf{u}_1(t-1), 1]^T$$

$$\emptyset_2 = [y_2(t-1), \mathbf{u}_2(t-1), 1]^T$$

$$\mathbf{u}_1(t) = \frac{1}{(g_0^{22} + q_1)}(-f_0^{11} y_1(t) - \mathbf{d}_1 + \mathbf{R}_1 w_1(t))$$

$$\mathbf{u}_2(t) = \frac{1}{(g_0^{22} + q_2)}(-f_0^{22} y_2(t) - \mathbf{d}_2 + \mathbf{R}_2 w_2(t))$$

8.34

8.2.4 Measurable load disturbances

In the case where the load disturbances are measurable the SISO system can be represented as

$$\mathbf{A}(z^{-1})\ y(t) = z^{-k}\ \mathbf{B}(z^{-1})\ u(t) + \mathbf{C}(z^{-1})\ \xi(t) + d(t) + z^{-kf} vL(t) \qquad 8.35$$

where L is the measured disturbance, and v is the disturbance parameter. The introduction of feedforward compensation is done in the same manner as the interactions between the loops were introduced in the MD1-STC and MD2-STC designs. Assuming kf = 2 then the predictor model becomes

$$\mathbf{P}y^*(t+k) = f_0 y'(t) + g_0 u(t) + d + z^{(k-kf)} vL(t)$$

$$= f_0 y'(t) + g_0 u(t) + d + z^{-1} vL(t) \qquad 8.36$$

The parameter vector is now $\theta = [f_0, g_0, d, v)$ and the control law becomes

$$u(t) = (-f_0 y'(t) + \mathbf{R}w(t) - d - v\ L(t-1))/(g_0 + q) \qquad 8.37$$

The "one in the data vector" method of estimating the offset level d has been assumed in the controller designs presented above. The "proxy of residuals" method was also allowed in the control algorithms. This method is given by Equation 2.76 as:

$$mn(t) = mn\ (t-1) + \varpi ep(t) \qquad 8.38$$

where ϖ is a scaling factor chosen depending on the speed of adaptation required. The mn(t) replaces the 1 in the data vector. The value of $\varpi = 0.8$ was used in this work. Note that the "proxy of residuals" approach is also a method of compensation for unmeasured or unmeasurable load disturbances (Morris et al. (86)).

8.2.5 Parameter estimation

The recursive least squares (RLS) scheme was employed to estimate the control parameters The scheme is given in Section 2.9.4 as follows:

Step 1 $\theta(t) = \theta(t-1) + K(t)\ [y(t) - \emptyset^T(t-k)\ \theta(t-1)]$

$$= \theta\ (t-1) + K(t)\ ep(t) \qquad 8.39$$

Step 2 $K(t) = PP(t-1)\varnothing(t-k)/[1 + \varnothing(t-k)^T PP(t-1)\varnothing(t-k)]$ 8.40

Step 3 $PP(t) = [I - K(t)\varnothing(t-k)] PP(t-1)/\upsilon(t)$ 8.41

where ep(t) is the prediction error, θ, is the parameter vector, Ø(t - k) is the data vector, K(t) is the estimator gain, PP(t) is the covariance matrix,υ(t) is the forgetting factor at time t and superscript T denotes transposition. The computation of PP(t) using Equation 8.41 led to instability in the estimator due to computer roundoff errors. Instability occurred after less than 150 repeats of Steps 1 to 3 on the System96. There is no provision for extra precision. The square root filter (SQRTF) algorithm was therefore used to update PP(t) since it is better conditioned numerically. The SQRTF algorithm, as presented in Kiovo (70), is given in Appendix A5.1.

8.2.6 Variable Forgetting Factors

In order to ensure that the parameter estimation scheme retains sensitivity, a forgetting factor $\upsilon < 1$ is inroduced into the estimation scheme as shown in Equation 8.40. As discussed in Section 2.9.6, a constant forgetting factor can cause estimator windup when the closed loop system is at near steady state conditions for long periods such that input changes in to the system are small. Estimator windup occurs because the covariance matrix, PP(t), would grow exponentially and become very large values as it is continually divided by υ which is less than 1. The most promising approach to reducing the risk of estimator wind-up is to introduce variable forgetting factor.

Three variable forgetting factor updating algorithms were incorporated into the STC controller designs considered in this work. In general, each variable forgetting factor algorithm reduces υ(t) when the prediction error increases. This has the effect of increasing the size of the covariance matrix and hence increases the speed of the adaptation. The forgetting factor converges to unity as prediction error becomes smaller. The algorithms are given in the following.

(A) Algorithm of Fortescue et al. (34)

The variable forgetting algorithm proposed by Fortescue et al (34) is a recursive algorithm that is based on the approach of defining a measure for the information content in the estimator. At each estimator cycle (Steps 1 to 3 above), the forgetting factor $\upsilon(t)$ is then chosen to keep this measure constant. The recursive forgetting algorithm is given as

$$\Sigma(t) = \upsilon(t)\Sigma(t-1) + [1 - \emptyset(t-k-1)^T K(t)]ep(t)^2 \quad 8.42$$

Here, the $\Sigma(t)$ is a measure of information content in the estimator. The strategy for choosing a forgetting factor, $\upsilon(t)$, is defined by keeping $\Sigma(t)$ constant such that

$$\Sigma(t) = \Sigma(t-1) = \ldots = \Sigma o \quad 8.43$$

The equation is given as

$$\upsilon(t) = 1 - \frac{1}{N(t)} \quad 8.44$$

$$N(t) = \Sigma o / [1 - \emptyset(t-k-1)^T K(t)] \, ep(t)^2 \quad 8.45$$

where $N(t)$ is the asymptotic memory length at time t. A memory length N implies that the information content in the estimator dies away with a time constant of N sampling intervals. Fortescue et al. gave a guideline on how to choose Σo. This is to choose Σo as

$$\Sigma o = \sigma_0^2 \, No \quad 8.46$$

where σ_0^2 is the expected measurement noise variance and No is the nominal asymptotic memory length. The algorithm will be referred to as VFF1 in this thesis.

(B) Algorithm of Ydstie et al. (146)

Ydstie et al. (146) presented another version of the Fortescue et al. updating formula given above. The formula is claimed to be better and is given as:

$$\upsilon(t) = No \Big/ \left[No + ep(t)^2 \{1 + \emptyset(t-k)^T PP(t) \emptyset(t-k)\}^{-1}\right] \quad 8.47$$

where No is still the nominal aymptotic memory length and $\upsilon(t)$ is chosen to keep No constant. According to Seborg et al. (140), typical values of No are from 10 to 10^4. This algorithm is sensitive to the choice of No. This algorithm will be referred to as VFF2 in this thesis.

The No (or Σo) controls the speed of adaptation. A small No (or Σo) will give a large covarianve matrix which will result in rapid adaptation and a sensitive estimator. If it is chosen too small, it may lead to blowing up of the covariance matrix and corresponding unstable control. A large No (or Σo) will result in a less sensitive estimator and slower adaptation. Fortescue et al. (34) have used the algorithm successfully and they showed that the algorithm was sensitive to the choice of Σo.

(C) Modified form of the Algorithm of Ydstie et al. (146).

Another version of the above algorithm is also available and is presented in Ydstie (151). It is given as:

$$\upsilon(t) = \mathrm{Tr}\,(PP(t))/\left[\mathrm{Tr}\,(PP(t)) + ep(t)^2\{r + \varnothing(t-k)^T PP(t)\varnothing(t-k)\}^{-1}\right] \qquad 8.48$$

where Tr (PP(t)) is the trace of PP(t) and r is a value that is chosen by the user. In this work, r = 1 was used. The formula is the same as Equation 8.47 except that Tr(PP(t)) replaces the No in Equation 8.47. As is claimed in Ydstie (151), the Tr(PP(t)) will be prevented from becoming too large since $\upsilon(t)$ will tend to 1 as Tr(PP(t)) tends to infinity; and the Tr(PP(t)) will be prevented from vanishing to zero since $\upsilon(t)$ will tend to zero as Tr(PP(t)) tends to zero. This algorithm will be referred to as VFF3 in this thesis.

8.3 Introducing parameter correction into the self tuning algorithm

In the absence of persistent excitation the parameter estimates may converge to values far from their true values and this may cause problems of stability and robustness. Assuming that there is no control weighting in the SV-STC algorithm ($Q = 0$), then problems will arise if, for example, the g_0 becomes close to zero since large control actions will be generated since the g_0^{-1} is equivalent to the controller gain. Another possibility is for g_0 to attain the wrong sign thus computing wrong control actions. This may lead to poor controller performance or even instability.

Similar problems may still arise even when $\mathbf{Q} \neq 0$. For example, if g_0 attains values of opposite sign to that of $\mathbf{Q}$ and magnitude as large as or greater than $\mathbf{Q}$.

As reported in Chapter 2 Section 2.9.8, in response to problems due to lack of excitation in adaptive control, workers such as Lozano-Leal and Goodwin (147), Kreisselmeir (184), and Ossman and Kamen (94) have suggested parameter estimation algorithms that do not require persistent excitation to give good parameter estimates. The method proposed by Ossman and Kamen (94) was chosen for investigation in this research work because of the simplicity and the flexibility of the method. The approach has been introduced in Chapter 2 Section 2.9.8. To recap, the method assumes that the system parameters, θ, belong to a known bounded interval $[\theta^{min}, \theta^{max}]$. The recursive least square estimation scheme was then modified in a way such that θ is forced into the bounded interval as the control progresses. The modified recursive least square algorithm also includes a data normalisation procedure, which was introduced to ensure that the magnitude of the covariance matrix, PP(t), converges to a value which is smaller than that of the initial covariance matrix, PP(0).

Ossman and Kamen applied the method for the case where the bounds of all the system parameters were known. They applied it in combination with an explicit MIMO self tuning regulator, so that the system parameters were first computed and then the feedback control law was calculated based on the system parameters. They conjectured that the method may be applicable in cases where the bounds of only a subset of the parameters in the vector, θ, are known. This possibility was considered in this work because it is usually the case in many chemical engineering applications of adaptive control that the number of parameters to be identified is large and a good knowledge of all the parameters is unlikely to be available. Also stronger conditions of persistent excitation must be satisfied as the number of parameters increase. Ossman and Kamen pointed out, however, that the stability of the controlled system cannot be guaranteed for all possible values of the system parameters when the bounds of only a subset of the parameters are known. In Chapter 3, their conjecture was restated to imply that it may be possible to still have a workable robust control algorithm even if only a few of the parameters are corrected. This is considered advantageous in applications where there are parameters for which the estimator cannot provide good estimates. Using this method, these can be moved into suitable regions, where it is known that good control will result. In this work, the method was considered to be useful in preventing key parameters such as the g_0, g_0^{11}, and g_0^{22} from attaining unrealistic values. This is because these parameters are the main factors determining the controller gains of their respective control loops.

In contrast to the work of Ossman and Kamen, an implicit form of an adaptive controller design method was used in this work.

8.3.1 A simplified form of parameter correction

The basic features of the method of Ossman and Kamen are as follows:

$$\mathbf{f}_i(t-1) = \begin{cases} \theta_i(t-1) - \theta_i^{max} & \text{when } \theta_i(t-1) > \theta_i^{max} \\ \theta_i(t-1) - \theta_i^{min} & \text{when } \theta_i(t-1) < \theta_i^{min} \\ 0 & \text{when } \theta_i^{min} \leq \theta_i(t-1) \leq \theta_i^{max} \end{cases} \qquad 8.50$$

where $\mathbf{f}$ is an N x 1 vector and N is the number of parameters and subscript i denotes the location on the vector. Equation 8.28 in the RLS scheme is then modified as follows:

$$\theta(t) = \theta(t-1) + K(t)\{\ y(t) - \varnothing^T(t-k)\theta(t-1)\} - \alpha PP(t-1)\mathbf{f}(\theta(t-1)) \qquad 8.51$$

where α is a positive scalar chosen such that

$$\alpha PP(0) < 2I \qquad 8.52$$

where PP(0) is a diagonal matrix with equal elements. Together with this correction term, a data normalisation procedure was used in the standard least square algorithm for reasons explained in the previous section. It is clear that the "correction term" $-\alpha PP(t-1)\mathbf{f}(\theta(t-1))$ can be included in any suitable estimation algorithm. It was introduced into the algorithms considered in this work. As mentioned earlier, the standard method of updating PP(t) in the RLS scheme was not used in this work due to estimator stability problems. The square root filtering method was used to update PP(t) and the option given of the three variable forgetting algorithms as mentioned earlier in the chapter. The parameter correction approach detailed above was combined with this RLS method. The data normalisation procedure used by Ossman and Kamen was not used here, instead a simpler approach was employed with the RLS. This is

$$\eta = \max(1, \| \varnothing(t-k) \|_{\infty}) \qquad 8.53$$

so that $\varnothing(t-k) = \varnothing(t-k)/\eta$. Where $\|x\|$ is the maximum element in the vector x. In this thesis, the resulting correction algorithm is referred to as the *parameter correction* (PC) method.

The interest in this work was to examine whether the algorithm can be applied to the case where the bounds of only a subset of the control system parameters are known. In this case, it is necessary to make an assumption about the $\mathbf{f}_i(\theta(t-1))$ entries corresponding to those parameters whose bounds are not specified. In this work, the assumption was made that they are zero. The assumption implicitly implies that the corresponding parameters are always within their bounds, or that their estimates are always good. The assumption does not mean, however, that the corresponding elements in the vector $\alpha PP(t-1)\mathbf{f}(\theta(t-1))$ will have zeroes since there will be nonzero entries in the off-diagonal elements in PP(t-1). The corresponding parameters in the vector θ will therefore be corrected even though their corresponding $\mathbf{f}(\theta(t-1))$ entries will be zero.

A simpler form of the PC method was also considered in this work. It is called the simplified parameter correction (SPC) method. In the SPC method, the nature and the rate of correction were "decoupled" from the estimator, rather than being tied to the behaviour and the magnitude of the covariance matrix, PP(t). One reason why this was done was that in the PC method, when the magnitude of PP(t) becomes very small relative to the magnitude of PP(0), the correction made to the parameters will become negligible. This was experienced in the simulations performed in this work as the PP(t) usually converged very fast to small values meaning sensitivity of the estimator was quickly lost. One *ad hoc* method that can be used to avoid this is to hold the PP(t) constant if Tr(PP(t)) falls bellow a user specified limit which is large enough to ensure adequate correction is always maintained. Another method is to add a non zero diagonal matrix to the PP(t) if the elements become too small. This is similar to reinitialising the covariance matrix. These two methods have their short comings. In the first approach it may be dificult to find the appropriate lower limit of Tr(PP(t)) which will maintain parameter correction but at the same time ensuring the the limit is not too large to make the estimator unduly sensitive at all times. In the second approach, experience in this work demonstrated that bursting of the estimator will occur when the covariance matrix is reinitialised and this bursting can result in bad estimator performance and hence poor control.

The simplification of the PC method was done by replacing αPP(t) in Equation 8.51 with a scalar μ. The correction term is then

$$- \mu \mathbf{f}(\theta(t-1)) \qquad 8.54$$

The parameter updating equation becomes

$$\theta(t) = \theta(t-1) + K(t)[y(t) - \varnothing^T(t-k)\theta(t-1)] - \mu \mathbf{f}(\theta(t-1)) \qquad 8.55$$

By this replacement a constant correction rate is maintained as it is implicitly assumed that PP(t) = I for $t = 0$ to $t = \infty$, that is at all times during operation of the adaptive control algorithm. In this case then $\mu < 2$. The correction of the parameters is "decoupled" from the estimator as the correction term $- \mu \mathbf{f}(\theta(t-1))$ is now independent of PP(0) and PP(t). Also each parameter $\theta_i(t)$ can be moved independently as the trajectories of only the parameters that have their bounds specified will be modified.

Note that Equation 8.55 is similar to the leakage approach shown in Equation 2.93 in Section 2.9.6. It can be viewed as an estimator the parameters of that system are "externally" forced into suitable regions specified by the upper and lower bounds.

The next problem was on the proper choice of μ. Consider when estimator sensitivity is lost completely so that the parameters estimates have converged. Then choosing μ such that $0 < \mu < 1$, a parameter θ_i that violates the specified bound will approach the "target" bound asymptotically and will never cross the bound (Figure 8.2a). That is,

$$\theta_i(t) \text{ tends to } \theta_i^{max} \text{ as t tends to } \infty \text{ when } \theta_i(t) > \theta_i^{max}$$

$$\theta_i(t) \text{ tends to } \theta_i^{min} \text{ as t tends to } \infty \text{ when } \theta_i(t) < \theta_i^{min} \qquad 8.55$$

If μ is chosen such that $1 < \mu < 2$, the target bound will be crossed in one step. There is then the danger that the "corrected" parameter to overshoot the other bound opposite the target bound (Figure 8.2b). To prevent this behaviour, μ should be selected as $0 < \mu < 1$. Note that the width of the bounds of the parameters can be made very small if the exact values of the parameters are known.

These PC and SPC approaches were compared and used as a remedy in some difficult cases encountered in this work.

8.4 Software development for implementing the controllers

Computer algorithms for the self-tuning controllers were written on the System96 in Basic09 and on the IBM PC AT in Quickbasic. Those written on the System96 were for simulation studies and for on-line applications on the distillation column. Those on the IBM PC AT were used for evaluations on the column simulator since IBM PC AT is a much faster machine so that running times were shorter than on the System96. This has been discussed in Chapter 5.

8.4.1 Software on the System96

The objective was to develop a computer package that would implement self-tuning control schemes for the general case, regardless of whether the system is a real process or a model. It was considered beneficial to exploit the multitasking features of the System96, so that the self-tuning control computer program and the data logging program, or the process model, would be running concurrently. This way the

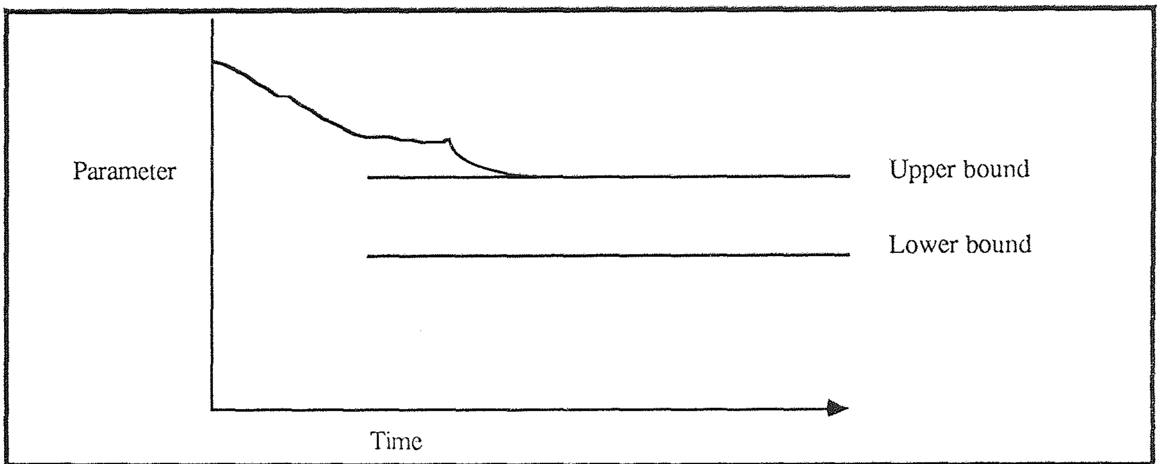


Figure 8.2a) Trajectory of a parameter using the SPC with $0 < \mu < 1$

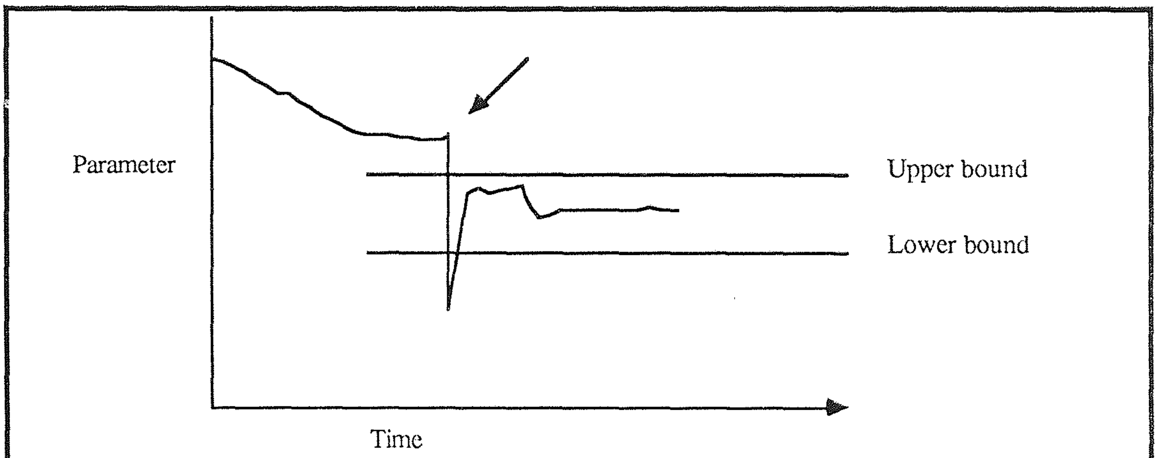


Figure 8.2b) Trajectory of a parameter using the SPC with $1 < \mu < 2$

intersample computational load and programs memory requirements could be distributed between the two programs.

The features intended for this general program were both single loop and multiple loop STC of general order and time delays, including multiple time delays. Multiple sampling rates and the extended RLS scheme for identifying the noise model were also included in the specifications. Other specifications include positional and k-incremental forms of the controller, moving average filtering, choice of constant or variable forgetting factors, and choice of " 1 in the data vector " and " the proxy of residuals " methods of offset estimation.

These tasks proved difficult to achieve, particularly the combination of multiple sampling, positional and k-incremental algorithms and the moving average filter within one program. The programming requirements of the necessary organisational, house-keeping and data handling facilities involved very complex logical operations. Program memory and computational requirements was also a limiting factor; it easily violated the 64K limit imposed by the System96 hardware.

For the concurrent execution of the process model (or data logging program) and the control algorithm, two-way communication between both programs (the source and the sink) was required. The source is the model in the case of simulation or the data logging program in the case of on -line application; the sink is the self tuning control software. Achieving this two way communication was possible when it was tried with simulations, but very untidy programming was required and marked reduction in program execution speed resulted. Furthermore, there was a frustrating tendency of both programs to "hang" while trying to access and send data to and from each other. The reason for this was that with the "pipe" facility on System96 can send data from only from source to the sink, as mentioned in Chapter 4. Sending data, from the sink back to the source, for example sending the control inputs to the simulator, has to be done through a file. This included having to suspend program execution in both source and the sink at many different locations in order to ensure that when data is sent from the source, the sink is ready to receive it in such a way that the sink does not have to wait too long (< 15 seconds) otherwise programming error resulted. In the case of the column simulator as the source, the fact that the self tuning control program completes its tasks within seconds while the solumn simulator required several minutes for every sample interval, made matters worse.

The positional and incremental self-tuning algorithms were therefore written separately, each including the data logging program (or process model) within it, and the option of extended least squares if the noise model is to be identified. The incremental version incorporates the option of a moving average filter. Up to four previous input and output sets of data are kept in the estimator memory at any one time to enable the moving average filtering of the data and allow larger time delays to be used. The positional and incremental algorithms differ very little in their computational logic.

In general each program module follows the sequence given below;

(A) *Initialisation* of the matrices, vectors and variables.

(B) *Menu*: The program is partly menu driven. The menu carries the user through to choose some model, controller and estimator parameters. Other inputs are read from a file on the floppy disk.

(C) *Parameter estimation and Control calculations:* This includes (1) collecting measurements from the experimental column, or from a process simulator (2) estimating new controller parameters, (3) updating the controller parameters and (4) calculating control action.

(D) *Data storage onto a floppy disk.* A large amount of the results needed to be stored at every sample interval for analysis purposes. In practice much less data will need to be stored at every sampling instant.

(E) *Error trapping*: The program continuously checks for programming and computational errors.

8.4.2 Computational and storage requirements of the software

The computer programs discussed above which implement self tuning control on the System96, have large memory requirements. The memory required to store data in the parameter estimator depends directly on the order of the assumed model of the process. As the orders n and m increase the length of the data and parameter vectors and the size of the covariance matrix will increase significantly. The computational effort required for each control cycle would also increase dramatically. If a first order system, with a single time delay, one load disturbance for feed forward compensation and using a common sampling interval for all the control loop is considered, the overall program memory requirements can be reduced by a factor of 4, at least. This is possible for the following reasons;

1) the length of the data and parameter vectors will be kept to a minimum

2) there will be no need for writing the program to handle cases such as multiple sampling rates and the identification of the noise model.

Preliminary tests showed that the time required for both estimation and control calculation in each self tuning cycle of MD1-STC is always within 5 seconds. For SV-STC case, this time is about 2 seconds, and for MD2-STC and MD3-STC the times are about 3 seconds. These times are reasonably small compared with 30 seconds sampling interval that was usually used to control the distillation column.

Preliminary tests of on-line application of the control algorithms showed that the retrieval of process data by the data logging program **Get-data**, the output of information on the VDU screen and storage of necessary data onto the floppy disk, took up a significant proportion of the sampling interval of 30 seconds. Table 8.1 shows the time required for each task in the adaptive control system for on-line control. Each cycle constituting, data logging, parameter estimation and control calculation, output of information on the VDU screen and data storage on the floppy disk was usually completed within 20 seconds.

Table 8.1 The computational times of the self tuning algorithms on the System96

Task	Average Time in seconds		
	SV-STC	MD2-STC**	MD1-STC
Data Logging	............7............		
Parameter estimation*	2	4	5
Data Storage	............5............		
* includes calculation of the control input			
** MD3-STC requires the same amount of time as MD2-STC			

8.4.3 Software on the IBM PC-AT

The self-tuning computer algorithms were translated into the Quickbasic programming language available on the IBM PC-AT microcomputer. In the translation, some of the generalities included in the algorithms were excluded and this reduced the program storage and memory requirements significantly.

All the four designs, SV-STC, MD1-STC, MD2-STC and MD3-STC, with the choice of positional and k - incremental (k=1) forms, moving average filtering (MAF), the parameter correction (PC) and the simplified parameter correction (SPC) methods were included in one single program package. The option of PI control (velocity algorithm) is also available. The program is entirely menu driven. The software details are presented in Appendix A5.2.1. The whole software package required about 1500 lines of program statements.

To simulate self-tuning control of the distillation column using the MD1-STC required about 50 minutes of computer time to simulate 100 minutes of process time; that is, the ratio of computer time to process time is 0.5 to 1 using an interval of 0.025 minutes to integrate the differential equations of the column simulation.

8.5 Chapter review

This chapter has described the design of four generalised minimum variance self-tuning controllers for single variable and multivariable control based on the column simulator. The next stage is evaluating these controllers on the column simulator and compare performance with PI control.

CHAPTER NINE

Evaluation of the Self Tuning controllers on the column simulator

9.1 Introduction

The self-tuning controller designs presented in the previous chapter were evaluated by computer simulations on the column simulator derived in Chapter 5 for the pilot scale distillation column used in this work. The simulations were carried out to demonstrate how the self tuning controllers work and to assess the benefits of using self tuning control over conventional PI control on the distillation column. The simulations were performed on the IBM PC AT microcomputer.

9.2 Simulation on a simple linear model

The servo and regulatory performances of the positional and 1-incremental STCs were compared on a simple linear discrete time SISO system:

$$y(t) = ay(t-1) + bu(t-1) + d(t) \quad 9.1$$

$y(0) = u(0) = d(0) = 0$, $a = 0.9$ and $b = 1.0$

The unmeasured load disturbances, d(t), in affecting the system is in form of a square wave as shown in Figure 9.1.

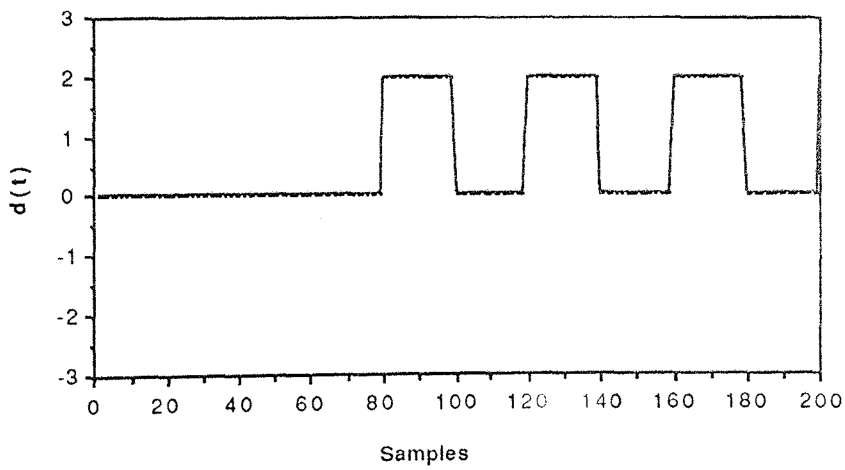


Figure 9.1 Load disturbance on the linear system

In these simulations the following settings $\mathbf{P} = \mathbf{R} = 1$ and $\mathbf{Q} = 0$. Three of such simulations to be discussed used the following:

1) Positional form, without estimation of a bias term (or DC level) (Figure 9.2)

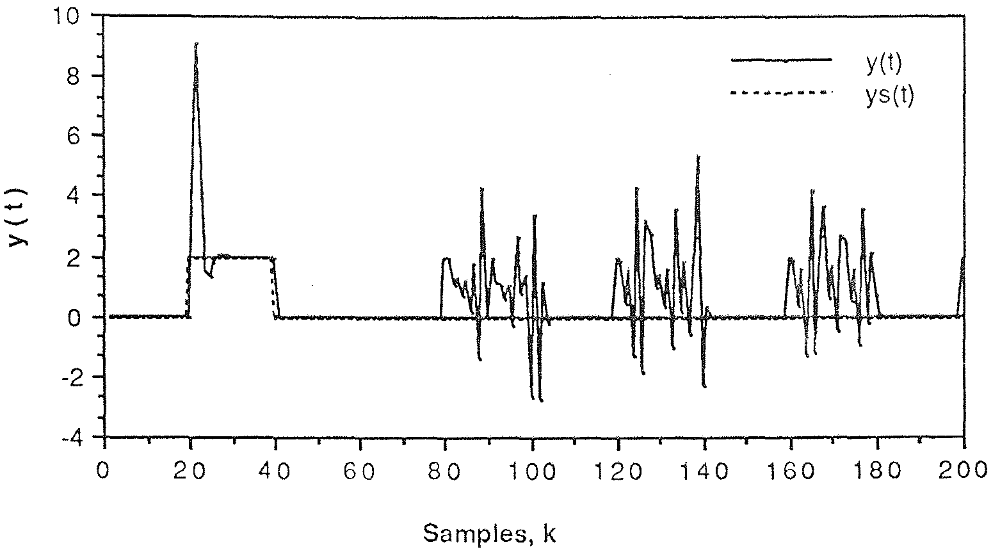
2) Positional form, with explicit estimation of bias using the "1 in the data vector" method (Figure 9.3)

3) Incremental form (Figure 9.4)

The positional and the incremental STCs give the same servo performance, but the incremental STC is superior to the positional STC when regulating against the unmeasured load disturbances in Figure 9.1. As Figure 9.3 shows, the positinal STC still gives a very poor regulatory performance even when the bias term is estimated.

The positional form relies on the accurate estimation of the bias to perform well in the presence of the unknown disturbances. The incremental form, on the other hand, does not required such an estimate because the control law provides an implicit estimate of the bias or disturbance effects affecting the system through the term {.} in Equation 8.21 in Section 8.2, which represents the estimation error.

To demonstrate the effect of non-linearities on these characteristics of the positional and incremental form, similar simulations were performed on the non-linear column simulator. The result for Cases 2) and 3) above are compared in Figure 9.5. The overall performances of both controllers were good, but the incremental form is better by 17% in terms of the IAE since it gives a faster closed loop response. Unlike the linear case, the servo performances of both controllers were not the same as the incremental form returns the output to the setpoint quicker than the positional form. The reason for this difference is due to non linearity since the $\mathbf{F}$ and $\mathbf{G}$ parameters are now time varying so that $\mathbf{F}_{(t-k)} \neq \mathbf{F}_{(t)}$.



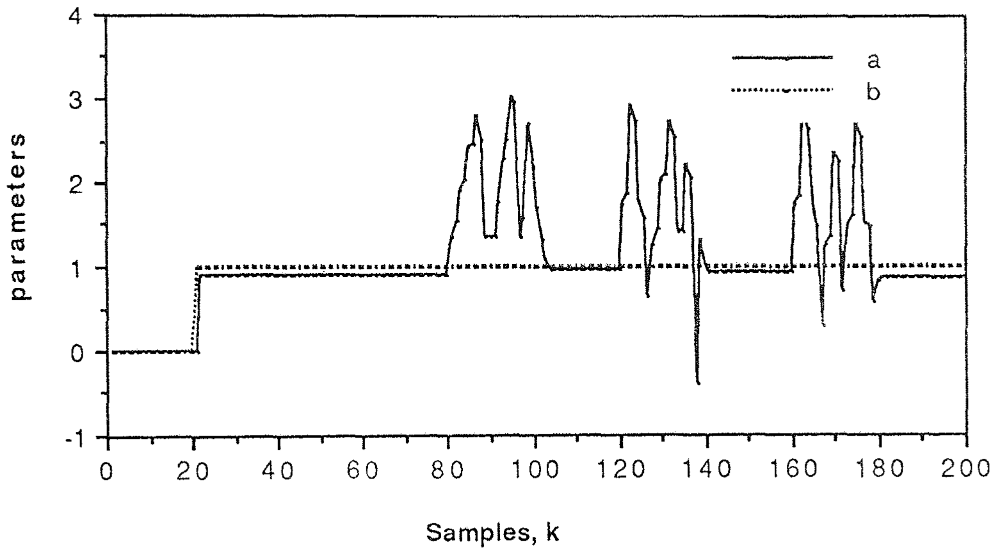
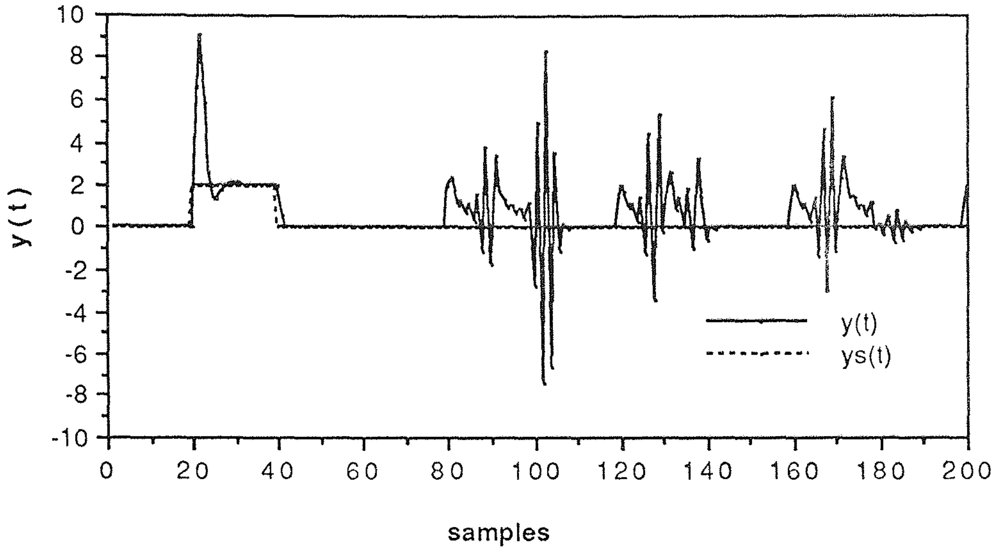


Figure 9.2 Servo and regulatory performance of a positional self tuning controller, without estimation of the bias term, d.



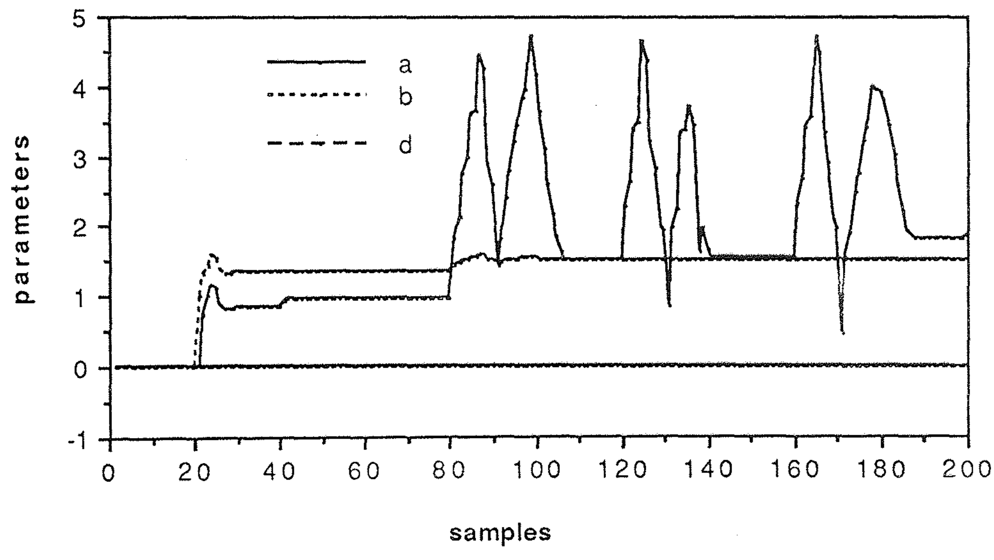
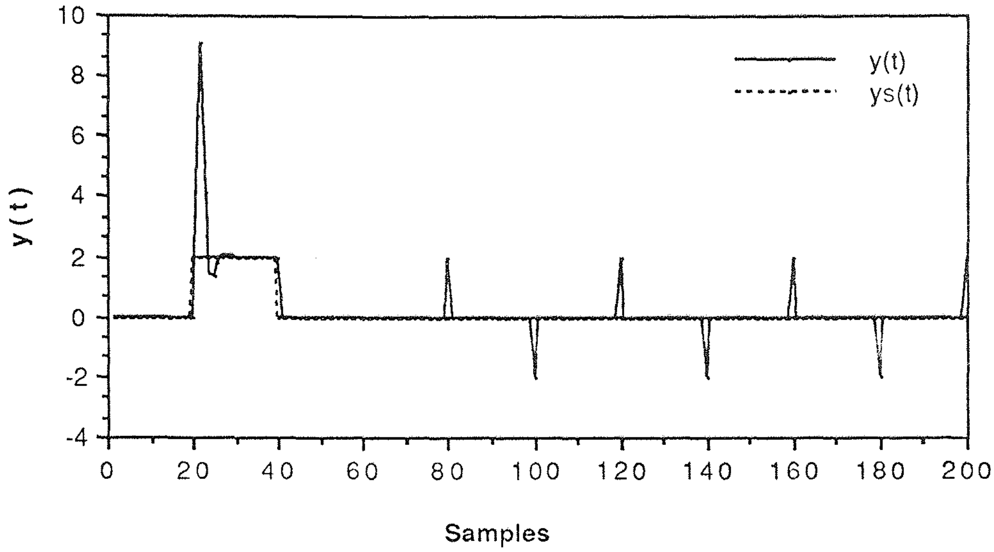


Figure 9.3 Servo and regulatory performance of a positional self tuning controller which includes estimation of the bias term, d.



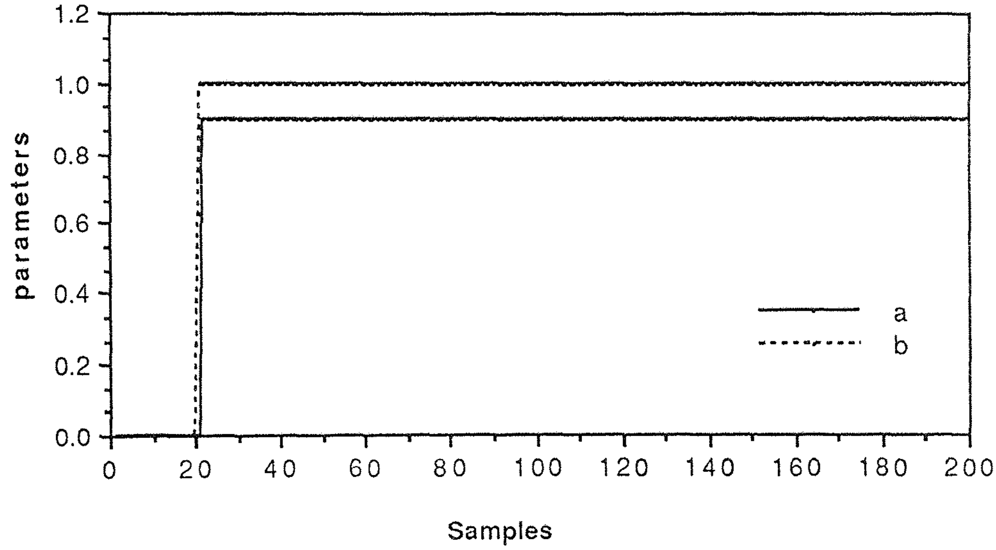
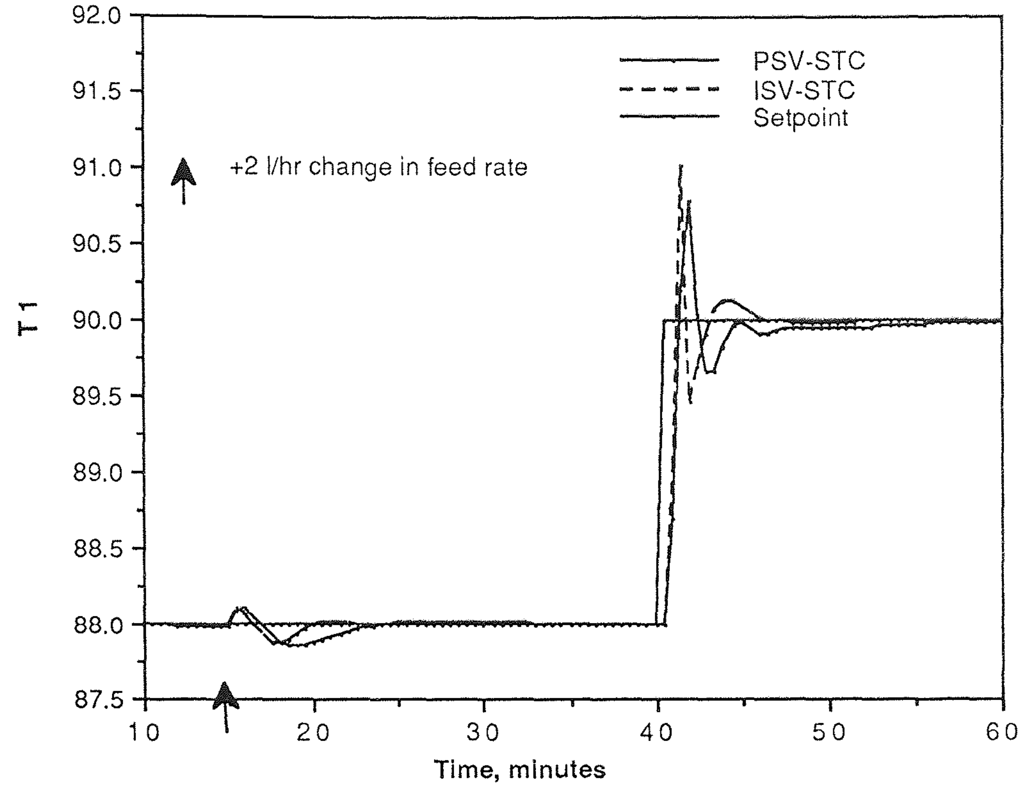


Figure 9.4 Servo and regulatory performance of an incremental self tuning controller



IAE = 5.20 - PSV-STC

IAE = 4.32 - ISV-STC

$\mathbf{Q} = -0.5\Delta_1$

Figure 9.5 Effect of non-linearity on the servo and regulatory performances of the positional and incremental self-tuning controllers

9.3 Single loop top tray temperature control

Single loop top tray temperature control of the column simulator were carried out with the simulator subjected to a series of load changes shown in Figure 9.6 below.

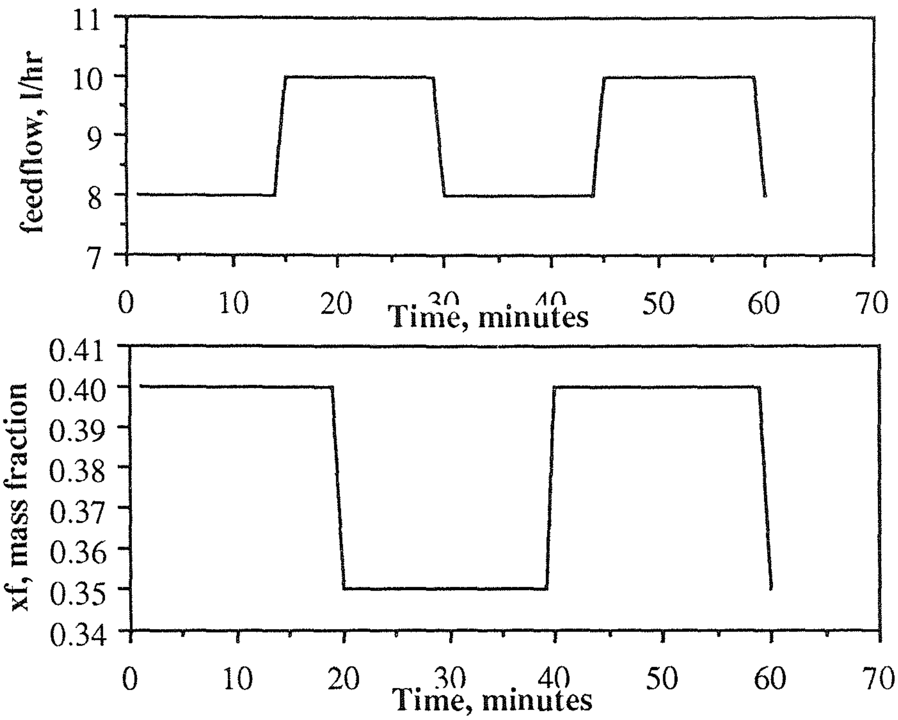


Figure 9.6 Load disturbances for single loop top tray temperature control

The following were specified for the; SV-STC, $\theta(0) = (0,-0.3,0)$, $PP(0) = 10I$, $\mathbf{Q} = \lambda\Delta_1 = -0.5\Delta_1$. The PI gain is within the range calculated in Table 5.6 using the Cohen and Coon controller tuning method for setpoint changes. The $\theta(t)$ were initially tuned in for 5 minutes using the velocity algorithm form of a PI controller with these setting. Controller performance will be measured using the Integrated Absolute Error (IAE) criterion.

Table 9.1 shows the IAEs for the various runs performed. Figure 9.7a shows the closed loop responses for two cases using PI with Kc = -2.0 l/hr/°C and Kc = -2.5 l/hr/°C. The closed loop responses in both cases were similar even with the 25% increase in Kc; this is evident in that IAE for the case with higher Kc is only marginally lower. As expected, the performance of the PI degraded by 25% when the control interval, ΔTc, of 1 minute was used (see plot in Figure 9.7c).

Table 9.1 Integrated Absolute Error (IAE) for single loop top tray temperature control

Controller	Controller Specifications	IAE
PI	Kc = -2.0 1/hr°C, τ_i = 3 min	24.59
PI	Kc = -2.5 1/hr°C, τ_i = 3 min	24.22
PI**	Kc = -1.15 1/hr°C, τ_i = 3 min, ΔTc = 1 min	30.91
PSV-STC	$Q(z^{-1}) = -0.5\Delta_1$, $\tau_{P(z)} = 0.5$ min, $\tau_{R(z)} = 0$	14.65
ISV-STC	$Q(z^{-1}) = -0.5\Delta_1$, $\tau_{P(z)} = 0.5$ min, $\tau_{R(z)} = 0$	12.73
ISV-STC**	$Q(z^{-1}) = -0.5\Delta_1$, $\tau_{P(z)} = 0.5$ min, $\tau_{R(z)} = 0$	18.43
ISV-STC	$Q(z^{-1}) = -0.8\Delta_1$, $\tau_{P(z)} = 0.5$ min, $\tau_{R(z)} = 0$	14.16
ISV-STC	$Q(z^{-1}) = -0.5\Delta_1$, $\tau_{P(z)} = 1.0$ min, $\tau_{R(z)} = 0.5$	15.41
ISV-STC	$Q(z^{-1}) = -0.6\Delta_1$, $\tau_{P(z)} = \tau_{R(z)} = 0.5$ min	14.28

** Control interval of ΔTc = 1 minute was used.

**** $\tau_{P(z)}$ denotes the time constant of the reference model $P(z^{-1})$, $\tau_{R(z)}$ denotes the time constant of the setpoint filter, $R(z^{-1})$.

Table 9.2 Integrated Absolute Error (IAE) for single loop top tray temperature control: Comparison of the performance of ISV-STC with the different variable forgetting factor algorithms.

Algorithm	Specifications	IAE
VFF1	Σo = 0.01	12.64
VFF1	Σo = 0.5	12.77
VFF2	No = 0.1	12.48
VFF2	No = 1.0	12.71
VFF3		12.73

Controller specification: $Q(z^{-1}) = -0.5\Delta_1$, $\tau_{P(z)} = 0.5$ min, $\tau_{R(z)} = 0$

Compared with PI , PSV-STC gave a much faster closed loop response and much tighter control, but at the expense of larger control actions and overshoot of the setpoint (Figure 9.8). The improvement in control compared with the PI is 40%. The ISV-STC improved control even further in that the IAE is 48% lower than for PI. There was, however, larger changes in the parameter estimates compared with PSV-STC (Figure 9.10) and hence the larger transient behaviour under ISV-STC, which is evident between t = 30 and t = 45 in Figure 9.9. As explained in Tham et al. (213), the larger changes in the parameter estimates of the ISV-STC is because when the responses are closer to the setpoints, data for estimation have small values due to the differencing operation so that parameter estimation is curtailed significantly. Subsequent transients, due to changes in setpoints or load disturbances, are magnified due to the differencing operation causing larger changes in the parameter estimates compared to the positional form. This, in general, is a characteristic of self-tuning algorithms that employ data differencing for parameter estimation.

The effect of increasing the control interval, specifying a slower reference model and specifying a larger control weighting on the performance of ISV-STC are shown in Figure 9.11 through to 9.13. In all these cases, the closed loop responses were slower, which is reflected in the fact that their IAEs on Table 9.1 are larger.

All the simulations reported above used the VFF3 variable forgetting algorithm of Equation 8.48. Figure 9.14 shows the plots of the $\upsilon(t)$ and Tr(PP(t)) for ISV-STC which correspond to the parameter estimates in Figure 9.10. The performances of the ISV-STC, in terms of the IAE, using the other two algorithms, VFF1 and VFF2 (Equations 8.45 and 8.47, respectively) are shown on Table 9.2. The effects of the parameters Σo and No on the performances of their corresponding algorithms are also shown. Comparing Figure 9.15 and 9.16 in which ISV-STC used VFF1, it is clear that the estimator is much more sensitive when $\Sigma o = 0.01$ than when $\Sigma o = 0.5$ as the plot of the forgetting factor and Tr(PP(t)) show. The result is faster adaptation of the controller parameters and hence a slight improvement in terms of the IAE as Table 9.2 shows. Similar results was also obtained with VFF2 as Figures 9.17 and 9.18 show.

Although Table 9.2 shows that using VFF3 the controller performance is not as good as with VFF1 and VFF2 with the smaller Σo and No, the VFF3 was preferred in this work as it does not require the choice of any parameter as is the case with the other two algorithms. Thus, compared with when VFF1 or VFF2 is used, there is one less parameter to select when implementing self-tuning control when the VFF3 algorithm is used for variable forgetting.

9.3.1 Summary

On the basis of the IAEs on Table 9.1, the ISV-STC is a much better choice than PSV-STC and PI control of the top tray temperature of the column simulator. If considerations such as sensitivity of the closed loop is more important, then PSV-STC is better than ISV-STC in that under the same set of conditions the parameter estimates for the latter would change more significantly when disturbances enter the system.

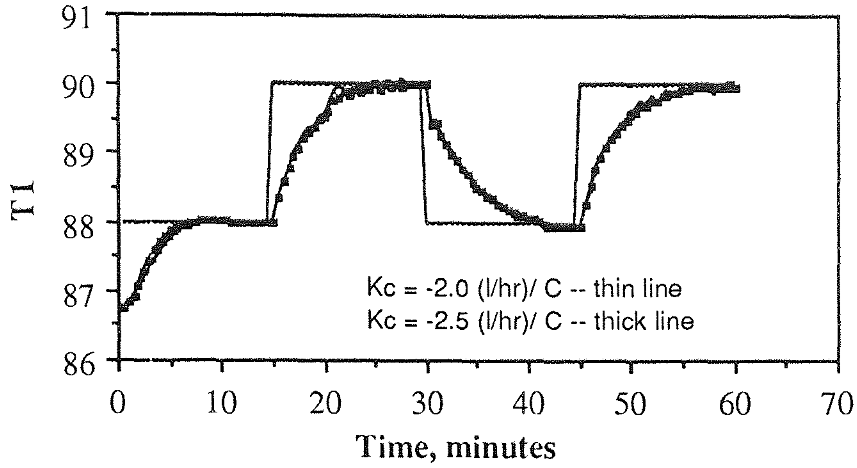


Figure 9.7a Top tray temperature control using PI: ΔTc = 0.5 minute Kc = -2.0 l/hr/°C, τ_i = 3.0 minutes vs. Kc = -2.5 l/hr/°C, τ_i = 3.0 minutes

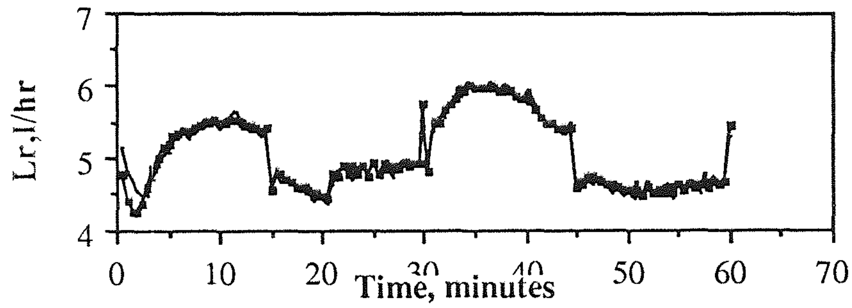


Figure 9.7b Control actions for Figure 9.7a

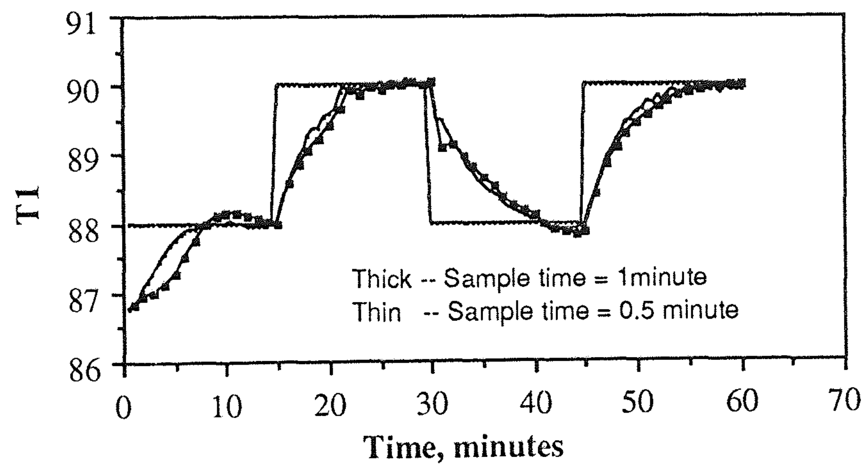
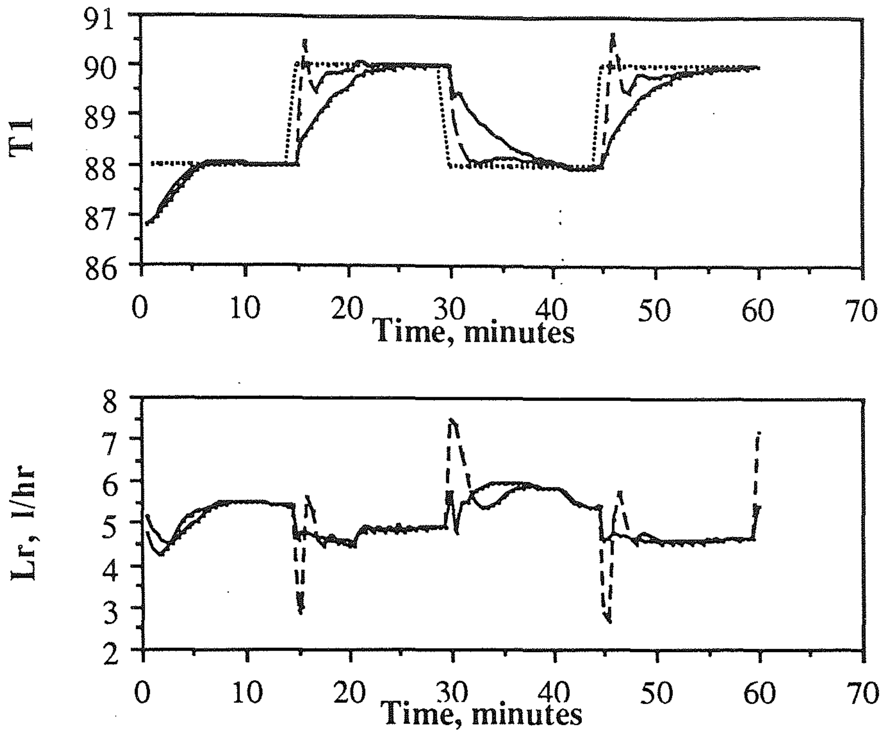
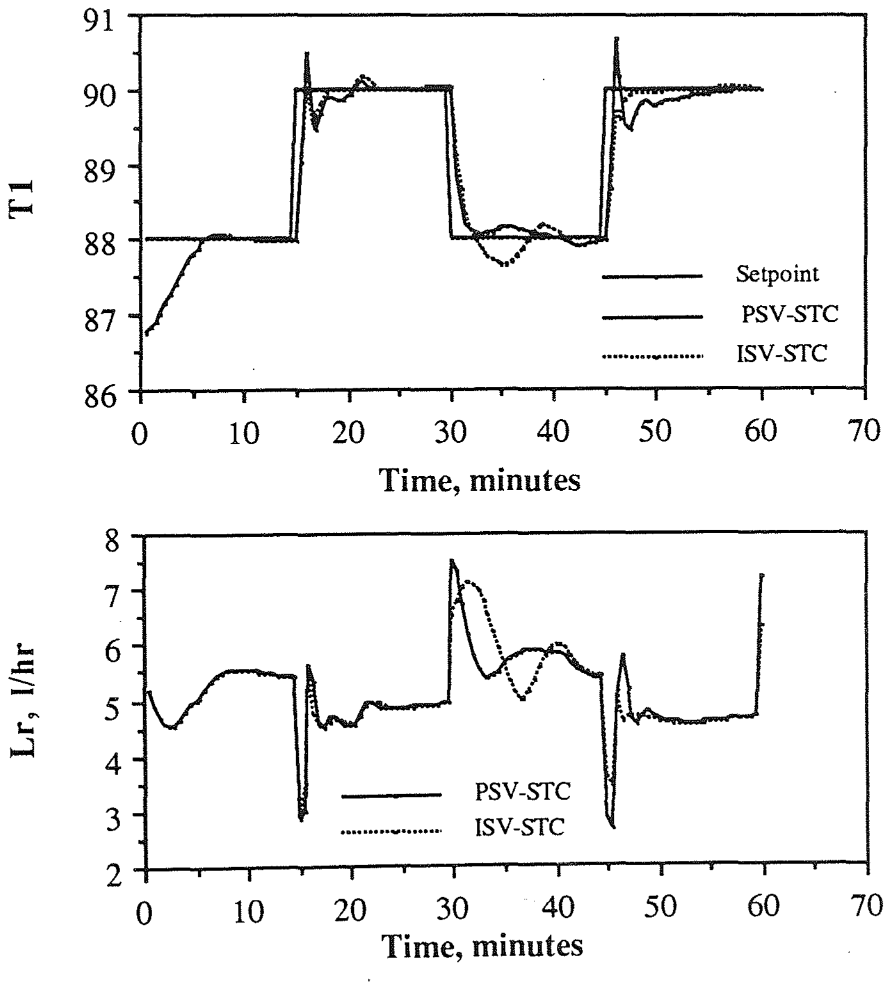


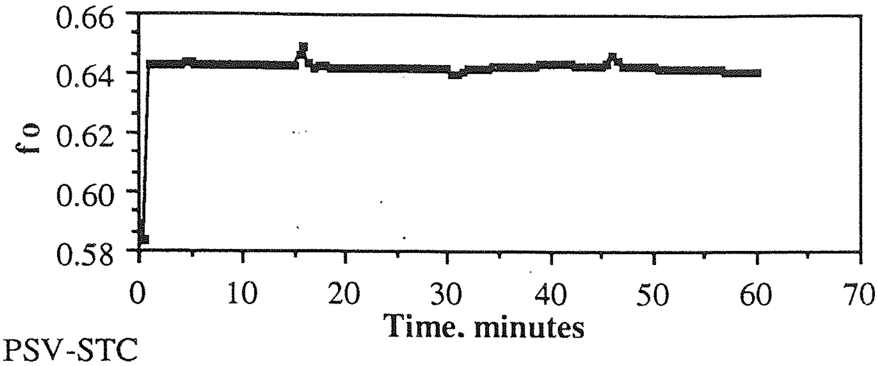
Figure 9.7c Top tray temperature control using PI: ΔTc = 1.0 minute, Kc = -1.15 l/hr/°C, τ_i = 3.0 minutes



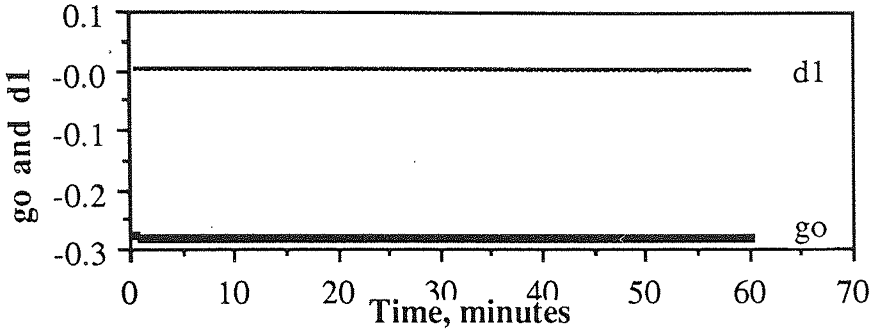
PSV-STC - dashed lines, PI - continuous line; ΔTc = 0.5 minutes
Figure 9.8. Top tray temperature control: PI vs. PSV-STC



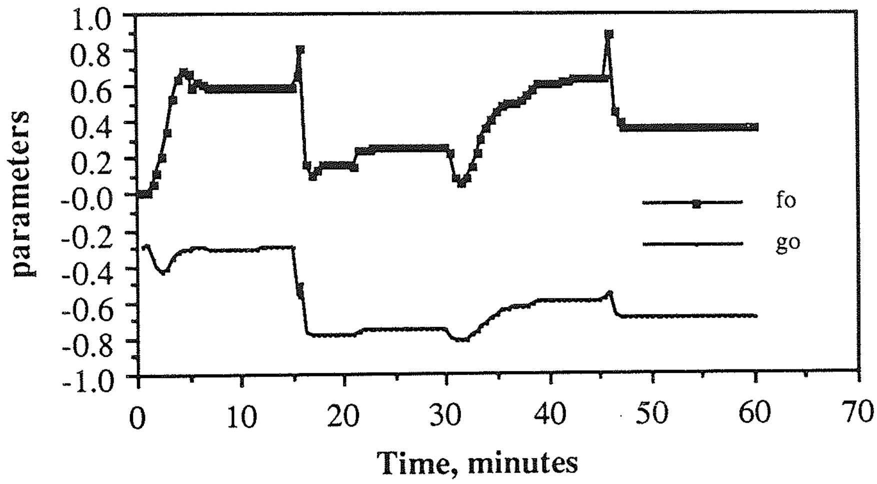
ΔTc = 0.5 minutes
Figure 9.9 Comparison of ISV-STC with PSV-STC



PSV-STC



PSV-STC



ISV-STC

Figure 9.10 Parameter estimates for ISV-STC

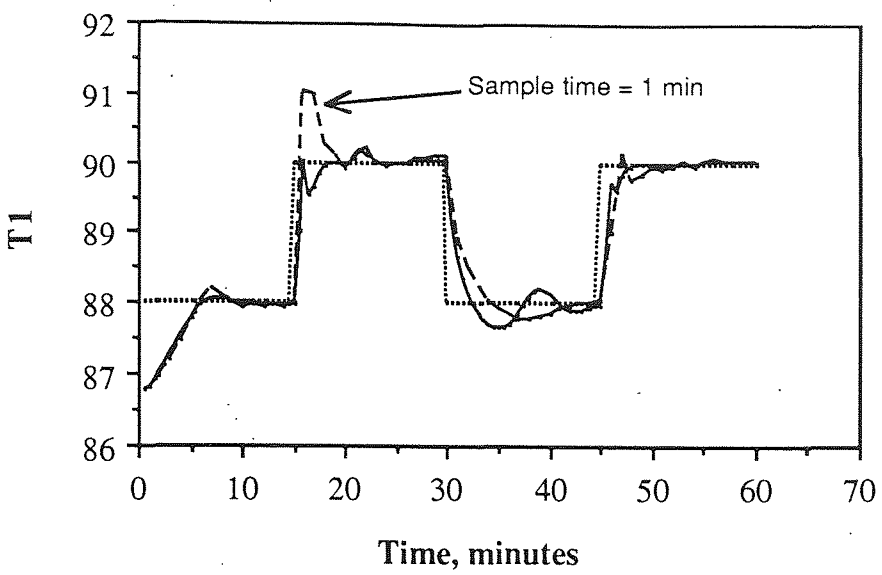


Figure 9.11 Effect of increase in the the control inteval, ΔTc, to 1 minute on the performance of ISV-STC

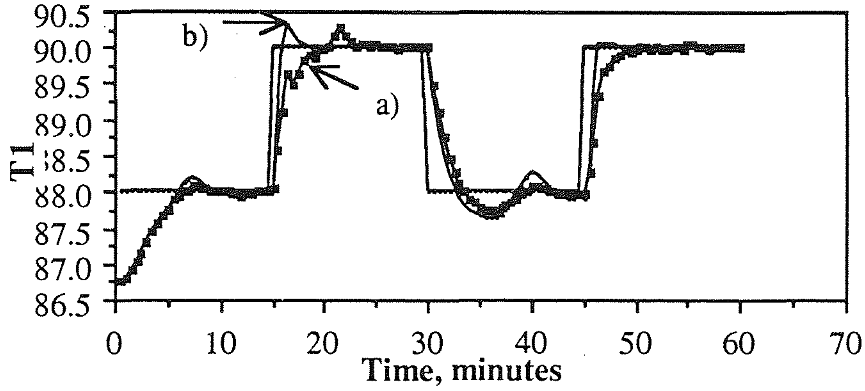


Figure 9.12 Effect of a) slower reference model and b) larger control weighting on the performance of ISV-STC: ΔTc = 0.5 minutes

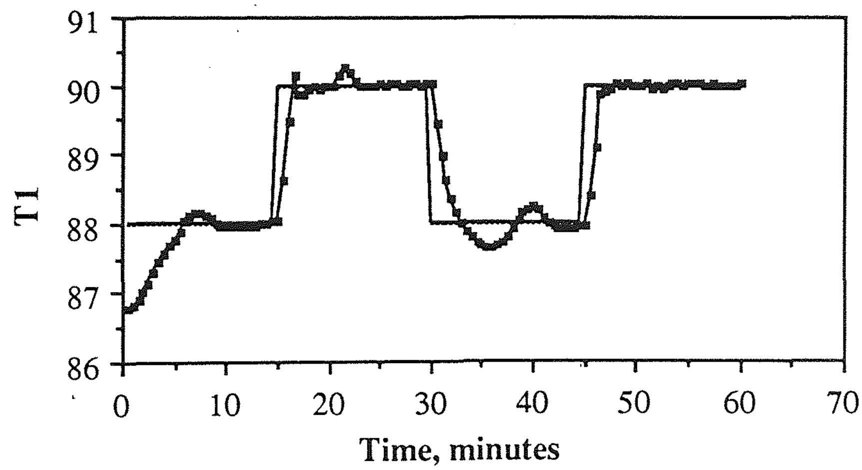


Figure 9.13 Performance of ISV-STC: First order **P** and **R** each with a time constant of 0.5 minutes **Q** = $-0.6\Delta_1$, ΔTc = 0.5 minutes

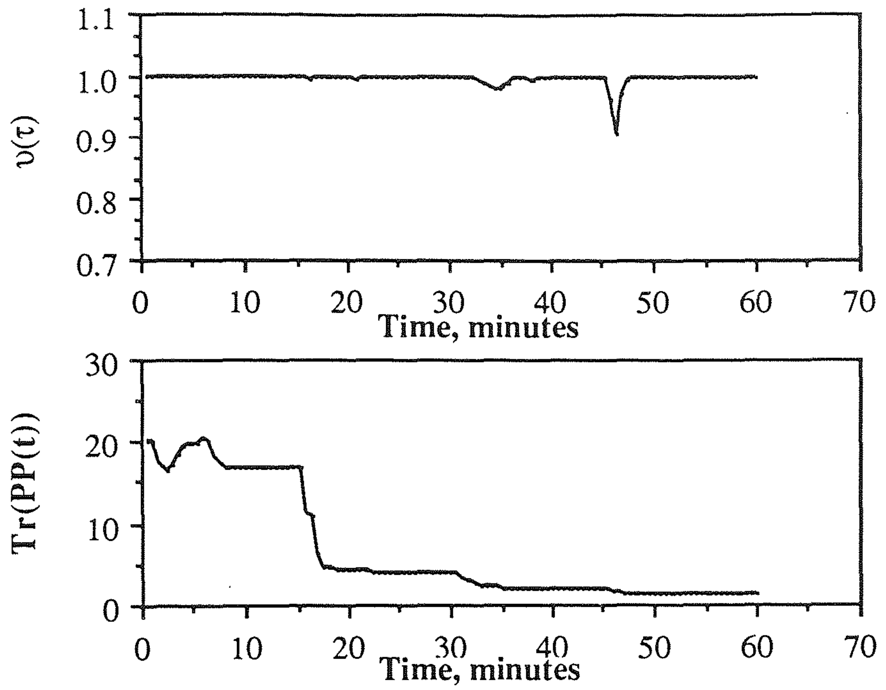


Figure 9. 14 Behaviour of estimator for ISV-STC using VFF3 algorithm

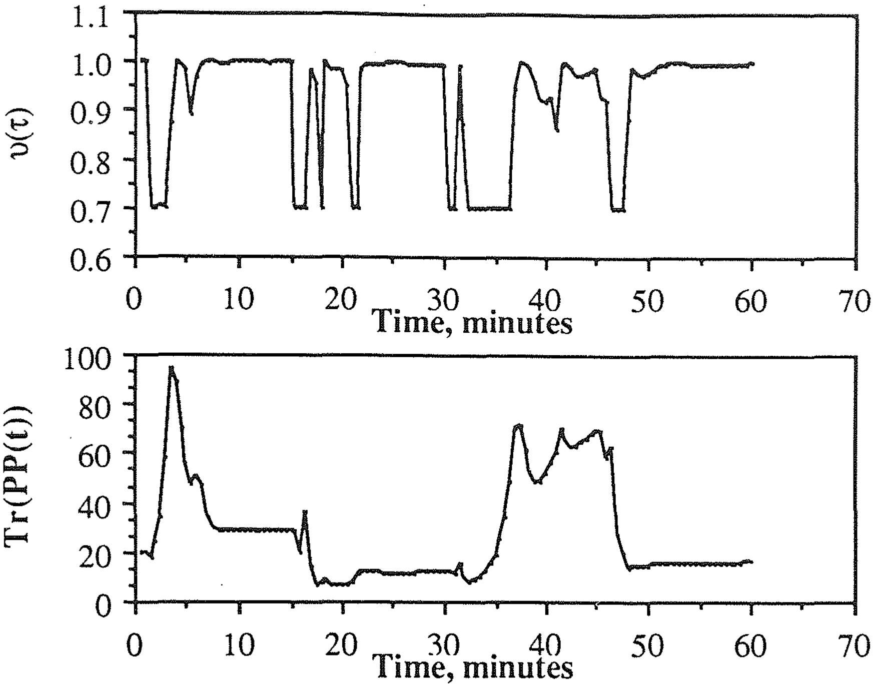


Figure 9.15 Behaviour of estimator estimator for ISV-STC using VFF1 algorithm with $\Sigma o = 0.01$

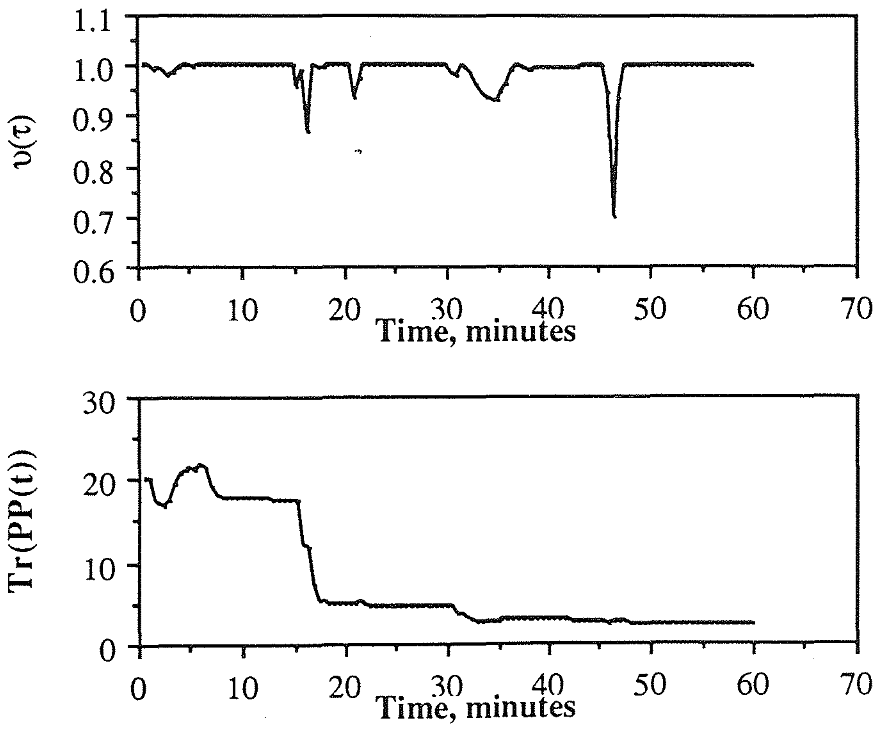


Figure 9.16 Behaviour of estimator estimator for ISV-STC using VFF1 algorithm with $\Sigma o = 0.5$

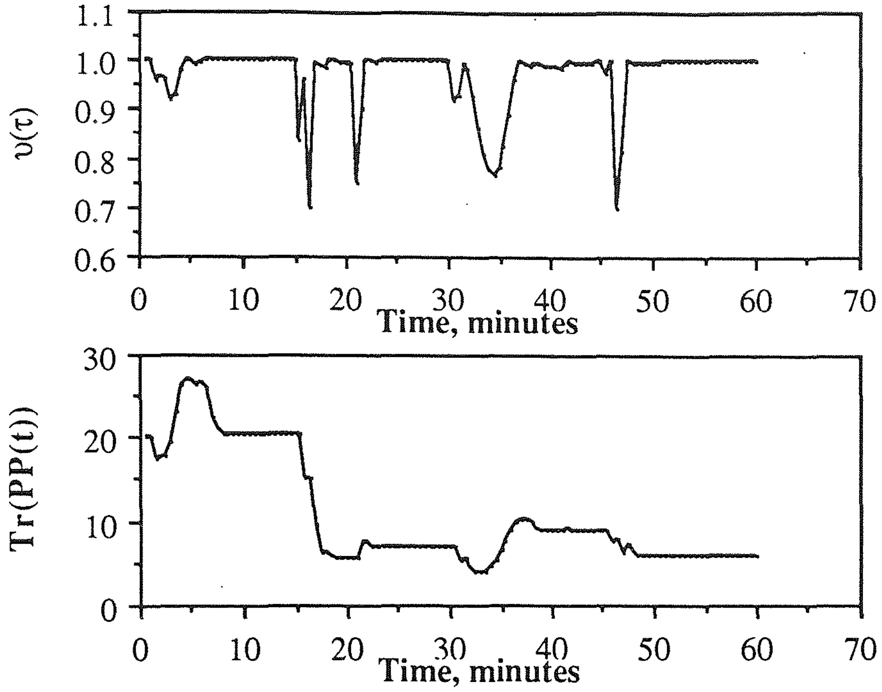


Figure 9.17 Behaviour of estimator estimator for ISV-STC using VFF2 algorithm with No = 0.1

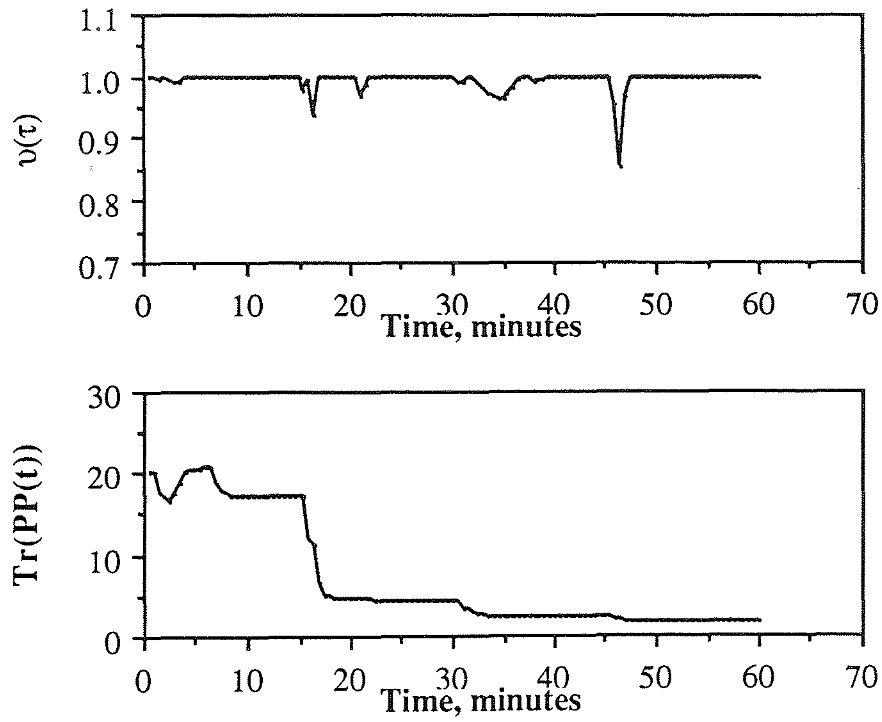


Figure 9.18 Behaviour of estimator estimator for ISV-STC using VFF2 algorithm with No = 1.0

9.4 Simultaneous control of the top tray and the bottom tray temperatures

The performance of the multiple loop self tuning controllers MD1-STC, MD2-STC and MD3-STC, were all assessed for setpoint tracking and load disturbance rejection. Multiple loop PI control, with and without steady state decoupling, was also performed. The Kc and τ_i used were -2.0 l/hr/°C and 3 minutes, respectively for the T_1-Lr loop and 0.08 KW/°C and 1.2 minutes, respectively, for the T_{10}-Qrb loop. Approximate values of the steady state decouplers were calculated using Equations 2.36 and 2.37 and the gain matrix

$$\mathbf{G} = \begin{bmatrix} G_{11} G_{12} \\ G_{21} G_{22} \end{bmatrix} = \begin{bmatrix} -0.5 & 3.56 \\ -9.53 & 80.77 \end{bmatrix} \qquad 9.2$$

whose elements were obtained from Table 5.4.

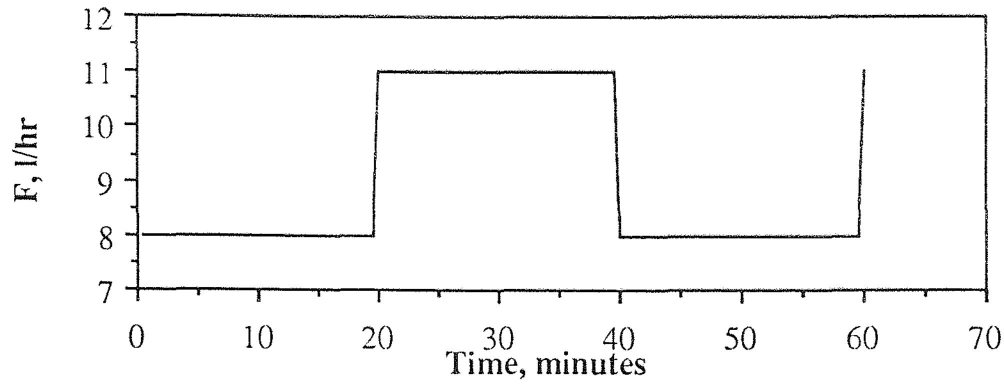
The following were prescribed in the multiple loop self tuning controllers: $\mathbf{Q}$ = diag $(-0.5, 15)\Delta_1$, $\mathbf{P}(z^{-1}) = \mathbf{R}(z^{-1}) = \mathrm{I}$, $\theta_1 = \theta_2$ = (0.5,-0.3,2.3,0) for MD1-STC and MD2-STC, $\theta_1(0)$ = (0.5,-0.3,0), $\theta_2(0)$ = (0.5,2.3,0.0) for MD3-STC, $PP_1(0) = PP_2(0) = 0.1\mathrm{I}$ and the VFF3 variable forgetting. The subscript i denotes the control loop, where i = 1 represents the T_1-Lr loop and i = 2 represents the T_{10} - Qrb loop, respectively.

For servo control the following changes in the setpoints were induced on the column simulator every 20 minutes :

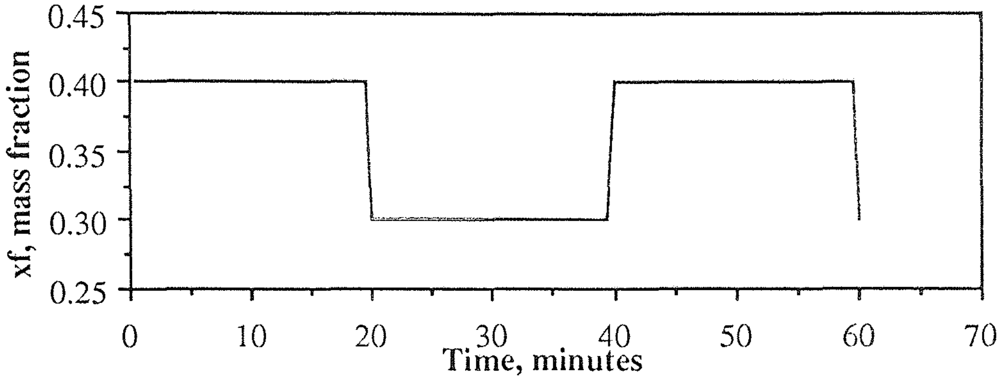
(i) a sequence of positive and negative changes of 1°C on the setpoint of T_1

(ii) a sequence of positive and negative changes of 2°C on the setpoint of T_{10}

For regulatory control the load disturbances made on the simulator are shown in Figure 9.19. The ΔTc = 0.5 and the period of 15 minutes was allowed to initially tune the self tuning controller parameters. Table 9.3 contains the IAEs for the simulations reported.



a) Feed flow disturbance



b) Feed composition disturbance

Figure 9.19. Load disturbances for simultaneous control of the top and bottom tray temperature: Load disturbance rejection

For servo control, the following observations were made:

1) The multiple loop PI control system gave good stable control, but the performance was degraded by 46% when the steady state simplified decouplers were introduced. This degradation was particularly significant in the bottom loop as is evident from the corresponding IAE on Table 9.3 and in Figure 9.20.
2) The PMD1-STC gave much tighter control compared with the multiple loop PI (Figure 9.21) and hence the improvement of 35% over PI control was achieved, while IMD1-STC gave a 29% improvement.

Table 9.3 Integrated Absolute Error (IAE) for the simultaneous control of the top tray and bottom tray temperatures

Controller	IAE		Total IAE
	Top Tray	Bottom Tray	
SERVO CONTROL			
PI	14.73	19.32	34.05
PI + SS Decoupling	17.71	32.24	49.95
PMD1-STC	6.11	15.99	22.1
PMD2-STC	5.35	15.84	21.19
PMD3-STC	15.65	16.11	31.76
PMD3-STC (*)	22.61	18.64	41.25
IMD1-STC	6.83	17.24	24.07
IMD2-STC	6.93	18.4	25.33
IMD3-STC	7.3	17.84	25.14
DISTURBANCE REJECTION			
PI	6.84	13.08	19.92
PI + SS Decoupling	12.22	12.24	24.46
PMD1-STC	5.22	12.31	17.53
PMD2-STC	5.5	12.69	18.19
PMD3-STC	9.21	13.25	22.46
IMD1-STC	5.64	12.60	18.24
IMD2-STC	5.13	13.39	18.52
IMD3-STC	5.25	12.68	19.93

(*) Denotes exclusion of "1in data vector" for estimation of unknown disturbances.

NB. All self tuning simulations were done using PP(0) = 0.1I, $\mathbf{P}(z^{-1}) = \mathbf{R}(z^{-1}) = I$, $\mathbf{Q}(z^{-1}) = \mathrm{diag}(-0.5, 15)\Delta_1$.

3) Table 9.3 shows that PMD1-STC and PMD2-STC performed better than their respective incremental counterparts, IMD1-STC and IMD2-STC. Figures 9.22a and 9.22b shows this clearly for MD2-STC. As regards the positional forms, the PMD3-STC was worse than the other two and more so when the bias terms in **d** were not estimated .

4) Unlike the MD1-STC and MD2-STC, IMD3-STC was significantly better than PMD3-STC, particularly in the control of the top tray temperature (see Figures 9.22c and 9.22d). The IAE for the incremental form is 21% better than the positional form, ond 39% better when compared with the case with no estimation of bias terms.

5) Figure 9.23 shows the performances of IMD1-STC and IMD3-STC where the setpoint of one output is changed, keeping the setpoint of the other constant. After a change in the setpoint of one output, the peak of the deviation of the other output from its setpoint was always higher under IMD3-STC. This shows the improvement in control provided by interaction compensation in IMD1-STC.

Some similar observations as in the servo case were also made in the case of regulatory control. Steady state decoupling degraded the performance of the multiple loop PI controllers by 23% (see Figure 9.24 for graph). The PMD1-STC and IMD1-STC performed better than the multiple loop PI control by 12% and 8%, respectively. The transient behaviour under IMD1-STC was more oscillatory compared with control under PMD1-STC, particularly in the response of the top tray temperature (Figure 9.25). Like the servo case, the PMD1-STC and PMD2-STC perfomed better than their incremental forms, but the IMD3-STC was much better than PMD3-STC (Figures 9.26) and hence the 11% improvement in the IAE.

Like the SISO case, the parameter estimates for the positional form remained virtually constant after initial tuning in period, while, they changed significantly after disturbances or setpoint changes entered the system under incremental control (Figures 9.27a and 9.27b for servo control). Thus, the incremental forms generally gave responses with larger transients and more oscillatory behaviour, as is shown in

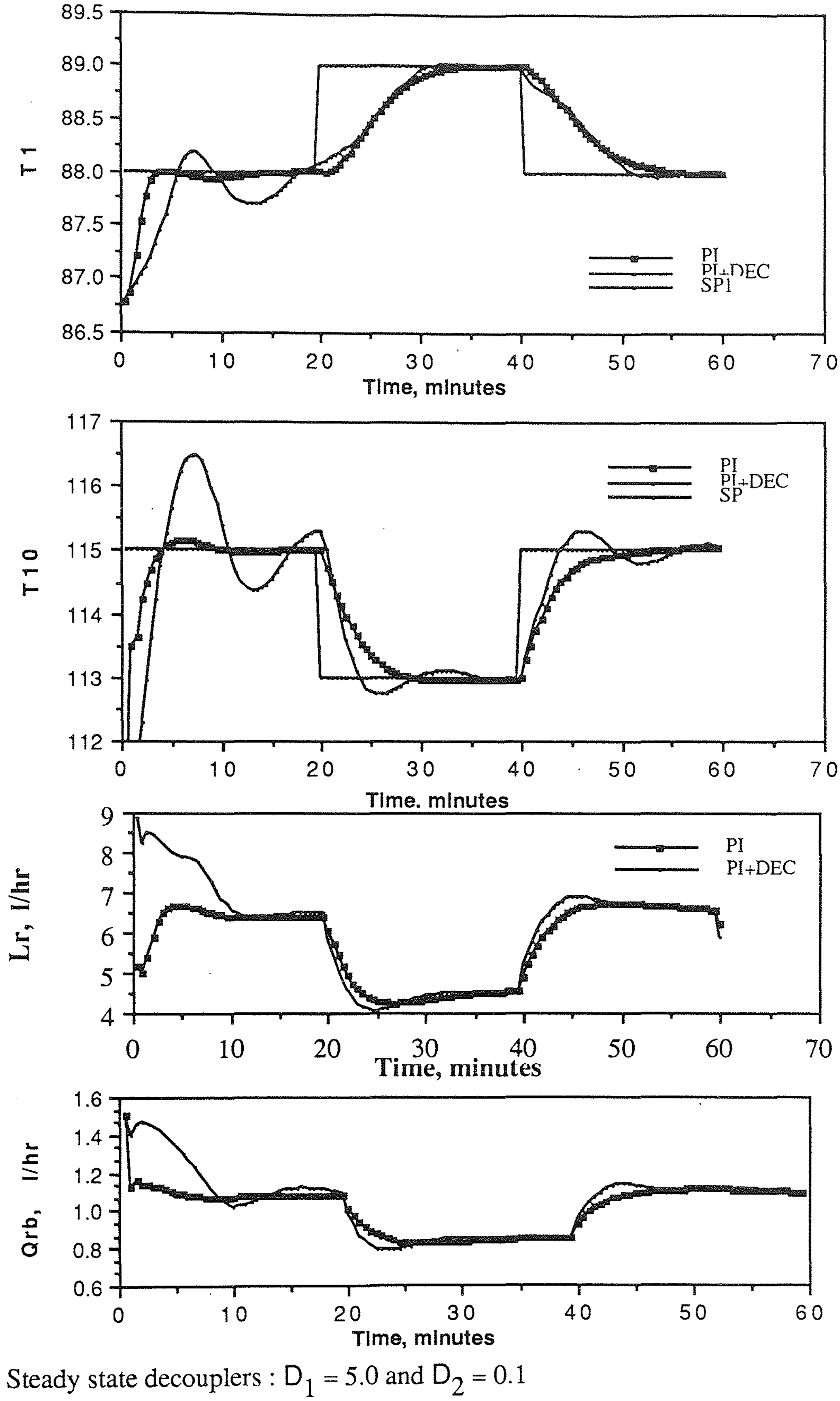
Figure 9.24 for regulatory control under MD1-STC. Unlike the SISO case, however, the incremental forms if the multiple-loop STCs gave higher IAEs compared to the positional forms, except for the MD3-STC where the incremental form was significantly better than the positional form.

The reason why both PMD1-STC and PMD2-STC performed much better than PMD3-STC can be attributed to the fact that the former two algorithms provide control loop decoupling, while the latter does not. In the absence of control loop decoupling as in PMD3-STC, the interaction effect of one loop on the other loop is now in the form of unmeasured, or unknown, load disturbances; the effects of which each loop must regulate against. It has been shown earlier by linear simulations (Figures 9.2 - 9.3) that a positional STC performs badly when regulating against unknown or unmeasured load disturbances as it relies on the accurate estimation of the bias or disturbance effects; an incremental STC, on the other hand, performs very well (see Figure 9.4) as its control law provides implicit estimates of the disturbance effects as discussed in Section 8.21 and Section 9.2. This characteristic could be used to explain why the IMD3-STC performed much better than PMD3-STC for both servo and regulatory control.

The degrading effect that the steady state decouplers had on the performance of the multiple loop PI controllers is due to the non-linear nature of the column simulator. Therefore, the decoupling elements used with the PI controllers are in error and this caused the poor performance with steady state decoupling. Such problems were not evident with MD1-STC and MD2-STC, which also provide control loop decoupling, most probably because the self tuning nature of these two control algorithm has the ability to compensate for the errors in their decoupling elements.

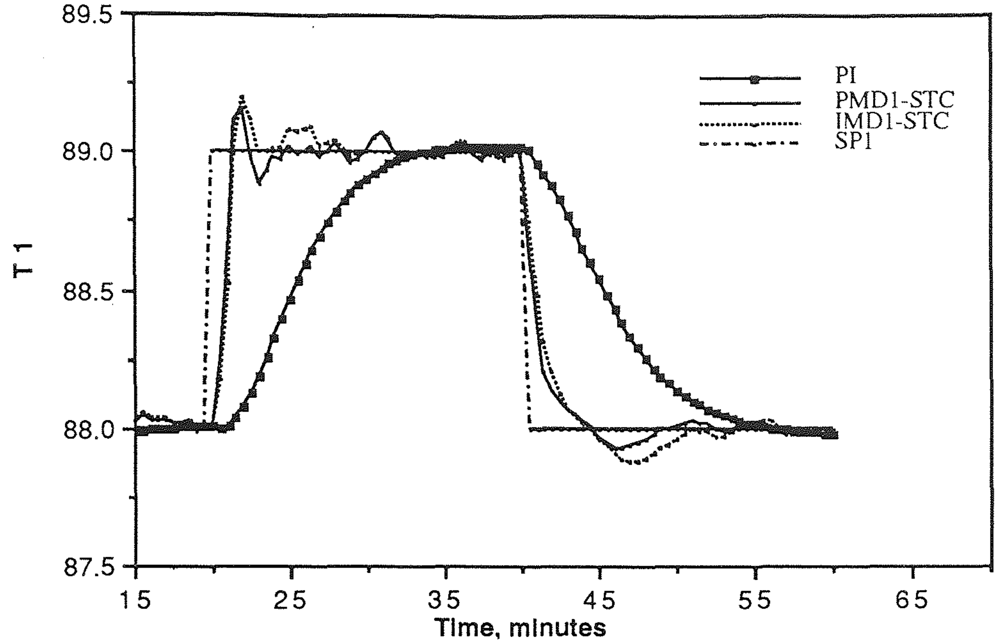
9.4.1 Summary

On the basis of the IAEs on Table 9.3, the PMD1- STC is the best choice for the simultaneous control of the top tray and bottom tray compositions. However, for regulation against unknown load disturbances the IMD1-STC is preferred because of the better regulatory property of incremental algorithms.



Steady state decouplers : $D_1 = 5.0$ and $D_2 = 0.1$

Figure 9.20 Simultaneous control of the top tray and bottom tray temperatures (Servo control): Multiple loop PI vs Multiple loop PI + steady state decoupling



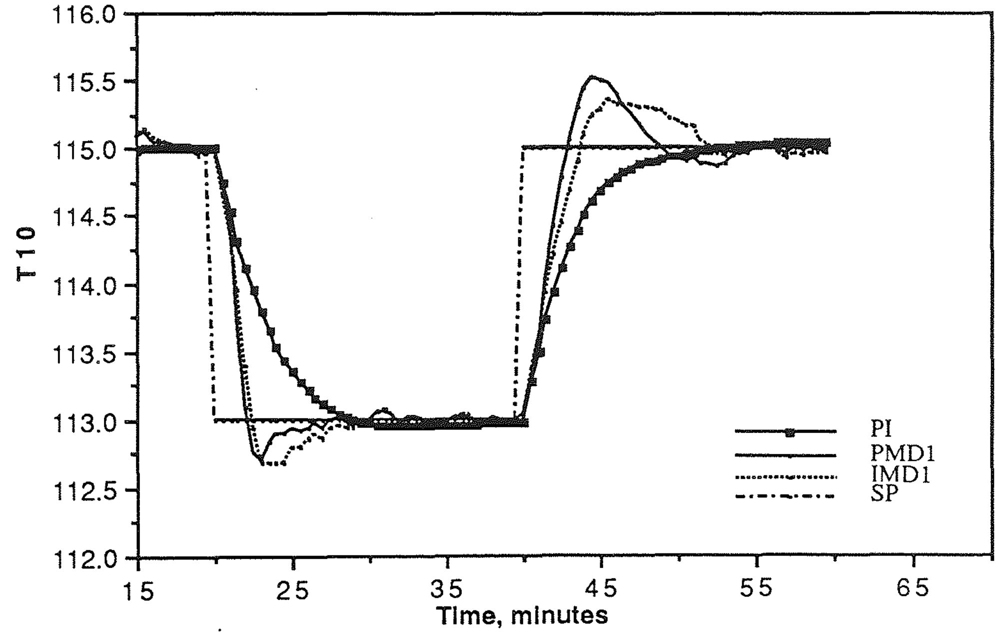


Figure 9.21 Simultaneous control of the top and bottom tray temperatures: Comparison of multiple loop PI, PMD1-STC and IMD1-STC

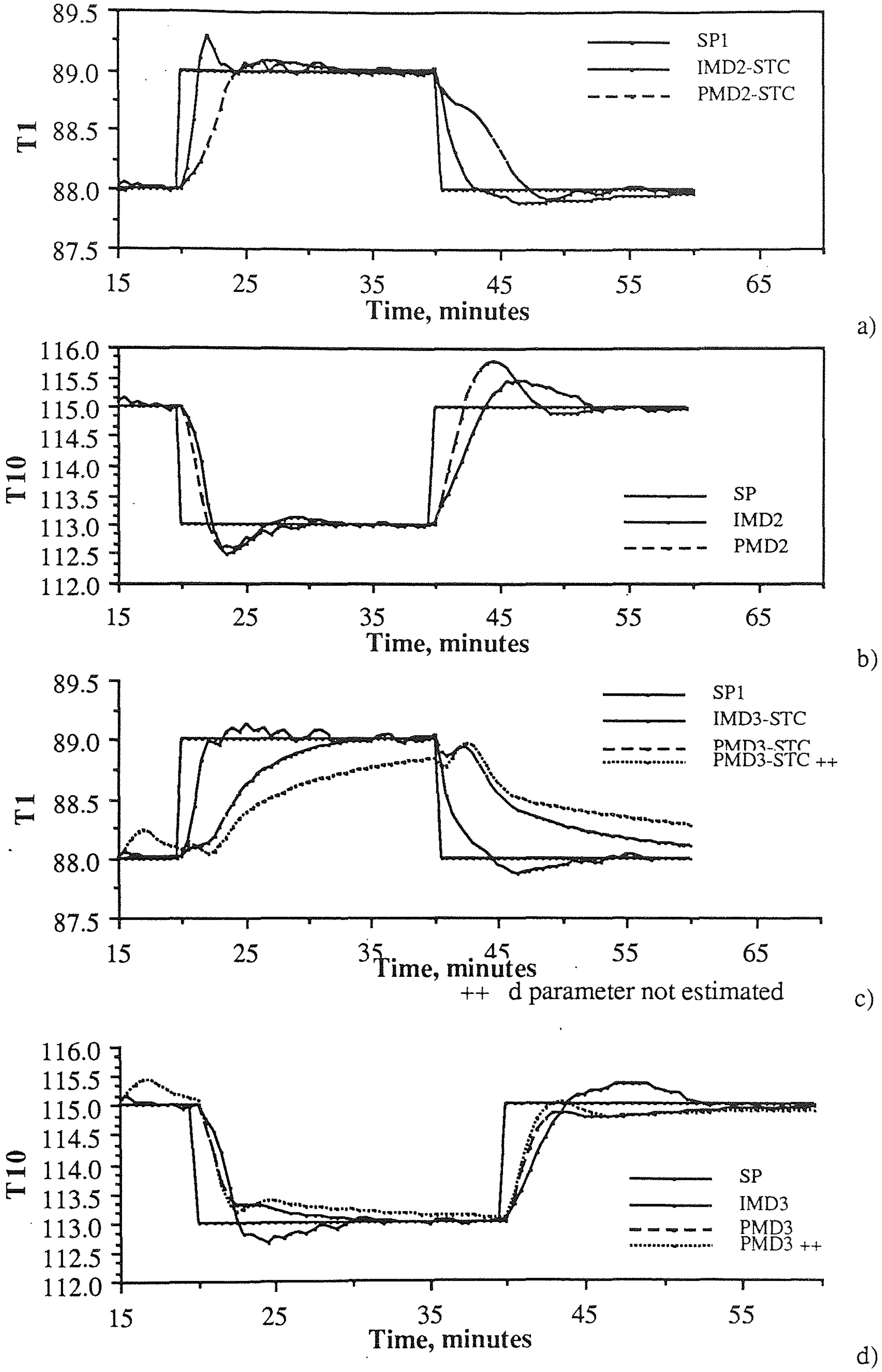


Figure 9.22 Simultaneous control of the top tray and bottom tray temperatures (Servo control): Comparison of the positional and incremental MD2-STC and MD3-STC

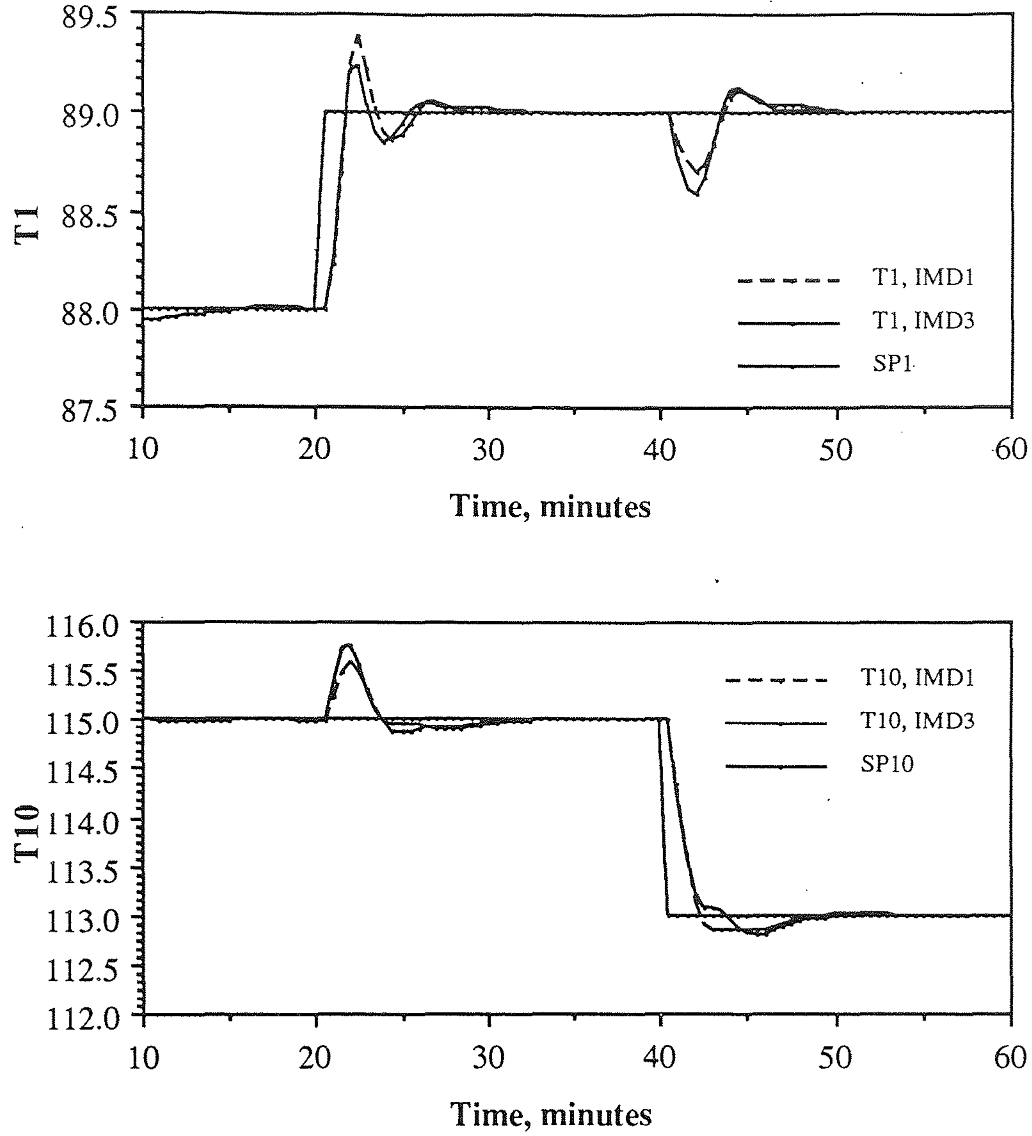


Figure 9.23 Simultaneous control of the top tray and bottom tray temperatures (Servo control): Comparison of IMD1-STC and IMD3-STC to demonstrate the benefit of interaction compensation.

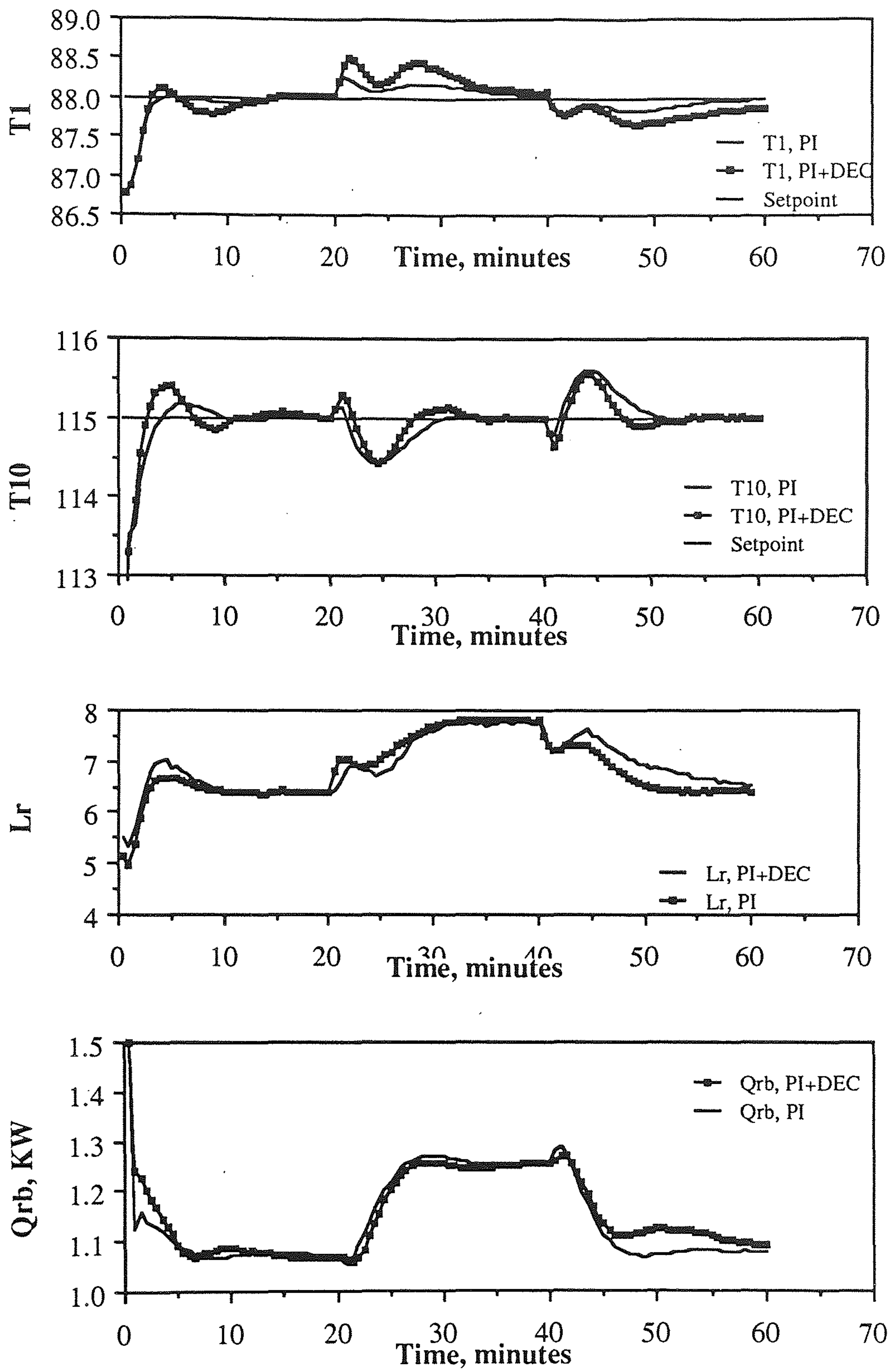
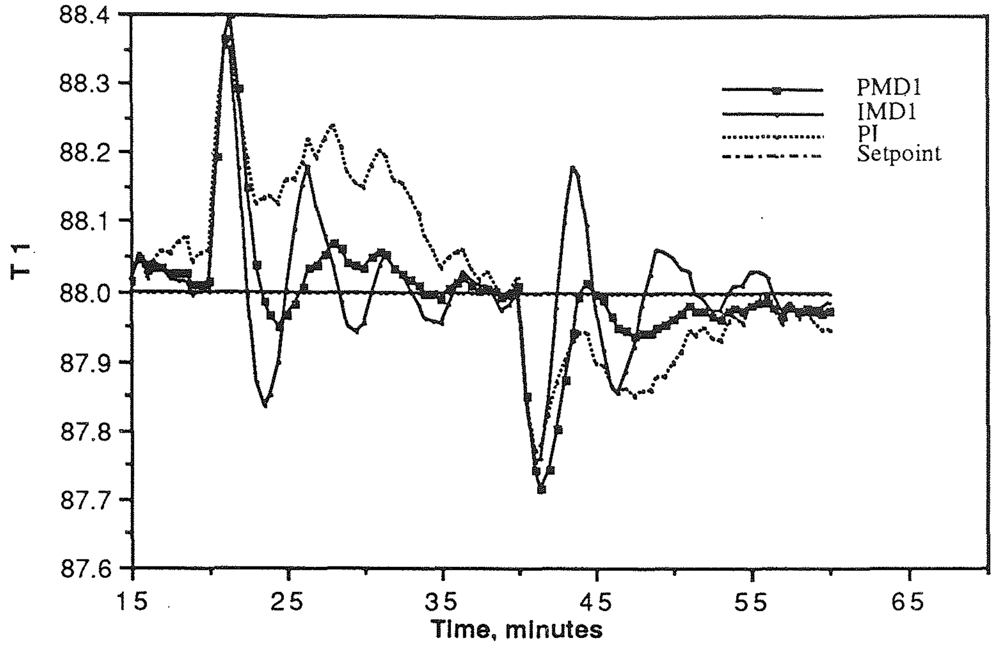


Figure 9.24. Simultaneous control of the top and bottom tray temperatures: Load disturbance rejection PI vs PI + steady state simplified decoupling



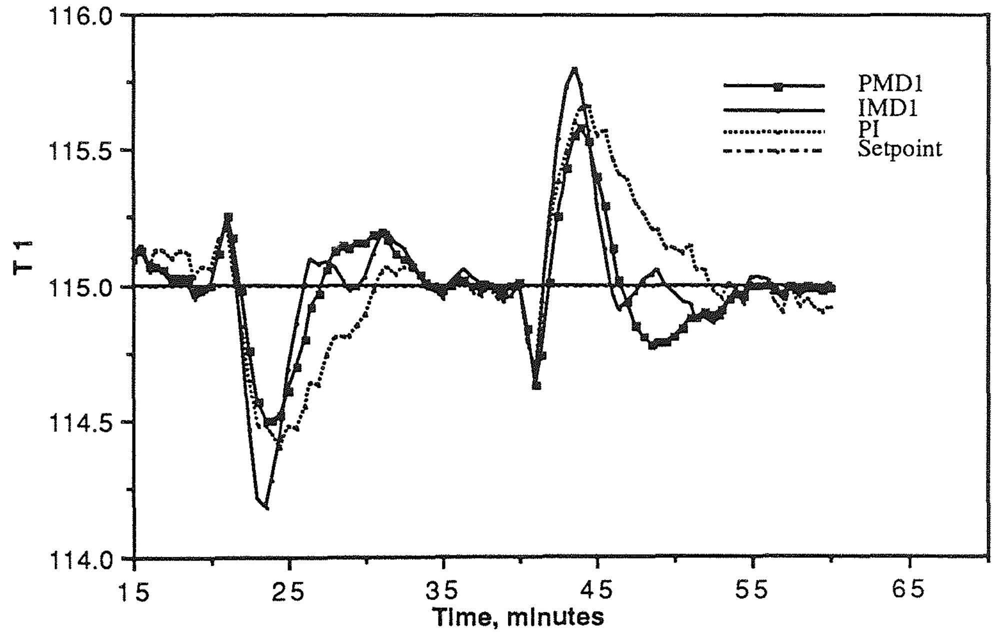


Figure 9.25. Simultaneous control of the top and bottom tray temperatures: Load disturbance rejection. Comparison of PMD1-STC, IMD1-STC and multiple loop PI

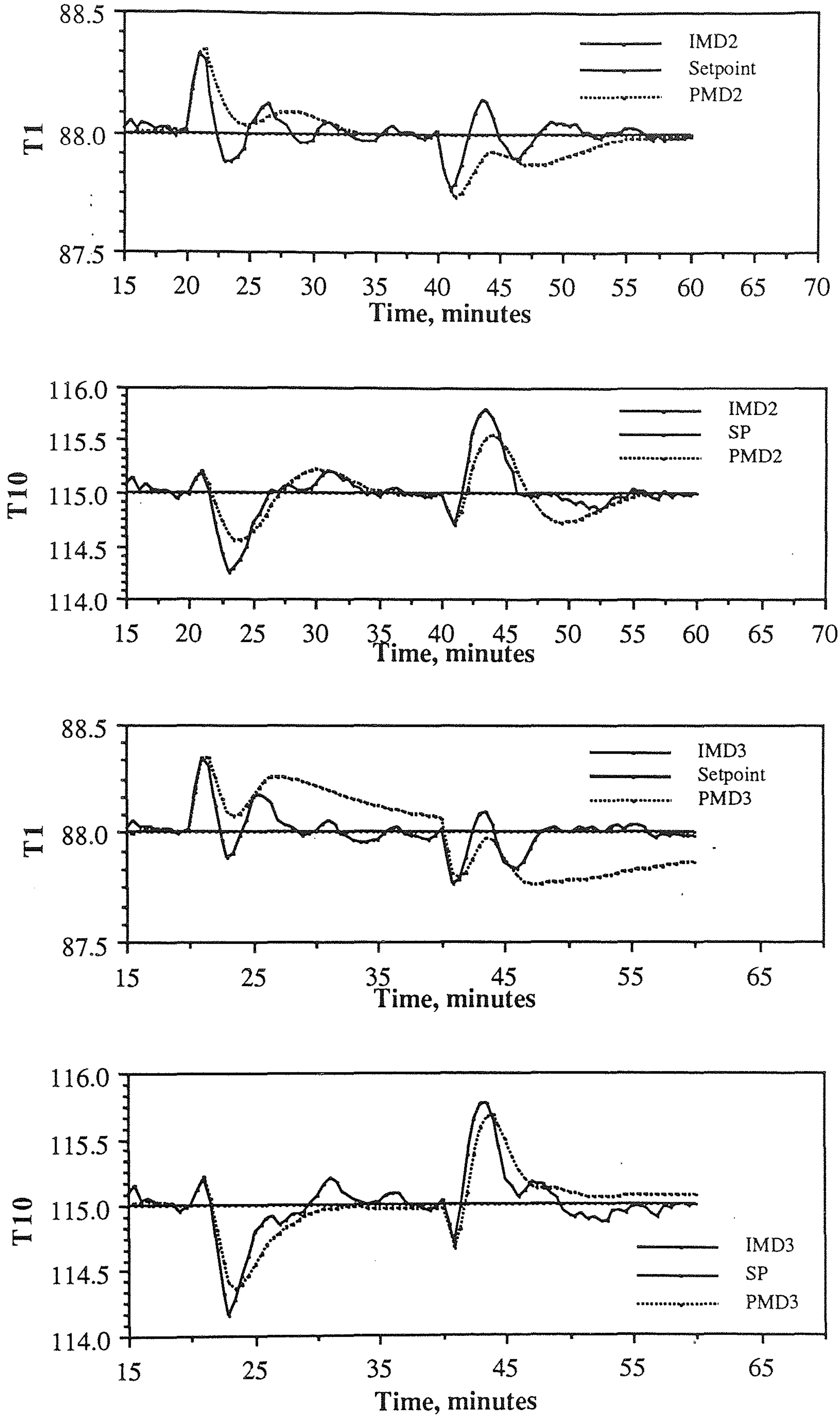


Figure 9.26 Regulatory performances of the positional and incremental forms of MD2-STC and MD3-STC

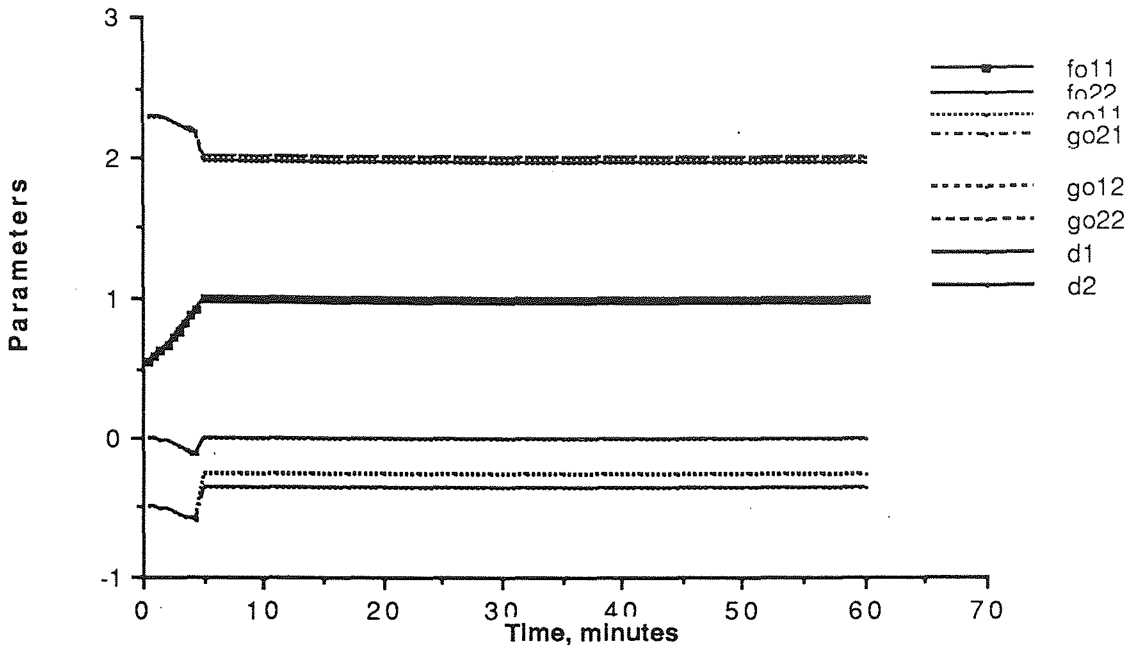


Figure 9.27 a) Parameter Estimates for PMD1-STC for servo control

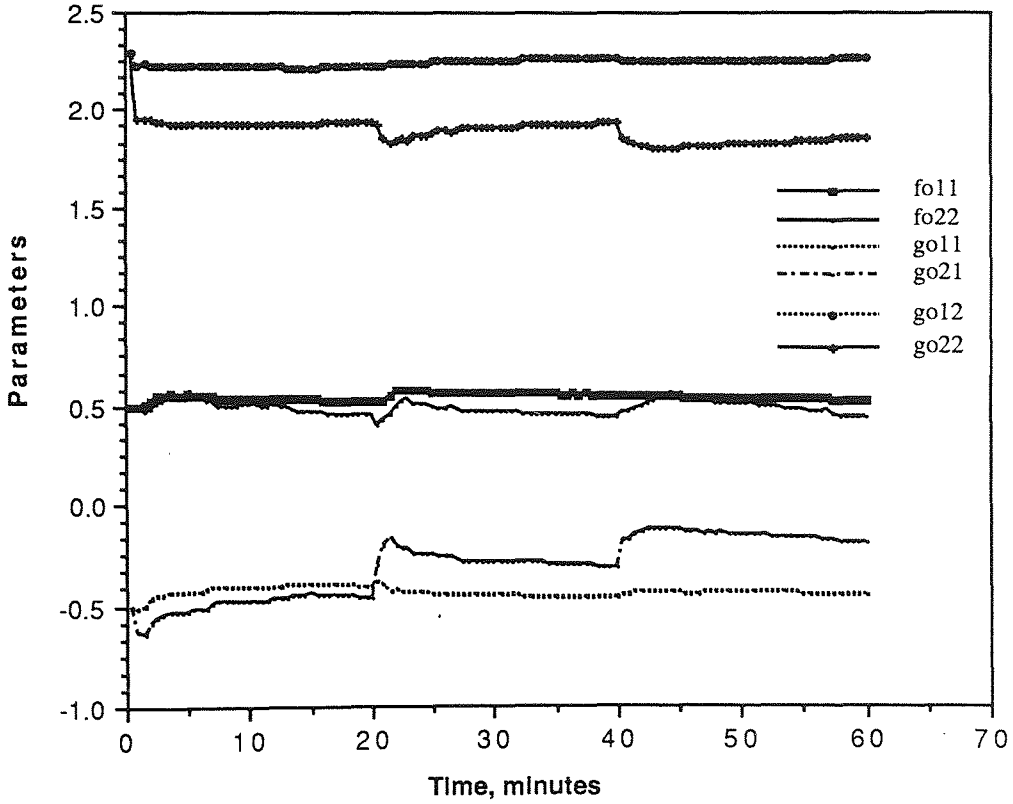


Figure 9.27 b) Parameter Estimates for IMD1-STC for servo control

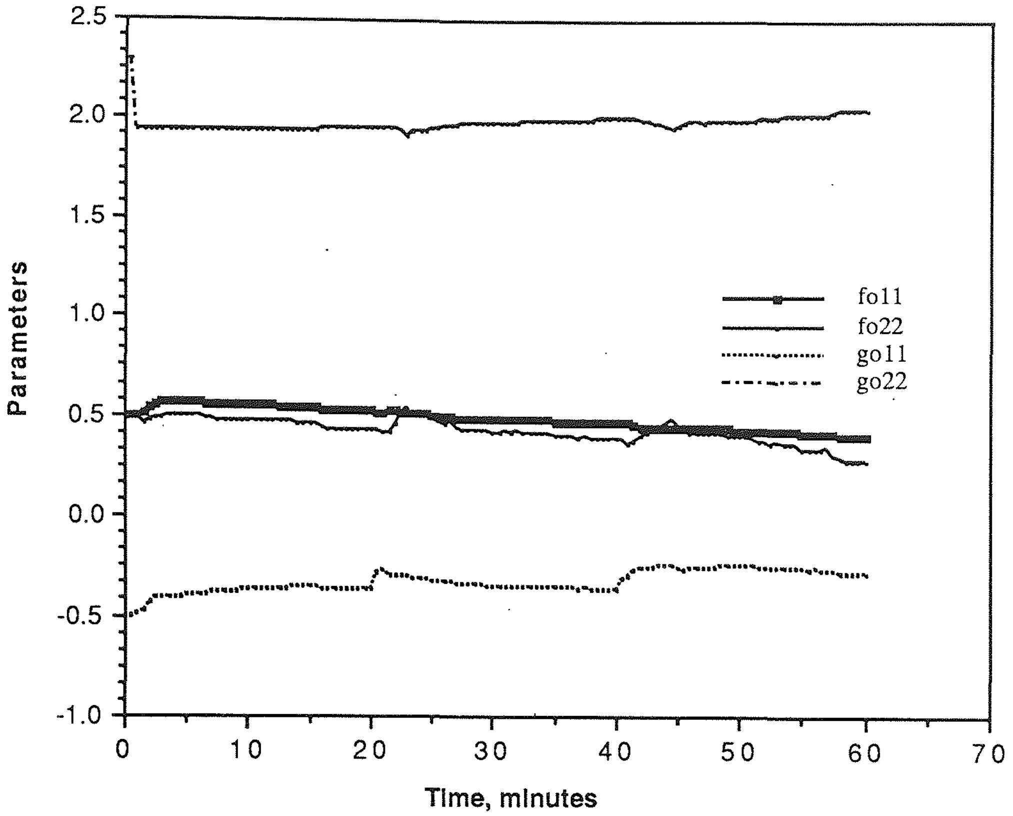


Figure 9.27 c) Parameter estimates for IMD3-STC for servo control

9.5 Application of the Parameter Correction (PC) and Simplified Parameter Correction (SPC) methods

The parameter correction (PC) method was formulated in Section 8.3. It is an approach based on the idea of Ossman and Kamen (94) discussed in Section 2.9.8. The aim is to prevent the parameter estimates of the self tuning controller from attaining bad values which may cause poor performance controller. The method involves moving any parameter back into the admissible range, should the parameter violate specified bounds.

A simpler form of the algorithm referred to as the simplified parameter correction (SPC) method was formulated in this work and is described in Section 8.3.1. The SPC uses a constant rate of correction and can be viewed as a method that compensates for drifts in the parameter estimates. An example of a situation where parameter correction could improve the quality of control is shown in Figures 9.28a (corresponding control actions in Figure 9.28b) where IMD1-STC was used for servo control. In this simulation $PP_1(0) = PP_2(0) = 10I$ which is 100 times larger than those used in the simulations discussed in the previous section, and $\theta_1(0) = \theta_2(0) = (0.0,-0.3, 2.0)$ which are different initial parameter estimates. The corresponding parameter estimates are shown in Figure 9.29. Immediately after the switch to self tuning control, large and oscillatory control actions were produced (Figure 9.28b) uptill about t = 30 minutes. During this period, large and violent changes in some of the parameter estimates resulted, as is evident in the **F** parameters and in g_0^{21} in Figure 9.29, causing the poor transient behaviour of the closed loop system (Figure 9.28a). A comparison of this simulation with that shown in Figure 9.28 (Figure 9.29 vs 9.27b for the parameter estimates) clearly demonstrates that the performance of self tuning controller on the non-linear column simulator is significantly affected by the choice of the initial covariance matrix, and the initial parameter estimates as well. The IAE for this simulation is 54.42 (Table 9.5), which is more the twice that for the case with smaller covariance matrix in Table 9.3. Performances such as this necessitated the need to use good initial parameters and, therefore, a small initial covariance matrix to indicate confidence in the initial parameters in the simulations discussed in Section 9.4.

Experience in the application of the self tuning controllers on the column simulator showed that the g_0^{11} and g_0^{21} estimates should have -ve values corresponding to the steady state effects of Lr to the tray temperatures and the g_0^{12} and g_0^{22} should have +ve values corresponding to the steady state effects of the Qrb on the tray temperatures; the **F** parameters should, ideally have +ve values. This was used to guide the selection of the bounds of the parameters given in Table 9.4 below

Table 9.4 Bounds on the controller parameters for use with the parameter correction methods.

Bound	f_0^{11}, f_0^{22}	g_0^{11}, g_0^{21}	g_0^{12}, g_0^{22}
Upper	1	-0.01	5
Lower	0.01	-2.0	1.0

9.5.1 Evaluation of the parameter correction methods

Simulation studies were performed to assess the potential benefits of using the PC and SPC algorithms to remedy difficulties such as those that caused the poor control shown in Figure 9.28. The improvement or degradation in performance, in terms of IAE, is measured against the performance of the case without correction in Figure 9.28. It was assumed that the g_0^{11} and g_0^{22} parameters, together with g_0^{12} and g_0^{21}, are the key parameters in the multiple loop self tuning control algorithms, as they are the main factors that determine the gains of their respective control loops. For example, in the MD3-STC algorithm the g_0^{11} is inversely proportional to the T_1-Lr loop gain and g_0^{22} is inversly proportional to the T_{10}-Qrb loop gain. In the simulations, parameter correction was made to commence at t = 10 minutes, to avoid correction in the initial stages were large prediction errors and large bursts in the estimator may occur.

Table 9.5 Integrated Absolute Error (IAE) for the simultaneous control of the top tray and bottom tray temperatures using IMD1-STC combined with the parameter correction.

Specifications	IAE		Total	
	Top	Bottom	IAE	Figure
IMD1-STC	12.43	41.99	54.42	9.28
IMD1-STC+PC, $\alpha = 0.15$, (†)	14.87	74.9	89.77	9.30
IMD1-STC+PC, $\alpha = 0.15$, (®)	12.8	30.68	43.48	9.32
IMD1-STC+SPC, $\mu = 0.15$, (†)	9.74	24.58	34.32	9.34
IMD1-STC+SPC, $\mu = 0.2$, (†)	9.58	21.64	31.22	9.34
IMD1-STC+SPC, $\mu = 0.2$, (®)	7.3	15.36	22.66	9.36
IMD1-STC+PC, $\alpha = .15$, (®) (***)	5.94	16.3	22.24	9.38
IMD1-STC (best case)	6.83	17.24	24.07	9.21

NOTES	
All simulations used $PP_1(0) = PP_2(0) = 10I$	
(†)	denotes bounds of only g_0^{11} and g_0^{22} are specified
(®)	denotes bounds of g_0^{11}, g_0^{12}, g_0^{12} and g_0^{22} are specified
(***)	denotes "correction term" used is $-\alpha PP(t-1)\mathbf{f}(\theta(t-1))$, and forgetting factor selected to keep trace of covariance matrix constant as 25.

The PC method, with the correction rate chosen as $\alpha = 0.15$, was combined with IMD1-STC and the bounds of only g_0^{11} and g_0^{22} were specified in the PC algorithm. The transient behaviour (Figure 9.30) was poorer than when correction was absent and hence the 65% degradation in performance in terms of the IAE (Table 9.5). Although the outputs had settled at their setpoints by t = 40 minutes, subsequent changes in the setpoints resulted in the saturation of the Qrb, generation of large changes in Lr, and hence the large excursions of the outputs from their setpoints.

The g_0^{21}, f_0^{22} and g_0^{21} parameters exhibited severe oscillatory behaviour immediately after the switch to self tuning control and after the changes in the setpoints (see Figure 9.31) which coincides with the period of poor control. It is evident from the trajectory of g_0^{22} that there was an initial period of effective correction of the parameters after which there was virtually no correction at all. This was because the magnitude of the covariance was large enough at the initial stages (t = 10 minutes to t = 20) after which the magnitude rapidly became relatively very small so that correction became negligible. The g_0^{22}, therefore, maintained a large -ve value from about t = 20 minute shortly after the switch to self tuning control.

When the bounds of all the four **G** parameters (g_0^{11}, g_0^{22}, g_0^{12} and g_0^{21}) were specified, there was an improvement of 20% of the performance of IMD1-STC, in contrast to the degradation that resulted in the previous case (Figure 9.30). Figure 9.32 shows this graphically. After the period of poor transients from t = 15 to t = 25 minutes, the performance of IMD1-STC was much better than in the previous cases (Figures 9.28 and 9.30) and, thus, the smaller IAE. As with the previous case, there was also an initial period of effective correction of the parameters after which correction became negligible because the covariance matrix became small. The g_0^{22} also maitained a large -ve value (see Figure 9.33), but, overall, the behaviour of the parameter estimates were less violent than in Figure 9.31. The 20% improvement in the IAE was achieved because all the four **G** parameters were moved closer to their specified bounds during the initial period when there was effective correction of the parameters, instead of only two parameters moved closer to the bounds in the previous case.

In the SPC method, which is a simpler form of the PC method, the correction of a parameter can be done independently and the effectiveness of the correction depends only on the correction rate, μ, and not on the magnitude of the covariance matrix. Figure 9.34 (Figure 9.35 shows the parameters) shows the result usiing the SPC with $\mu = 0.15$ with the bounds of only g_0^{11} and g_0^{22} specified. Unlike the previous cases with the PC method, the outputs were maintained at their setpoint after the switch to self tuning control, until the setpoint changes at t = 20 minutes. Large overshoots then occurred after these changes, but overall the closed loop responses were much more desirable than those in Figures 9.28, 9.30 and 9.32. This demonstrates that the fast setpoint tracking capabilities of IMD1-STC (shown in Figure 9.21) was restored by the introduction of the SPC to modify the trajectory of only 2 out of the 6 controller parameters of IMD1-STC. Table 9.5 shows an improvement of 37% compared with the degradation of 65% when the PC was used (Figure 9.30) in a similar situation, and a 37% improvement compared with 20% improvement when the PC method was used with the bounds of all the four **G** parameters specified (Figure 9.32).

Table 9.5 shows that the performance of IMD1-STC with SPC using $\mu = 0.2$ (graphs superimposed on Figure 9.34) is further improved (a 42% improvement) with the increase in the correction rate. With this μ, specifying the bounds of all the four **G** parameters gave a 58% improvement in the IAE which is much better than the 42% in the previous case. The plots for this is shown in Figure 9.36 and the corresponding parameter estimates are in Figure 9.37.

From the above discussions it can be concluded that the SPC method is better than the PC method to improve the performance of the self tuning controllers in the case where the parameter estimates attain bad values. The results also confirm the presumption made in the previous chapter that the success of the PC method depends on the magnitude of the covariance matrix and this dependence can be a limitation in the applicability of the method.

It may be possible to improve the performance of the PC method in the situations simulated in Figures 9.30 and 9.32, if a method can be devised which maintains the covariance matrix at a level where effective correction of the parameters will always be obtained. An example is to select the forgetting factor so as to maintain the trace of the covariance matrix at a required value large enough to guarantee effective correction, but small enough to ensure that the estimator is not overly sensitive at all times. Examples of such an algorithm is that of Lozano-Leal (79) and the simple formula

$$\upsilon(t) = Tr(PP(t))/Tr(PP(t))^* \qquad 9.3$$

where * denotes required value. A key consideration for application of such an algorithm is the choice of the appropriate $Tr(PP(t))^*$, which may require trial and error to find the best value which will also depend on the values chosen for α. Some trial and error simulations were performed to investigate the possibility of using Equation

9.3 in the IMD1-STC with PC. An example is shown in Figures 9.38 (and Figure 9.39 for the corresponding parameters) for the case where the bounds of all the four **G** parameter are specified, $\alpha = 0.15$ and Tr(PP(t))* = 25. It is clear that the performance of the IMD1-STC is greatly improved by the PC method as the IAE is even less than the best servo performance of IMD1-STC reported in this chapter (Figure 9.21 and Table 9.3) by about 7%.

9.5.2 Discussions and Conclusions

The cases discussed in the previous section show that the SPC correction method is a simple and practical way of preventing the self tuning controller parameters from attaining bad values and so cause poor control. The results of the cases studied to assess the benefit of the SPC method confirmed that the correction of only two of the four key parameters which determine the controller gains was sufficent to restore good stable performance of the multiple loop self tuning controllers. The PC method could not give similar improvements, except when the covariance matrix was maintained at a large enough level, and, in a case reported above, the performance was even poorer when the parameter correction was not introduced.

The SPC method may also be applied in other areas. For example, in applications were the number of parameters that need to be estimated is large, conventional control may require a very long time to intially tune in the parameters of the adaptive controller. A method, such as the SPC, can be used for this process, as it has the potential for reducing the time required for the initial tuning.

A possible drawback of the SPC approach, and likewise the PC approach, is that problems will arise if the process is very non-linear so that the correct ranges of the parameters may change significantly with operating conditions.

9.6 Chapter Conclusion

Having assessed the viability of the self tuning algorithms on the column simulator, the next step is to apply the algorithms on a real system. These applications were carried out on the distillation column.

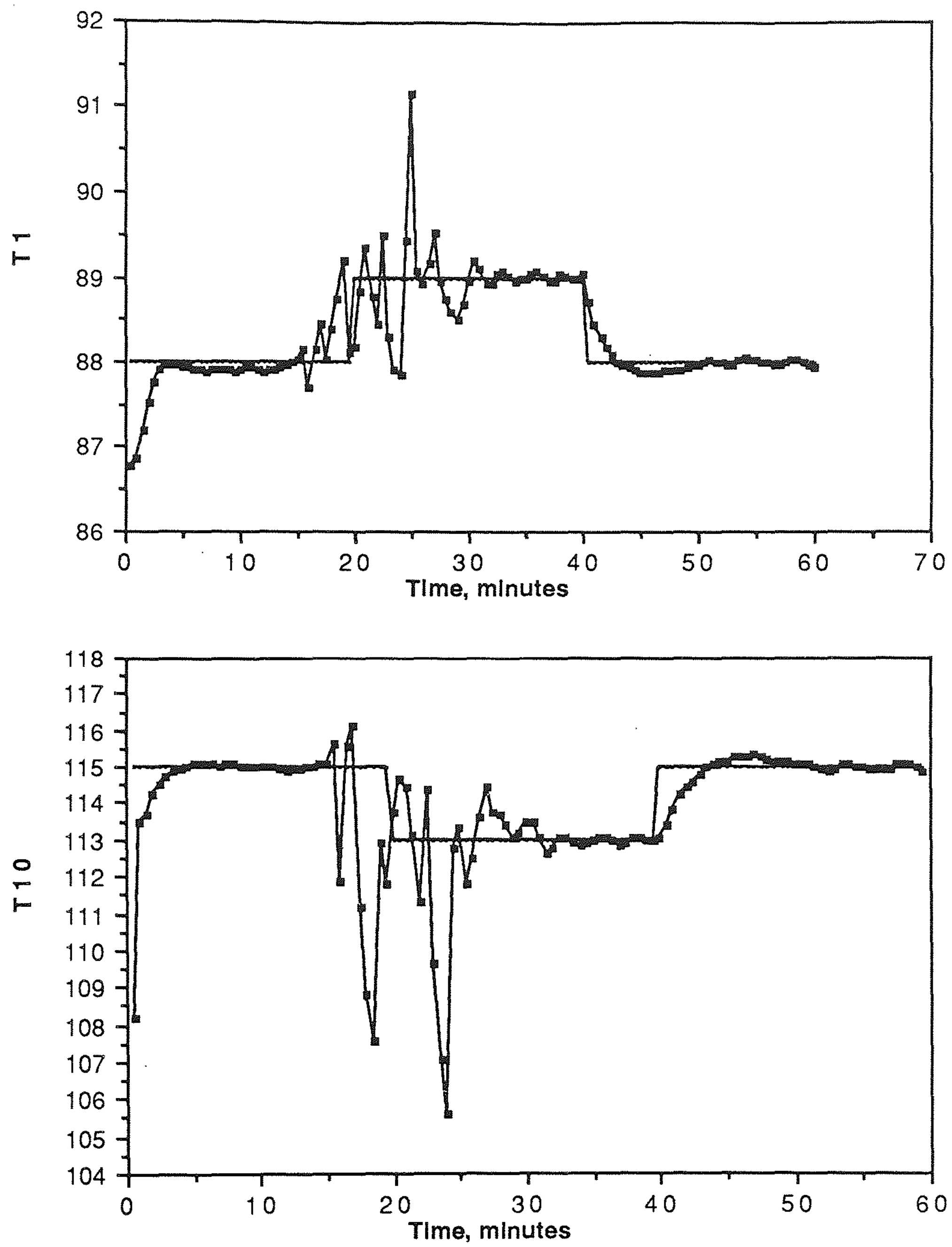


Figure 9. 28a The performance of IMD1-STC using a large initial covariance matrix PP(0) = 10I.

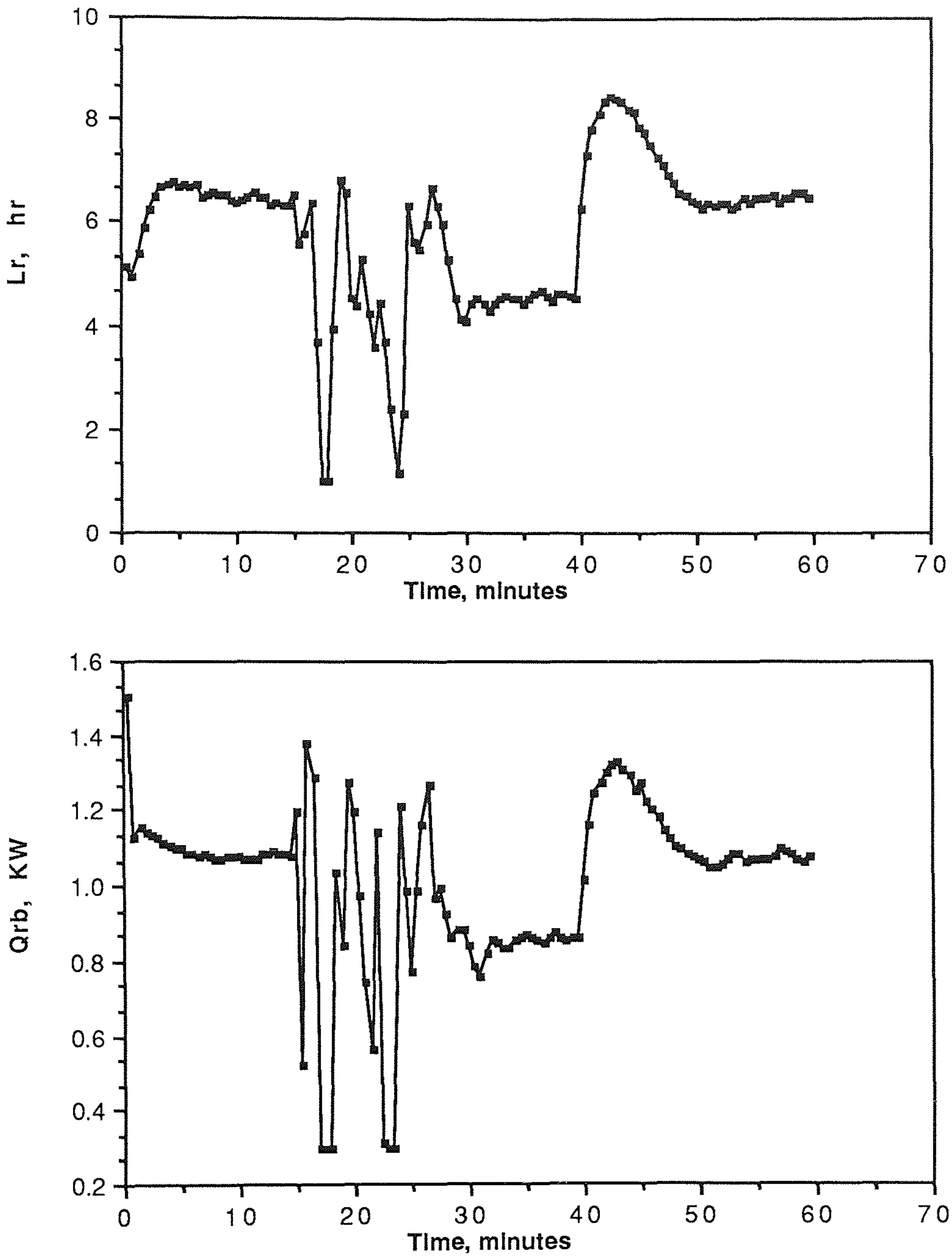


Figure 9.28b Control actions corresponding to Figure 9.28a

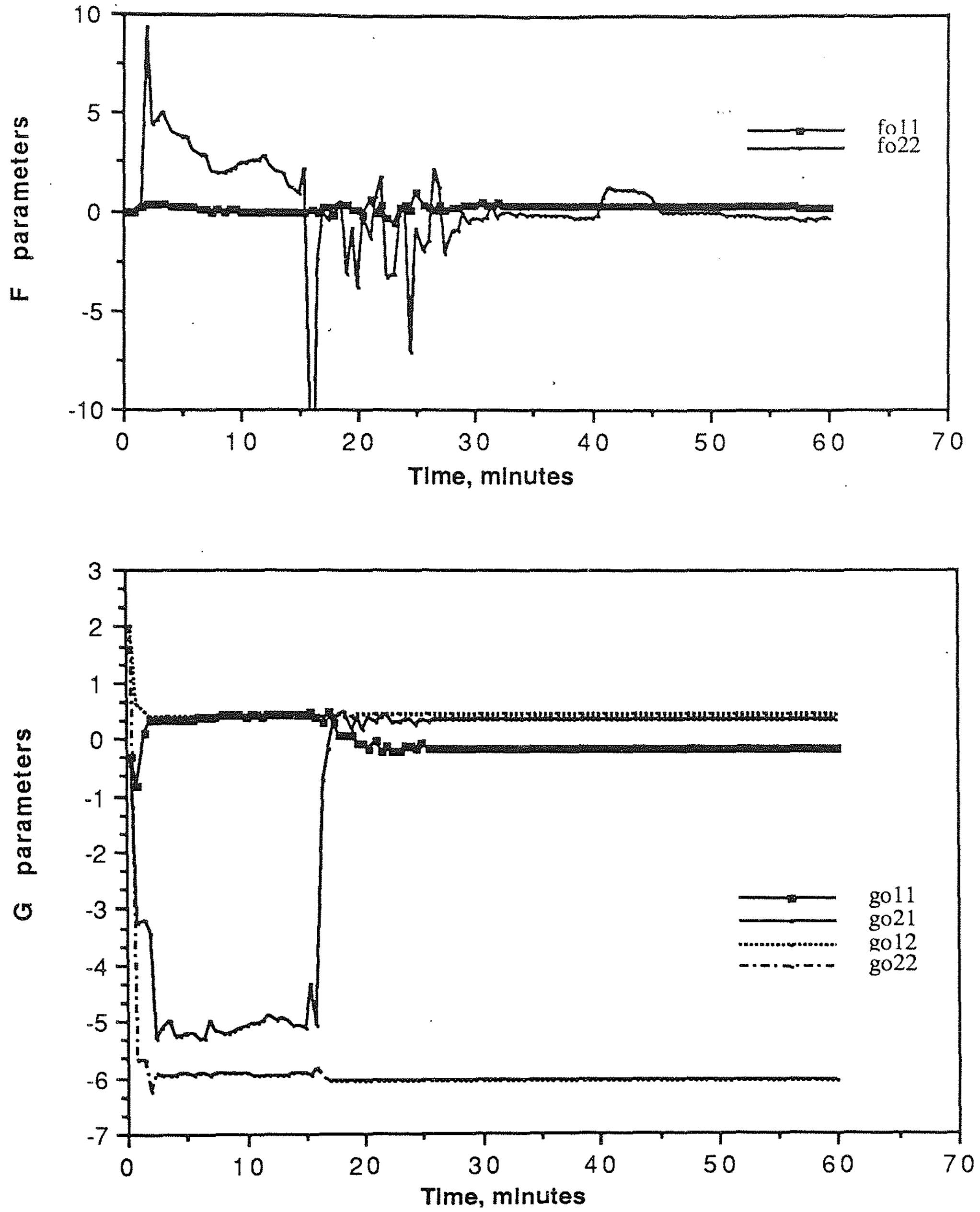


Figure 9.29 Parameter estimates corresponding to Figure 9.28

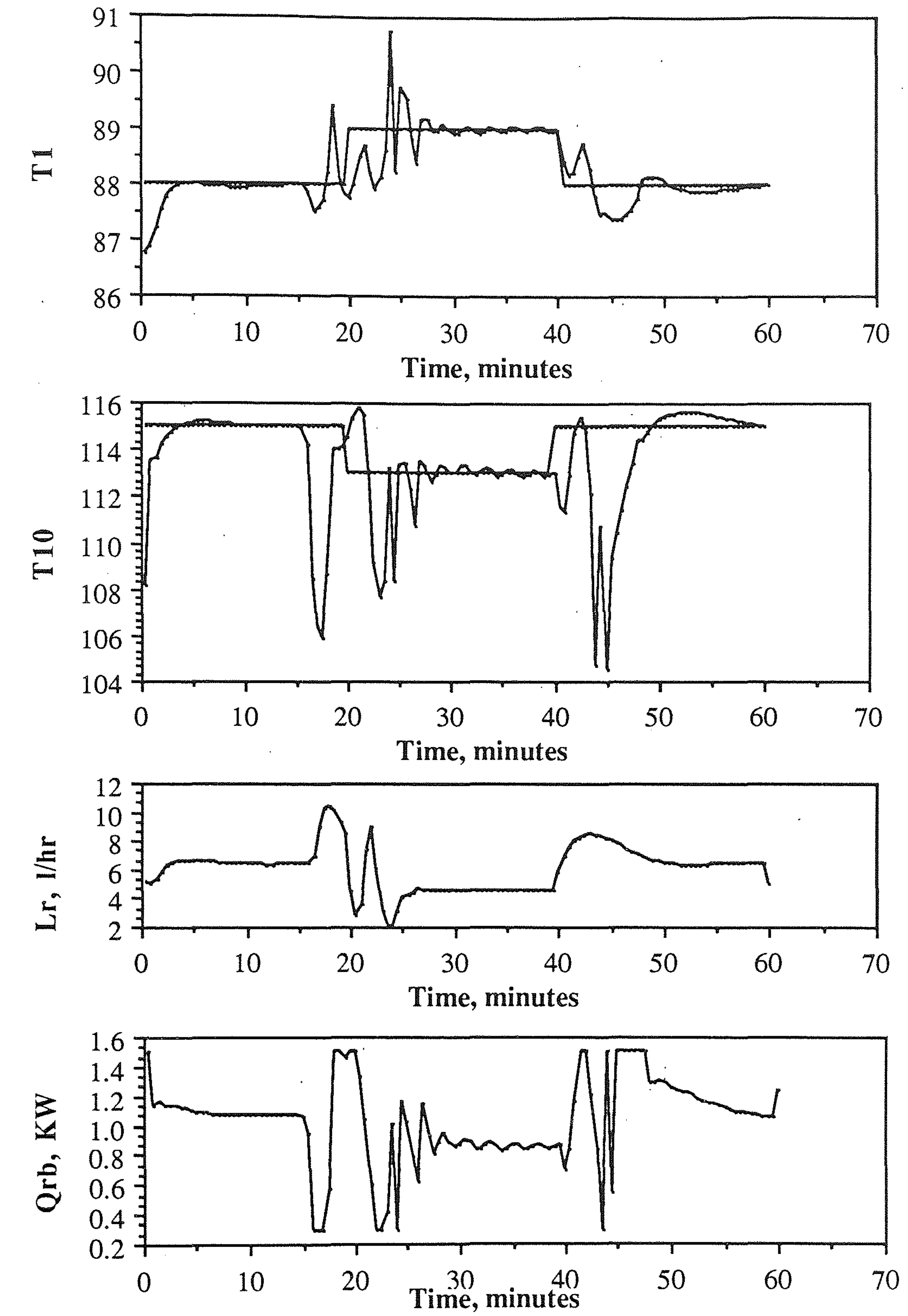


Figure 9.30 Performance of IMD1-STC combined with PC algorithm: The g_0^{11} and g_0^{22} parameters are specified and $\alpha = 0.15$ ..

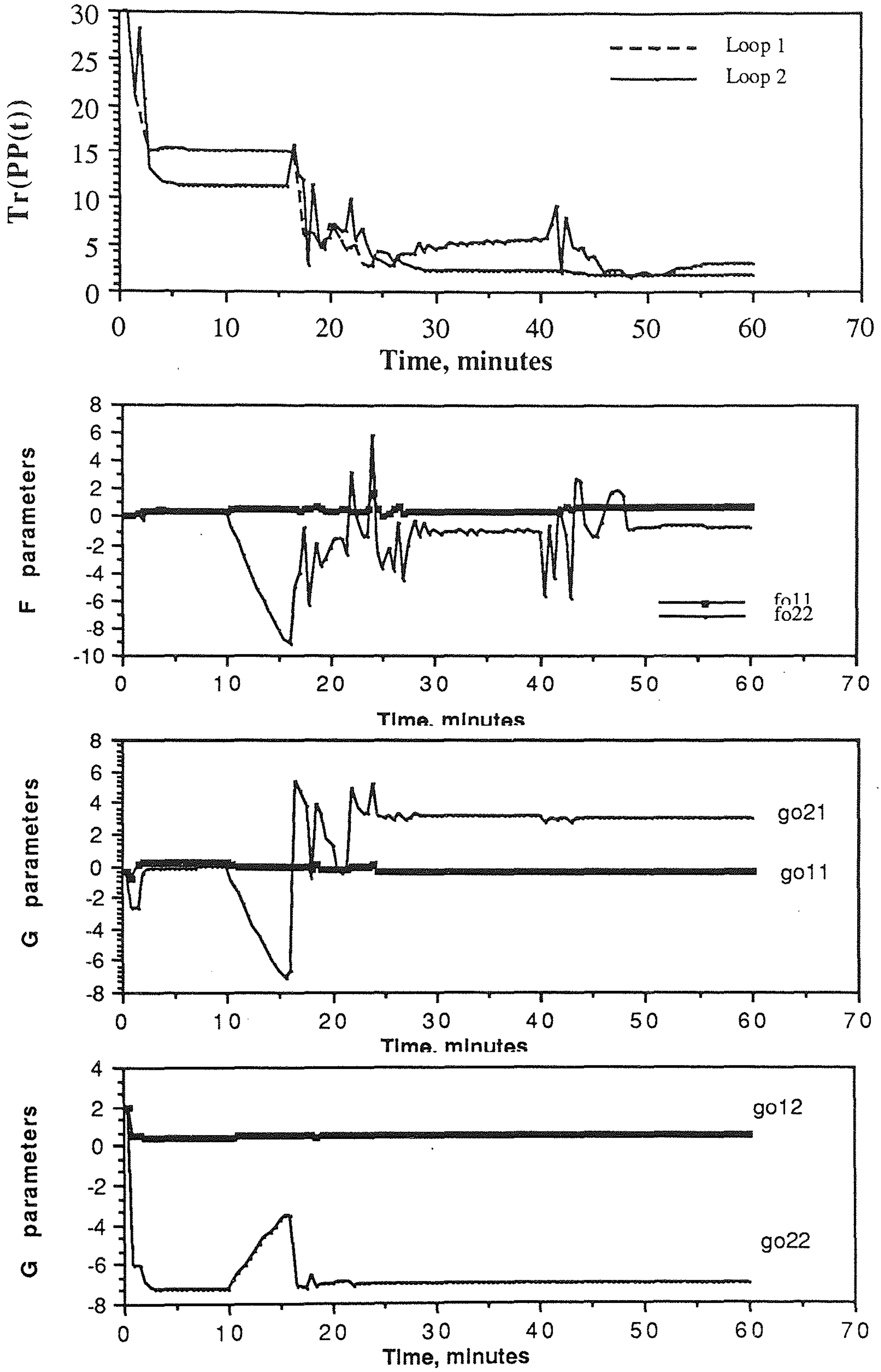


Figure 9.31 Behaviour of the traces of the covariance matrices and the parameter estimates: Graphs corresponds to Figure 9.30.

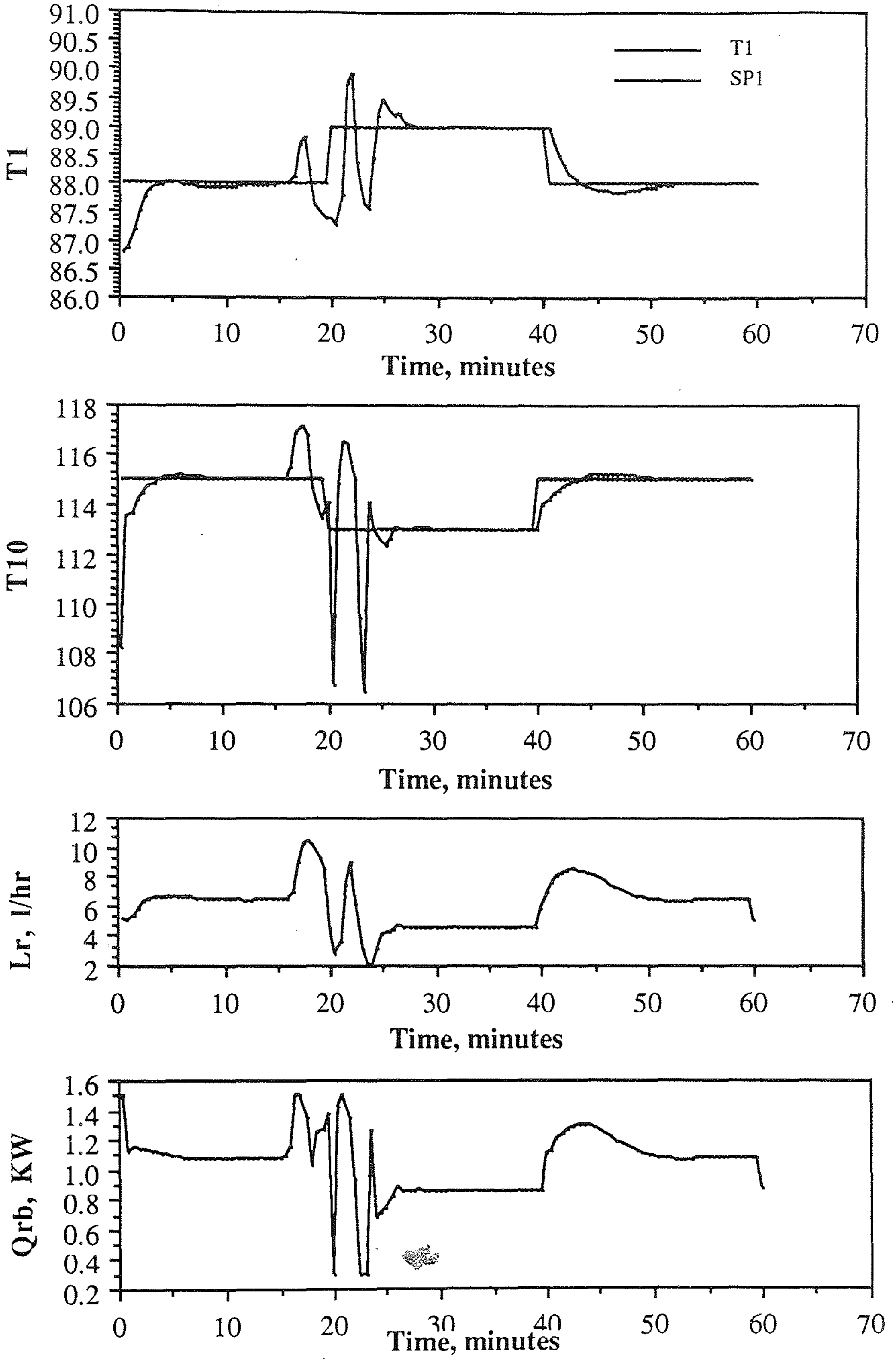


Figure 9.32 Performance of IMD1-STC combined with PC algorithm: The g_0^{11}, g_0^{12}, g_0^{21} and g_0^{22} parameters are specified and $\alpha = 0.2$

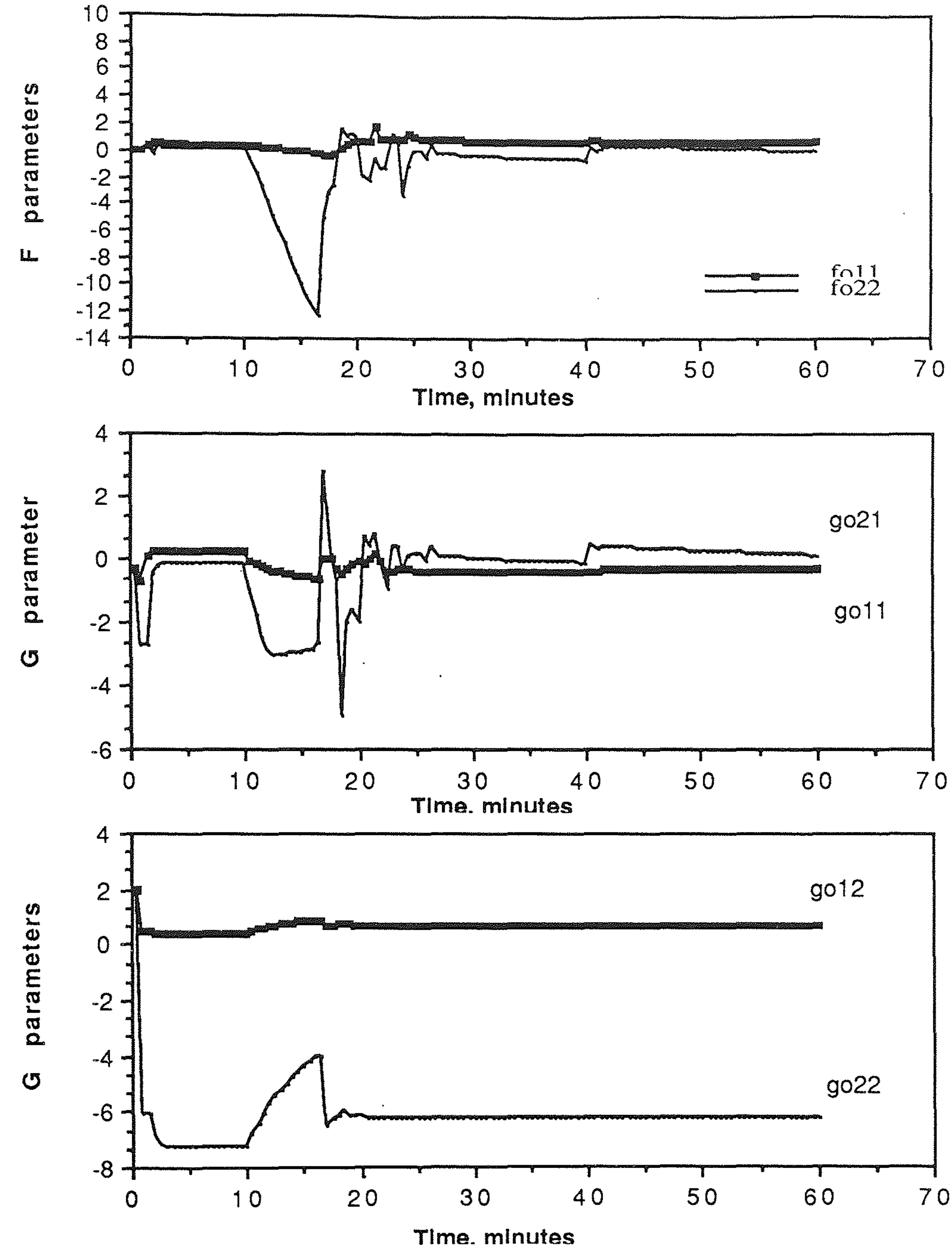


Figure 9.33 Behaviour of the parameter estimates: Graphs corresponds to Figure 9.32.

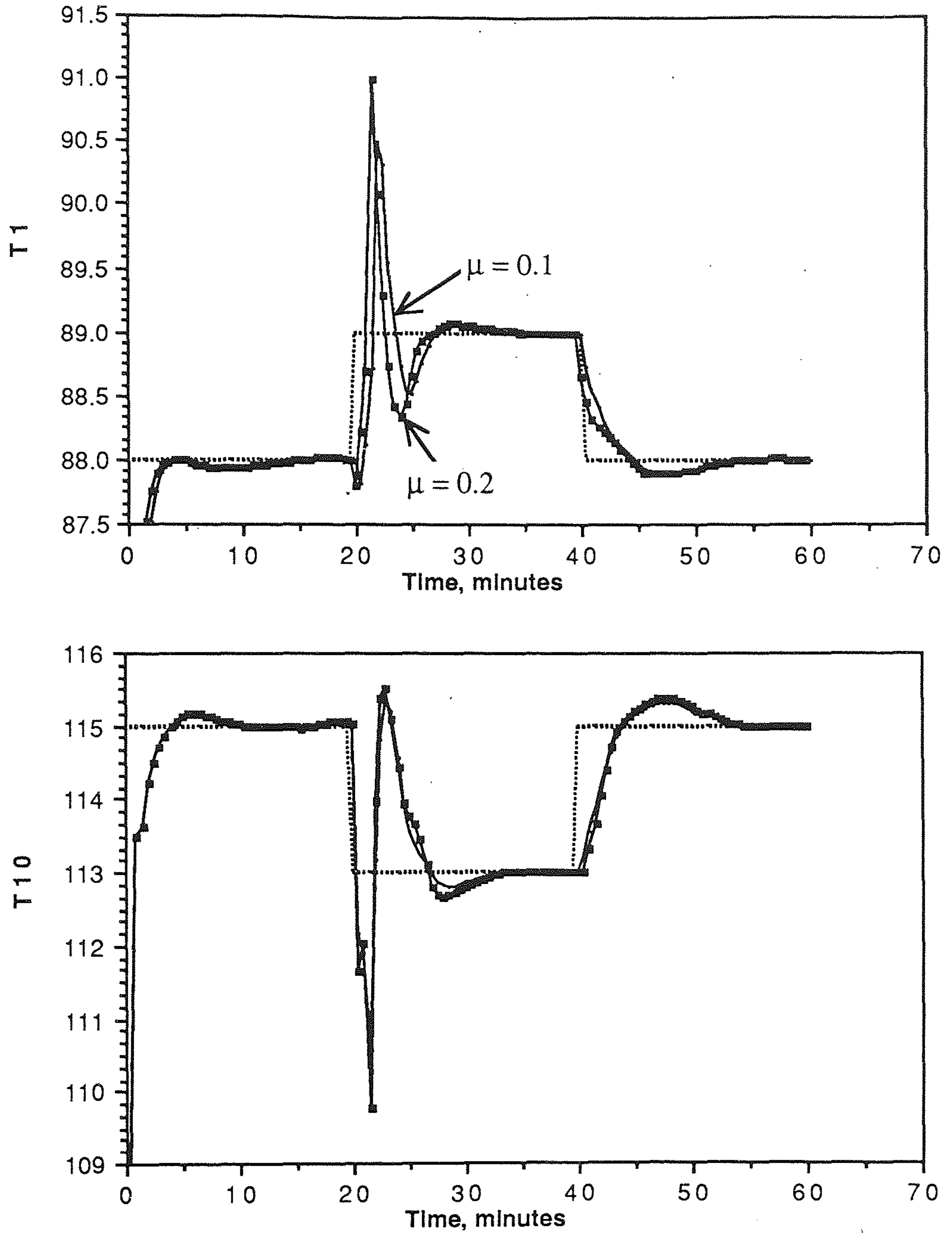


Figure 9.34a Effect of μ on the performance of IMD1-STC combined with SPC

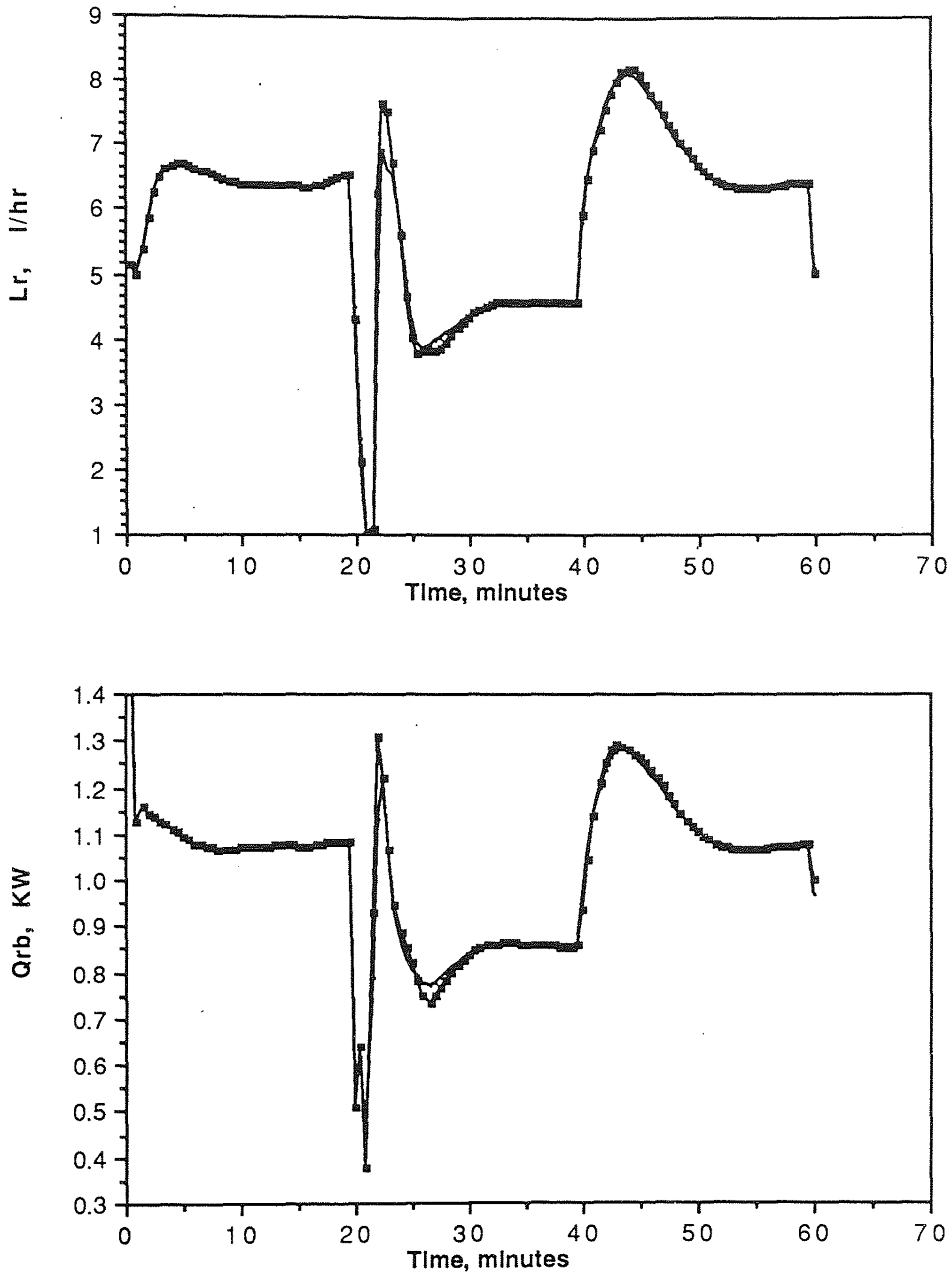


Figure 9.34b Control actions corresponding to Figure 9.34a

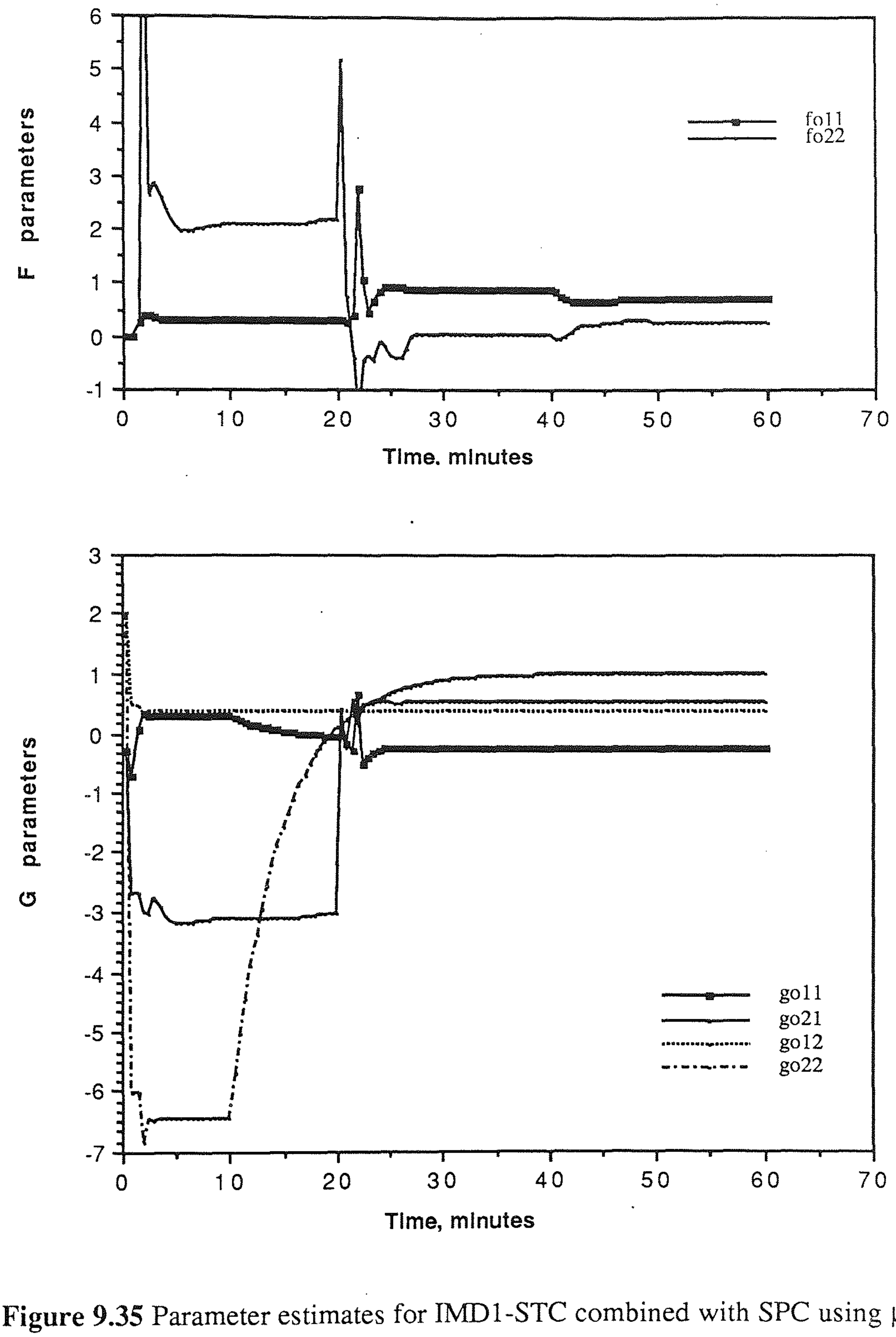
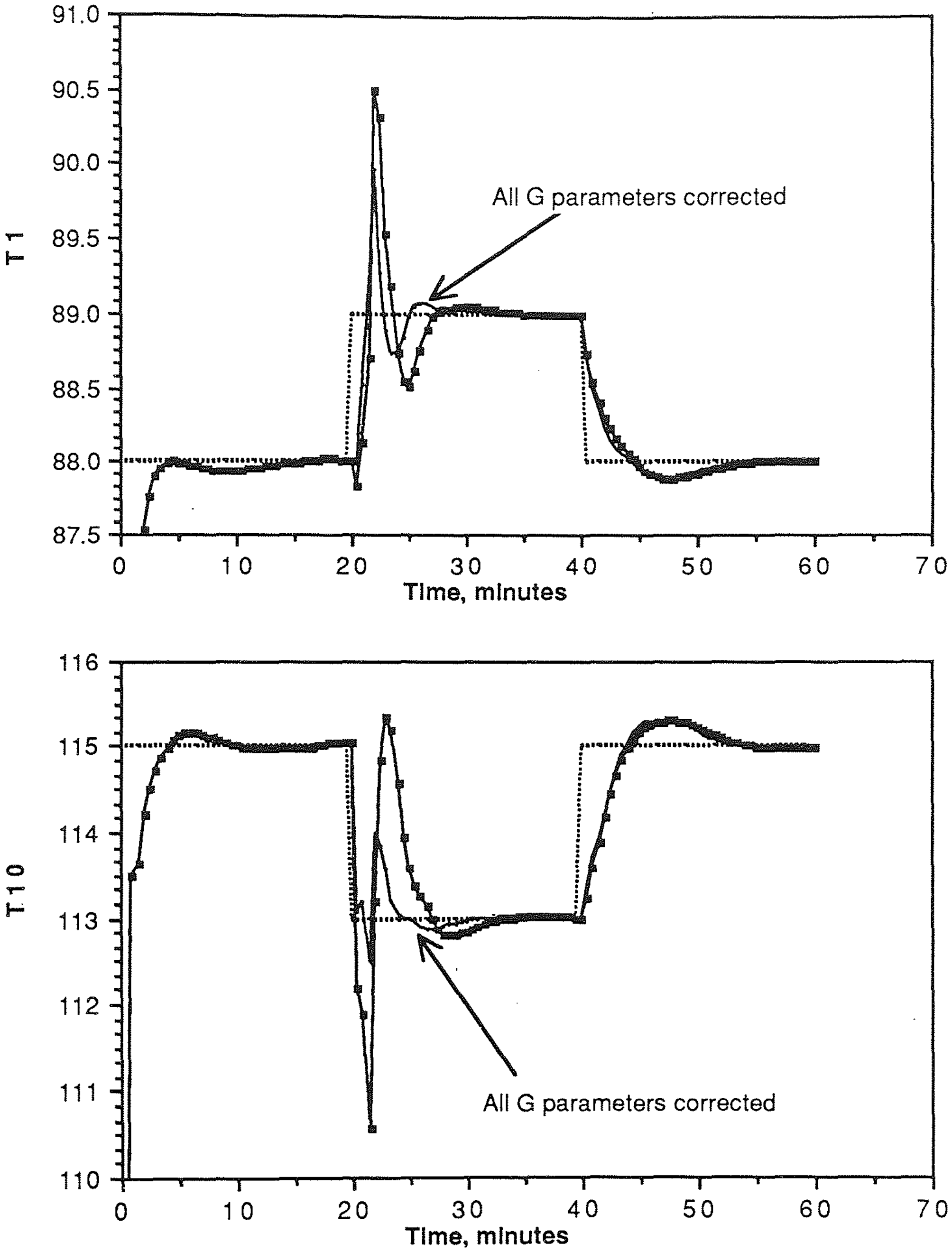


Figure 9.35 Parameter estimates for IMD1-STC combined with SPC using $\mu = 0.15$



thick lines - the g_0^{11} and g_0^{22} corrected
thin lines - all the **G** parameters (g_0^{11}, g_0^{21}, g_0^{12} and g_0^{22}) corrected

Figure 9.36a The performance of IMD1-STC combined with SPC for $\mu = 0.2$.

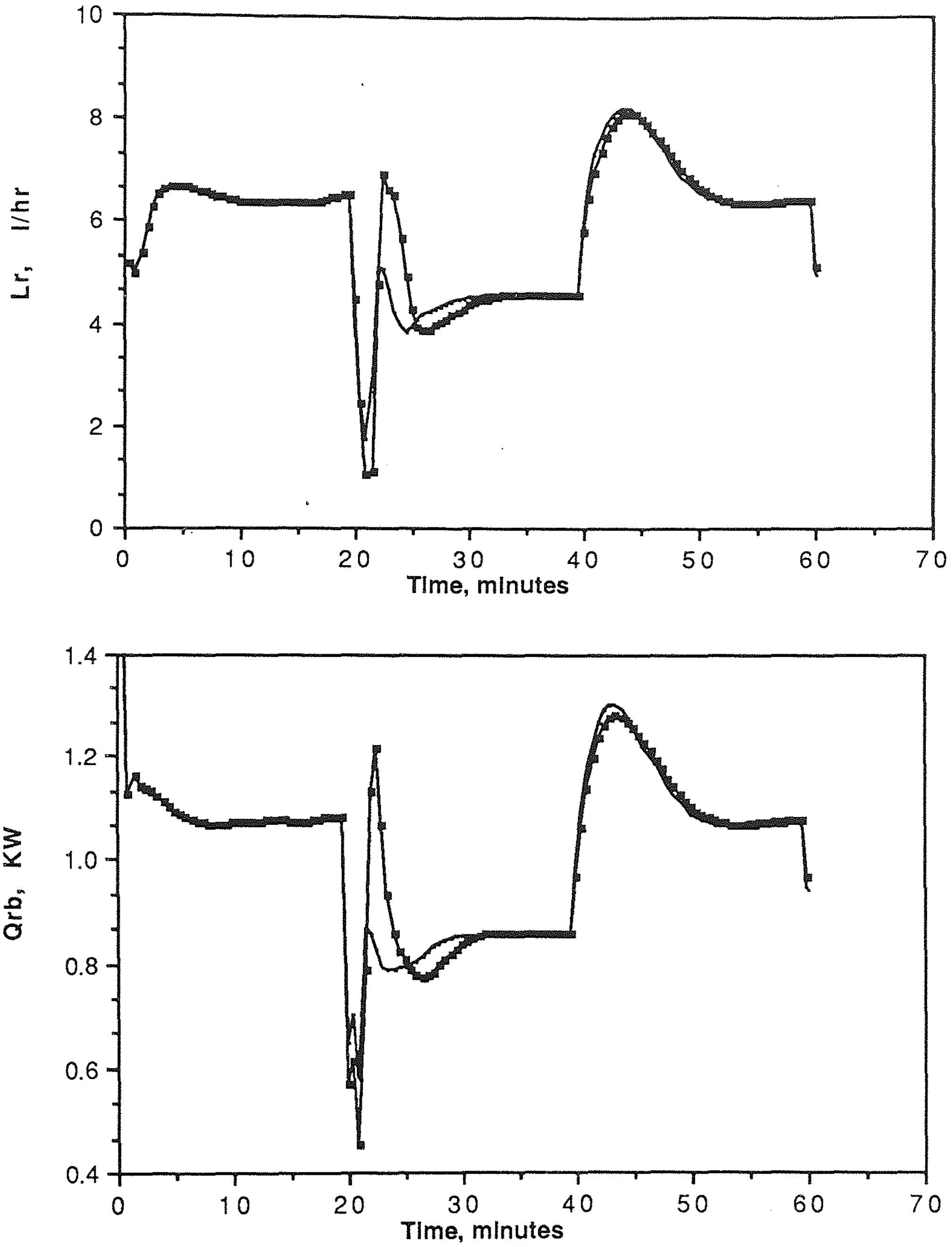
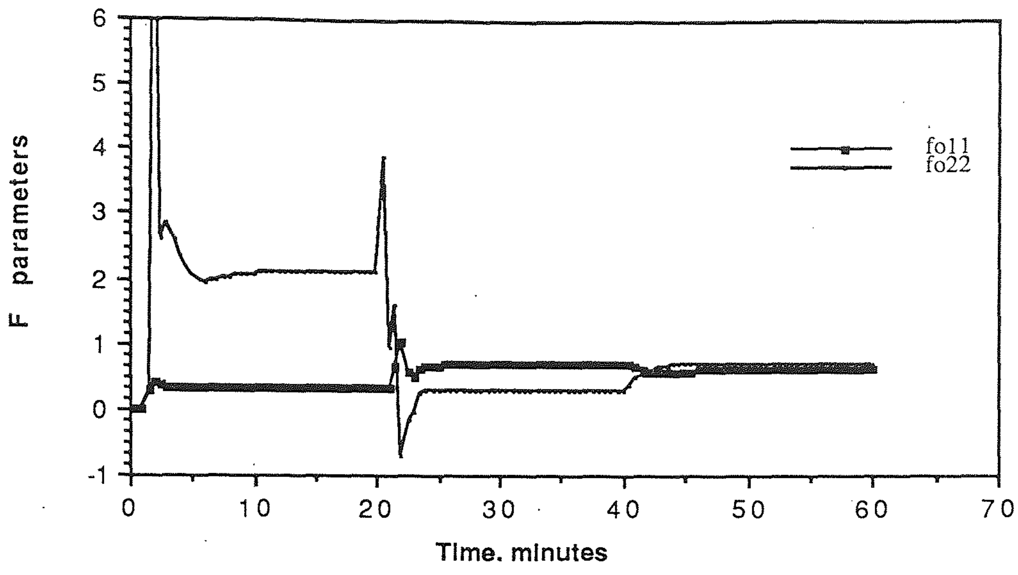


Figure 9.36b Control actions corresponding to Figure 9.36a



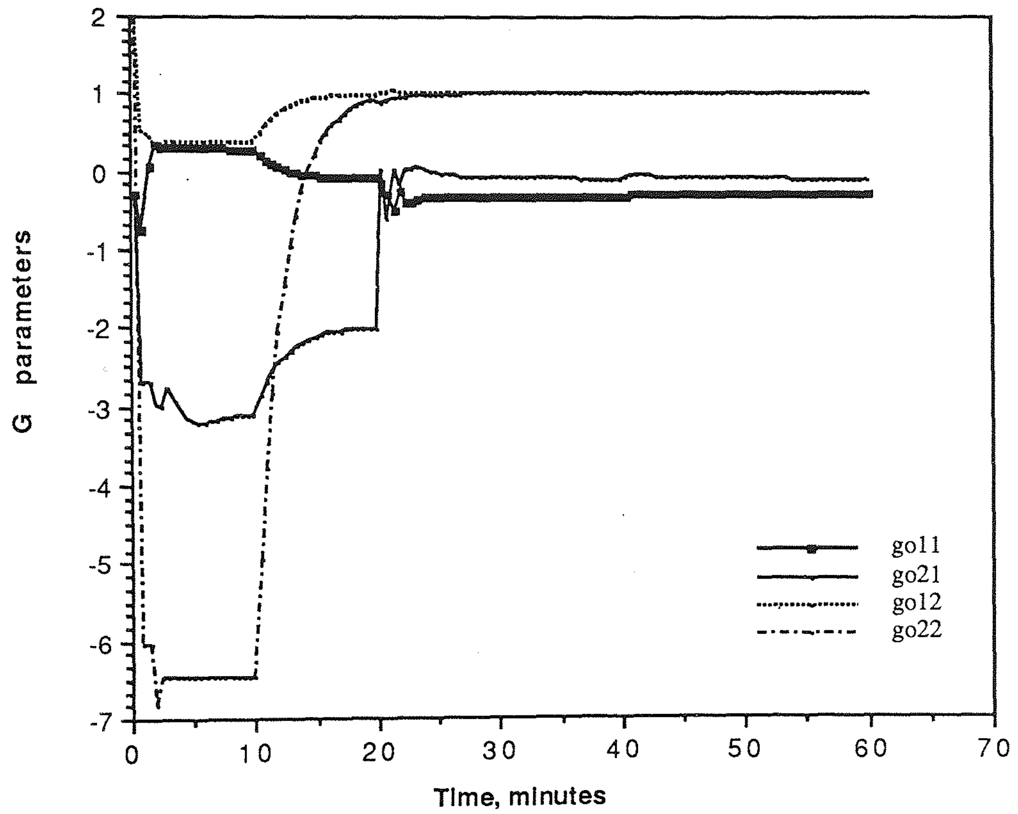


Figure 9.37 Estimator parameters for the case with the bounds of all the $\mathbb{G}$ parameters specified: IMD1-STC combined with SPC ($\mu = 0.2$)

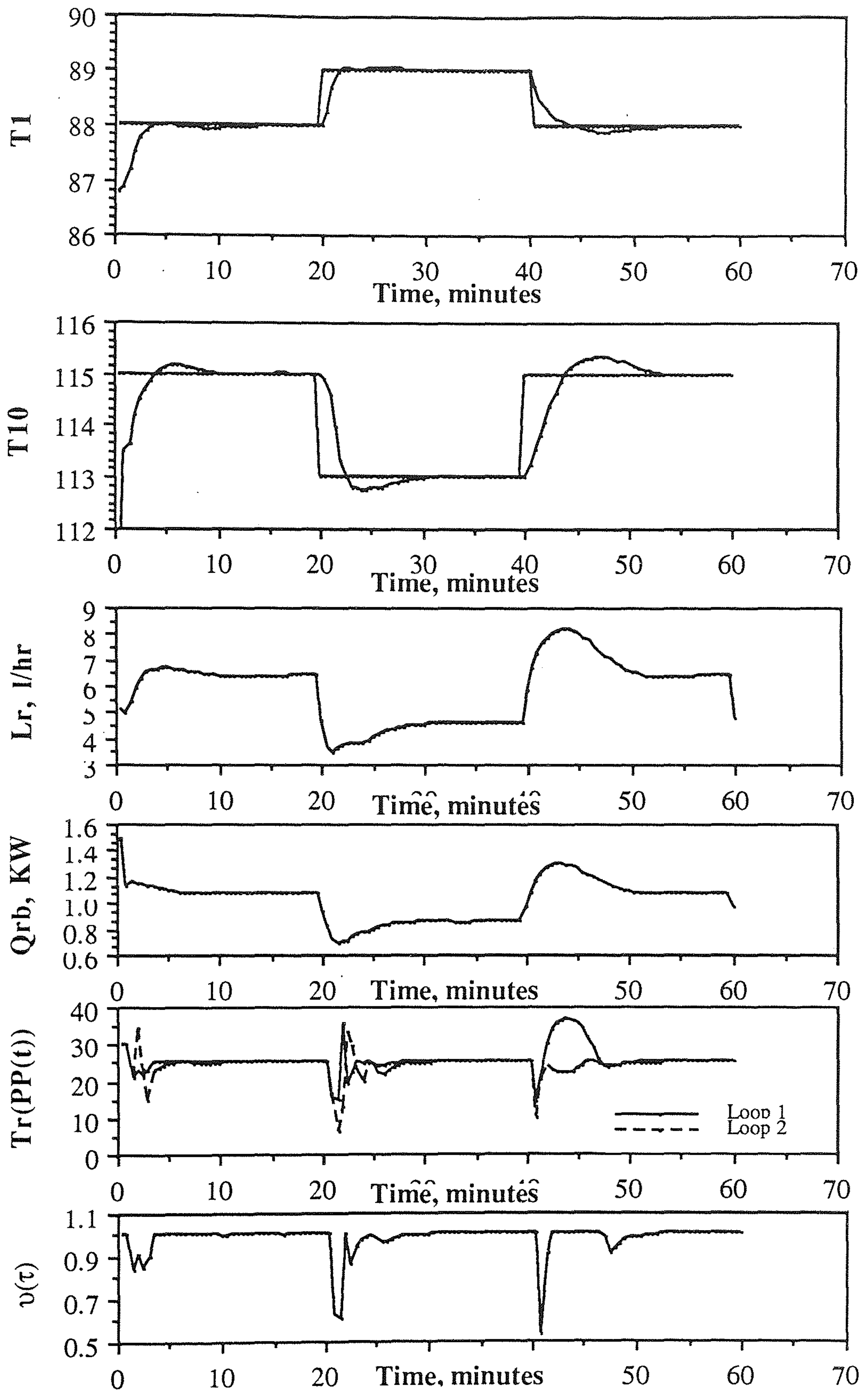
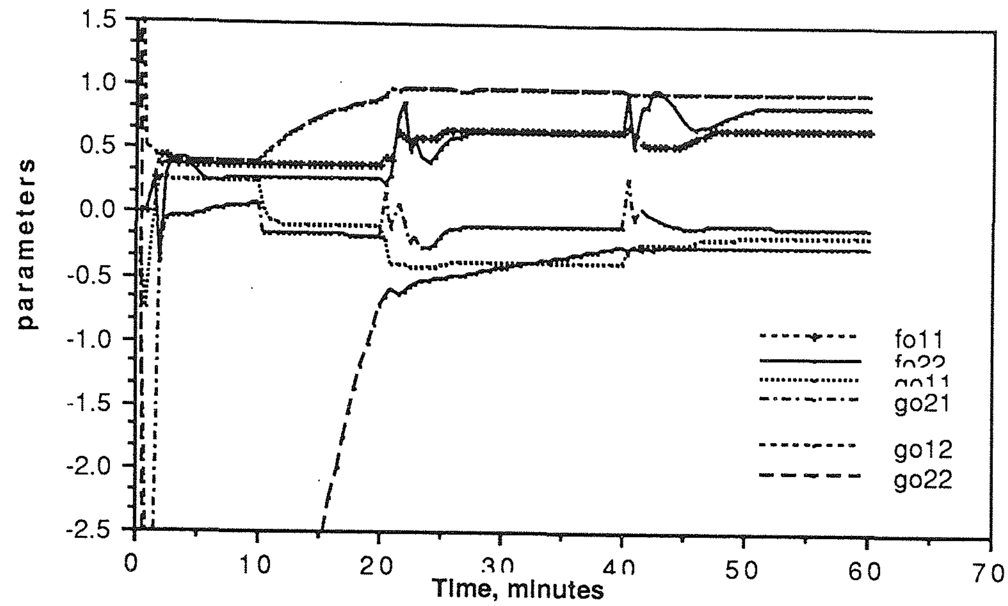


Figure 9.38 Performance of the PC method, $\alpha = 0.15$, with covariance matrix maintained constant at a large value by selecting the forgetting factor according to Equation 9.3.



The bounds of g_0^{11}, g_0^{12}, g_0^{21} and g_0^{22} specified

Figure 9.39 Parameter estimates corresponding to Figures 9.38

CHAPTER TEN

Microcomputer control of the pilot scale distillation column

10.1 Introduction

This chapter describes the computer control of the pilot plant distillation column which has been described in Chapter 4 and modelled in Chapter 5. Both single loop top tray temperature control and the simultaneous control of the top tray and bottom tray temperatures were performed. The tray temperatures were the controlled variables as composition analysers were not installed on the column, as was mentioned in Chapter 4.

Of the four controller design methods that were considered for application on the experimental column at the outset of the research work, the PI and the self tuning approaches were found to be applicable on the experimental distillation column. The decoupling and disturbance rejection control approach was not applied on the experimental column because the controller failed to provide satisfactory results when it was applied on the column simulator. This has been discussed in Chapter 6. The Estimator Aided Feedforward control approach, which used a Kalman filter to produce estimates of the tray compositions from process measurements, was also not applied on the real column. The reason for this is that the overall performance of the Kalman Filter was not satisfactory as it used a linearised state variable model of the column simulator as the filter model. Off-line studies using real process data, showed that the filter could not eliminate biases in the composition estimates and it produced unstable estimates of the tray compositions which did not have their corresponding tray temperatures measured. This has been discussed in Chapter 7. It was the PI and self tuning controller designs that provided satisfactory results on the column simulator as discussed in Chapter 9. They were therefore selected for application on the experimental control for both single loop control of the top tray temperature and simultaneous control of the top and bottom tray temperatures.

The results of the simulation studies described in the Chapter 9 demonstrated that the SISO and MIMO self tuning controllers perform better than corresponding PI control. Simulations using the MIMO self tuning controllers were also used to demonstrate the poor control that could result if the parameter estimates of the self tuning controller attained bad values. The parameter correction methods, SPC and PC, were designed to remedy this problem and their potential of improving controller performance were demonstrated. The control experiments on the real column were designed to demonstrate the capabilities of the controllers and examine if similar conclusions as those reached from the simulations can be made.

10.1.2 Implementing the controllers on the experimental column.

For the purpose of controlling the experimental column, the System 96 microcomputer was interfaced with the distillation column using the Monolog. This unit contains the A/D and D/A converters, and signal conditioners for appropriate data conversions. The equipment has been described in Chapter 4.

The programs that implement real-time control are written in Basic09 and run on the System 96. A subprogram called **Get-data** retrieves process measurements from the Monolog and **Drive-valve** sends the control actions through the Monolog to the final control elements which are the control valves and the firerod heater. The functions of these programs are explained more fully in Appendix A6.

The program that implements real-time PI control is called **PI-decouple**. It implements both single loop top tray temperature control and simultaneous control of the top and bottom tray temperatures with or without simplified steady state decoupling. The velocity form of the PI controller was used and is described in Appendix A2.2.1. Two real-time self tuning control program called **POSTC** and **KISTC** implements the positional and incremental forms of SV-STC, MD2-STC, MD3-STC and MD1-STC. The structure of the program that implement these algorithms have also been described in Section 8.4.

The controllers were operated in the presence of the operational difficulties encountered during operation of the experimental column which were discussed in

Chapter 4. Some of these difficulties, such as pipe blockages and uncertainties in the inputs into the column were considered as significant contributors to the error between the column simulator and the column. Another problem was the inconsistencies in the delivery of the distillate and bottoms products, partly due to the small flowrates required to operate the column, which made it necessary to use on-off control to control the reboiler liquid level. Furthermore, the tendency of the reflux flowmeter failing to provide flow measurements made it necessary, for control purposes, to approximate the reflux flow into the column by using the fractional opening of the reflux valve.

10.2 Single loop control of the top tray temperature

The control of the top tray temperature was performed using PI control, PSV-STC and ISV-STC. The control interval, ΔTc, of 0.5 minutes was used in all the experiments. The specifications given below were used in the for SV-STC:

$\theta(0) = (f_0, g_0, d) = (0.5, -0.33, 0)$ for PSV-STC

$\theta(0) = (f_0, g_0) = (0.5, -0.33)$ for ISV-STC

$\mathbf{Q} = -0.5\Delta_1$, $\mathbf{P}(z^{-1}) = (1 - 0.632z^{-1})/0.368$, $\mathbf{R}(z^{-1}) = 1$

$PP(0) = 0.5I$ representing confidence in $\theta(0)$

VFF3 algorithm (Equation 8.49) for variable forgetting

error limit $ep_{max} = 1°C$ for the parameter estimator.

Upper and lower limits of the reflux valve opening were set at 12% and 60%, respectively. A PI controller with the settings Kc = -2.0 (l/hr)/°C and $\tau_i = 3.0$ minutes was used to initially tune in the self tuning controller parameters.

10.2.1 Discussion of the results

Figure 10.1 shows the performance of a PI controller with Kc = -2.5 (l/hr)/°C and $\tau_i = 3.0$. Satisfactory control of the top tray temperature was maintained over the duration of the experiment. The overshoot after the setpoint increases at t = 15 minutes and t = 45 minutes were relatively small compared to the overshoot of about

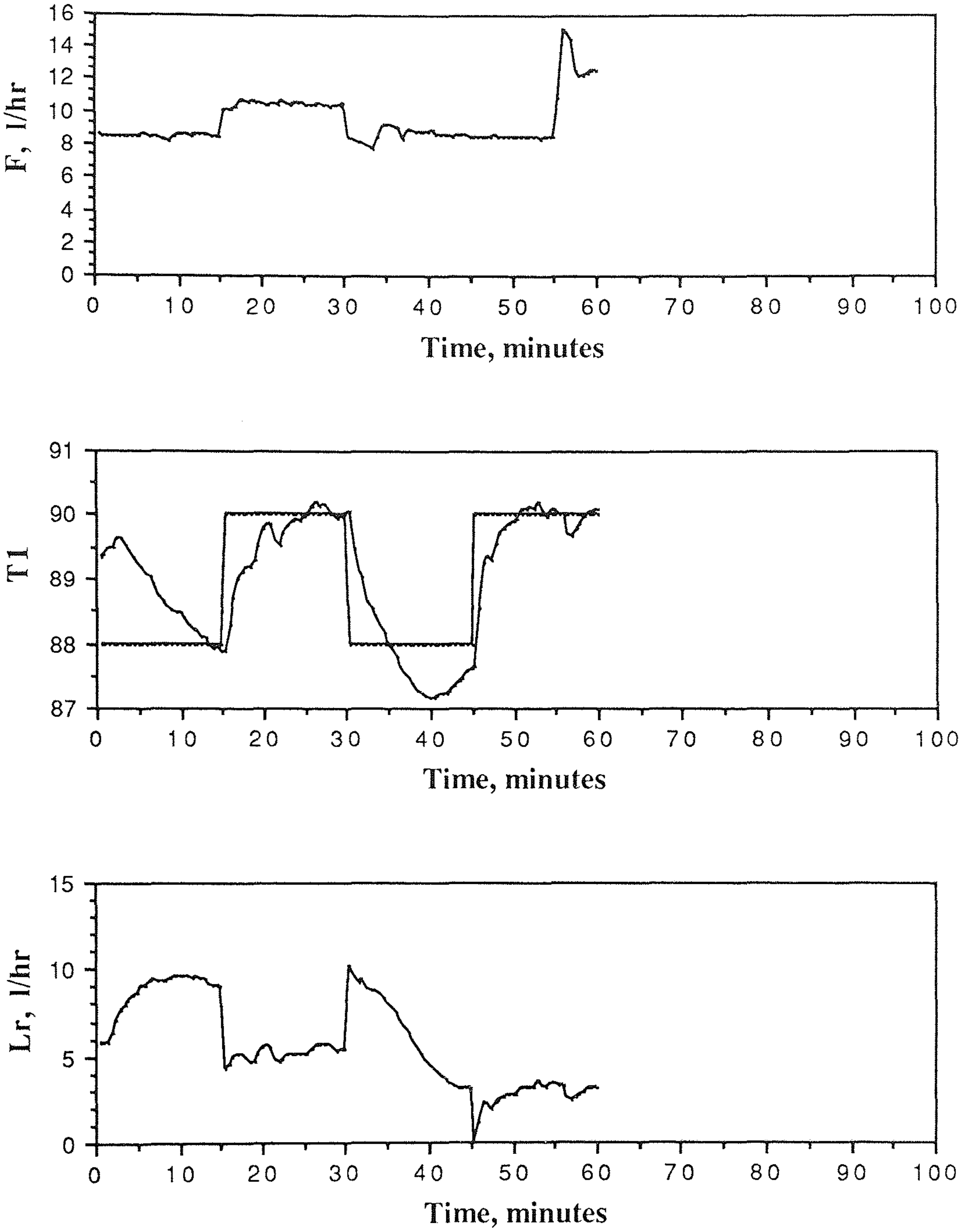
1°C after the setpoint decrease at t = 30 minutes. The PI controller was slow to return the top tray temperature back to this new setpoint.

The PSV-STC also provided satisfactory servo and regulatory control of the top tray temperature as is shown in Figure 10.2. Unlike PI, the response of the output under PSV-STC was oscillatory and there was no overshoot after the setpoint decrease at t = 30 minutes. The ISV-STC gave a much faster closed loop response than both the PSV-STC and the PI (Figure 10.3) particularly after the setpoint changes. The performance of ISV-STC is considered more satisfactory than that of the PSV-STC since the former did not give the oscillatory behaviour of the output observed in the performance of the latter.

It was also observed that the parameter estimates for the PSV-STC remained virtually constant after initial tuning while those for the ISV-STC changed significantly after setpoint and load changes (Figure 10.4). This observation is consistent with the observations made from the simulations on the column simulator (Figures 9.9 and 9.10).

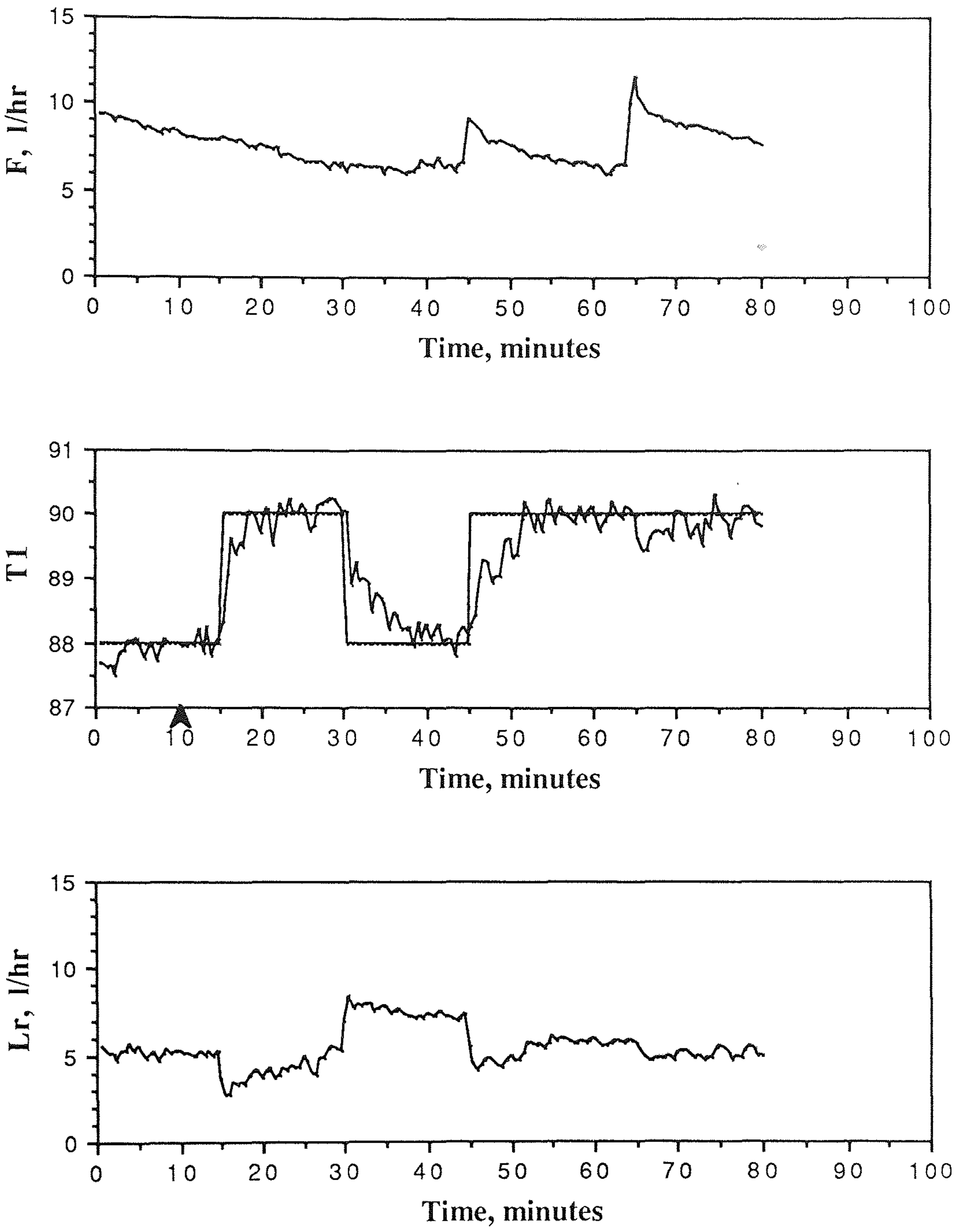
10.2.2 Summary

The experimental results reported in this section have demonstrated that the PI, PSV-STC and ISV-STC were able to provide satisfactory control of the top tray temperature of the distillation column when the column was subjected to unmeasured feedflow disturbances and setpoint changes. From these results it could be concluded that the ISV-STC is capable of providing tighter control than both PI and PSV-STC because it gave faster closed loop responses than the latter controllers.



Kc = -2.5 (l/hr)/°C and τ_i = 3 minutes
Feed composition 42/58 w/w % trichloroethylene/tetrachloroethylene

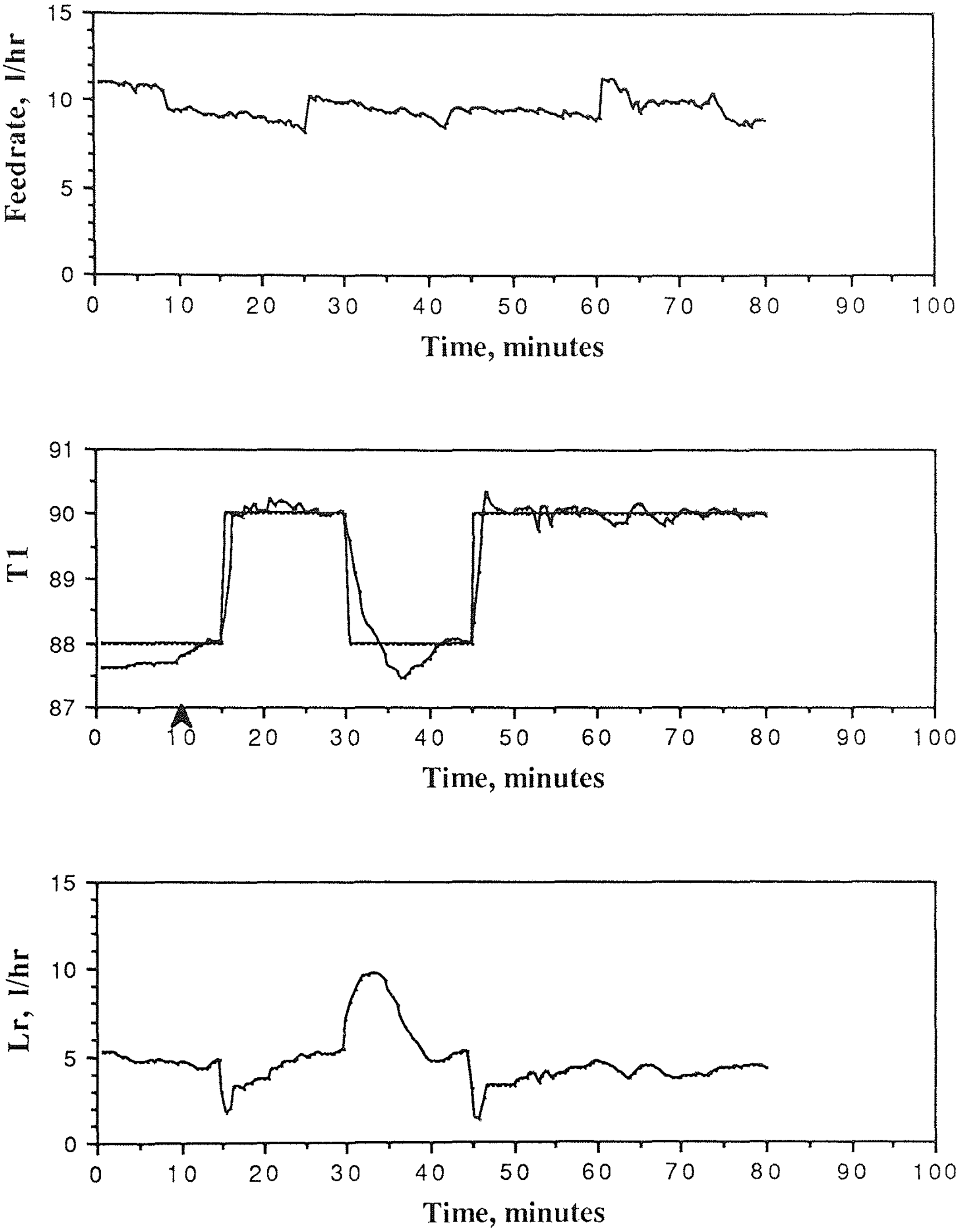
Figure 10.1. Top tray temperature control using Proportional + Integral controller



Feed composition 47/53 w/w % trichloroethylene/tetrachloroethylene

$\mathbf{Q} = -0.5\Delta_1$, $\mathbf{P}(z^{-1}) = (1 - 0.632z^{-1})/0.368$, $\mathbf{R}(z^{-1}) = 1$

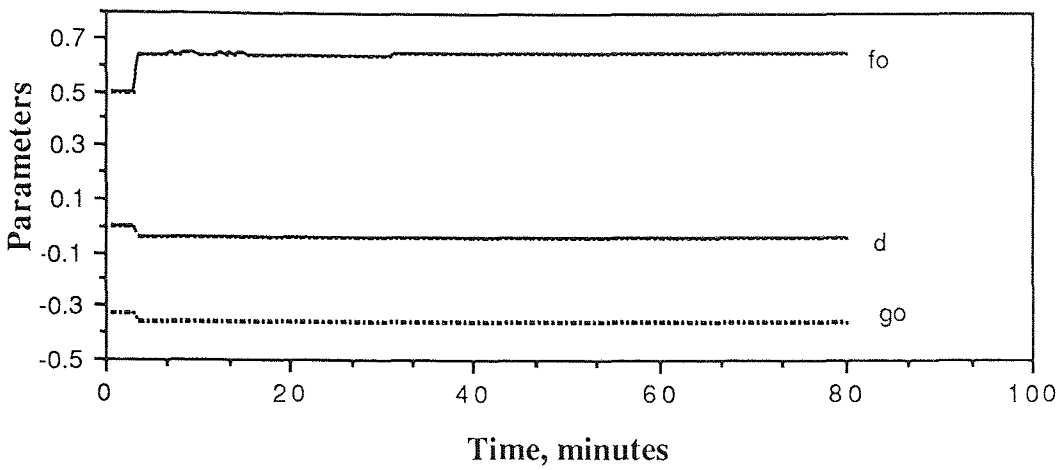
Figure 10.2. Top tray temperature control using PSV-STC



Feed composition 50/50 w/w % trichloroethylene/tetrachloroethylene

$\mathbf{Q} = -0.5\Delta_1$, $\mathbf{P}(z^{-1}) = (1 - 0.632z^{-1})/0.368$, $\mathbf{R}(z^{-1}) = 1$

Figure 10.3. Top tray temperature control using ISV-STC



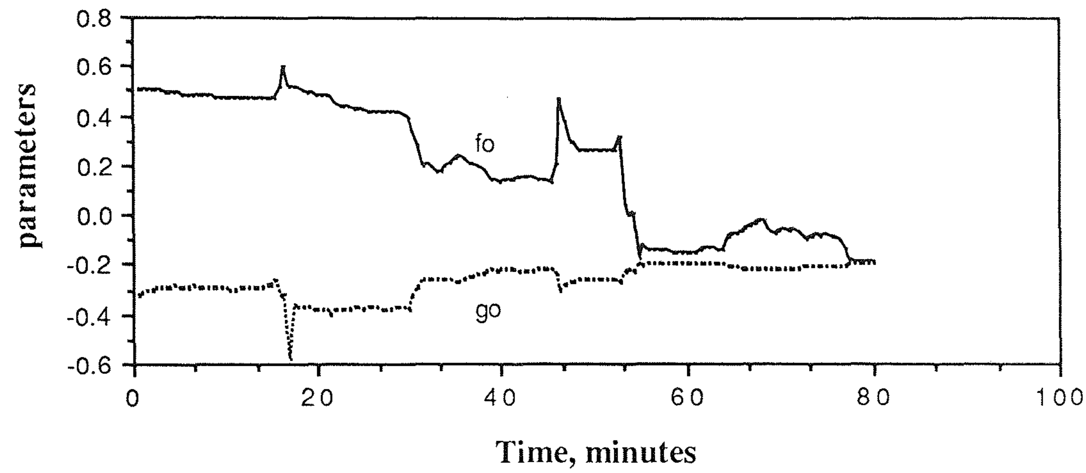


Figure 10.4 Self tuning controller parameters for PSV-STC (Top) and ISV-STC (Bottom)

10.3 Simultaneous control of top and bottom tray temperatures

In these experiments the capabilities of the multiple loop PI controllers, IMD3-STC and IMD1-STC were examined in the simultaneous control of the top tray temperature, T_1, and the bottom tray temperature, T_{10}, of the pilot plant distillation column. The configuration used on the column simulator, where Lr controls T_1 and Qrb controls T_{10} loop, was also used on the experimental column. The following settings were used in the self tuning controllers: $\theta_1(0) = \theta_2(0) = (0.5, -0.33, 1.8)$ for IMD1-STC and IMD2-STC, $\theta_1(0) = (0.5, -0.33)$, $\theta_2(0) = (0.5, 1.8)$ for IMD3-STC, $PP_1(0) = PP_1(0) = 0.5I$, $\mathbf{Q} = diag(-0.5,15)\Delta_1$, $\mathbf{P}(z^{-1}) = diag((1 - 0.632z^{-1})/0.368, (1 - 0.632z^{-1})/0.368)$, $\mathbf{R}(z^{-1}) = I$ and $\Delta Tc = 0.5$ minutes. The single loop PI controllers used to simultaneously control T_1 and T_{10} have the settings

$Kc = -2.0$ (l/hr)/°C and $\tau_i = 3.0$ for the Lr - T_1 loop

$Kc = 0.08$ KW/°C and $\tau_i = 1.2$ for the Qrb - T_{10} loop.

10.3.1 Discussion of the results

Figure 10.5 demonstrates the degrading effect of simplified steady state decoupling ($D_1 = 5$ and $D_2 = 0.1$) on the performance of the multiple loop PI control scheme when the system was subject to setpoint changes. After the setpoint changes at t = 20 minutes, the closed loop behaviour of both outputs were significantly worse than when the decouplers were not used. This shows that, with these decouplers, simplified decoupling was detrimental to the simultaneous control of the top and bottom tray temperatures using the multiple loop PI control system. The poor performance can be attributed to the errors in the decouplers which were those used on the column simulator as well.

Figures 10.6, 10.7 and 10.8 show, respectively, the performances of the multiple loop PI, IMD3-STC and IMD1-STC controllers in the presence of unmeasured feedflow disturbances. In the case of PI control, both outputs exhibited oscillatory responses and slowly aproached their setpoints after the feedflow disturbances at about t = 9 minutes. With IMD3-STC both outputs oscillated about their setpoints and this persisted for a relatively long period (about 30 minutes) before settling down.

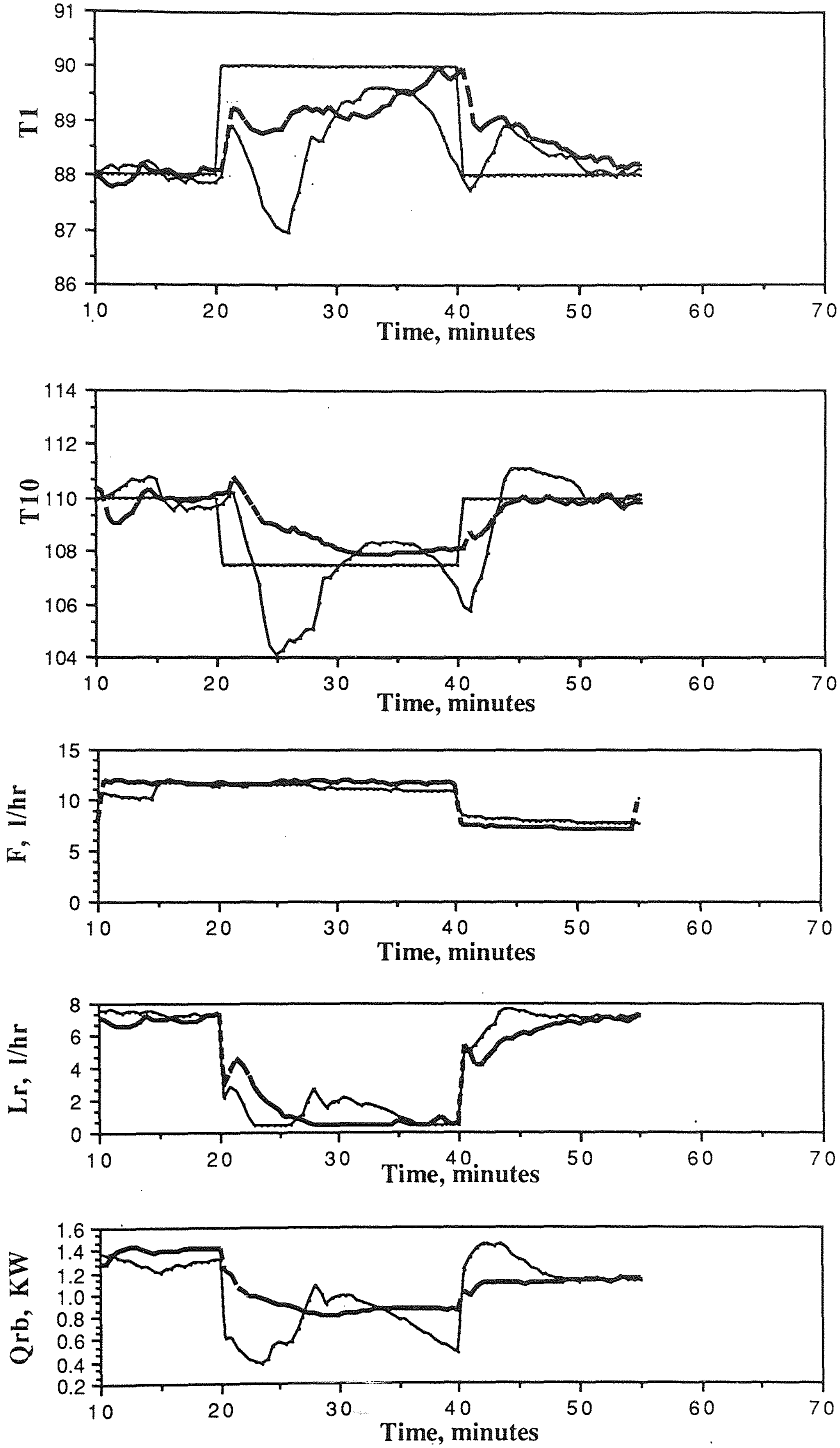
Although when under IMD1-STC the outputs also exhibited oscillatory responses around there setpoints, the oscillations diminished relatively quicker than with IMD3-STC. In this respect the IMD1-STC is clearly better than the IMD3-STC.

The oscillatory closed loop behaviour observed in the simultaneous control of the top tray and bottom tray temperatures can be mainly attributed to the interactions between their control loops as changes in the manipulated variable of one loop affects both outputs. The performance of IMD3-STC, which does not provide decoupling, demonstrate the severity of these interactions as the oscillatory behaviour persisted for a longer period compared with that of IMD1-STC which incorporates decoupling in its design.

The reason why decoupling was favourable in the self tuning control is because the adaptive nature of the control system produced a more accurate decoupling elements and therefore more accurate decoupling of the control loops. This is in contrast with multiple loop PI control system with steady state decoupling, in which the decouplers are in error and no facility available to automatically improve these values during control.

10.3.2 Summary

The experimental results have shown that the three control schemes all provided satisfactory control of the top tray and bottom tray temperatures. The IMD1-STC is capable of producing better control than the multiple loop PI system as the experimental results show that the closed loop response of the latter is relatively slow. The IMD1-STC is also better than the IMD3-STC as the former gave a less oscillatory closed loop behaviour. Therefore, the IMD1-STC is recommended for the dual composition control of the distillation column.



dashed lines - multiple loop PI, thin lines - multiple loop PI + decoupling

Figure 10.5. Effect of steady state decoupling on the performance of the multiple loop PI controllers

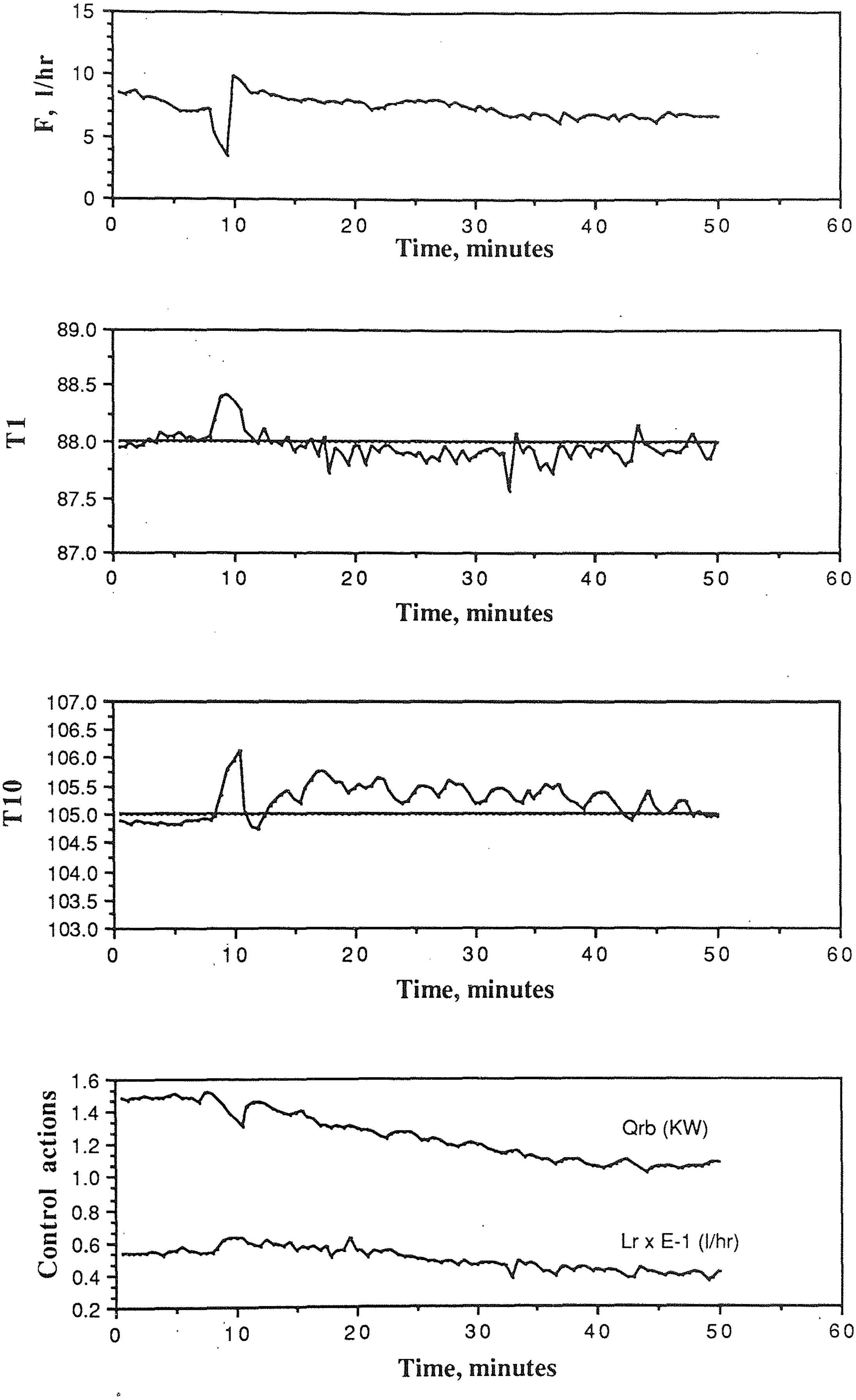


Figure 10.6 Multiple loop PI control in presence of feed flow disturbances

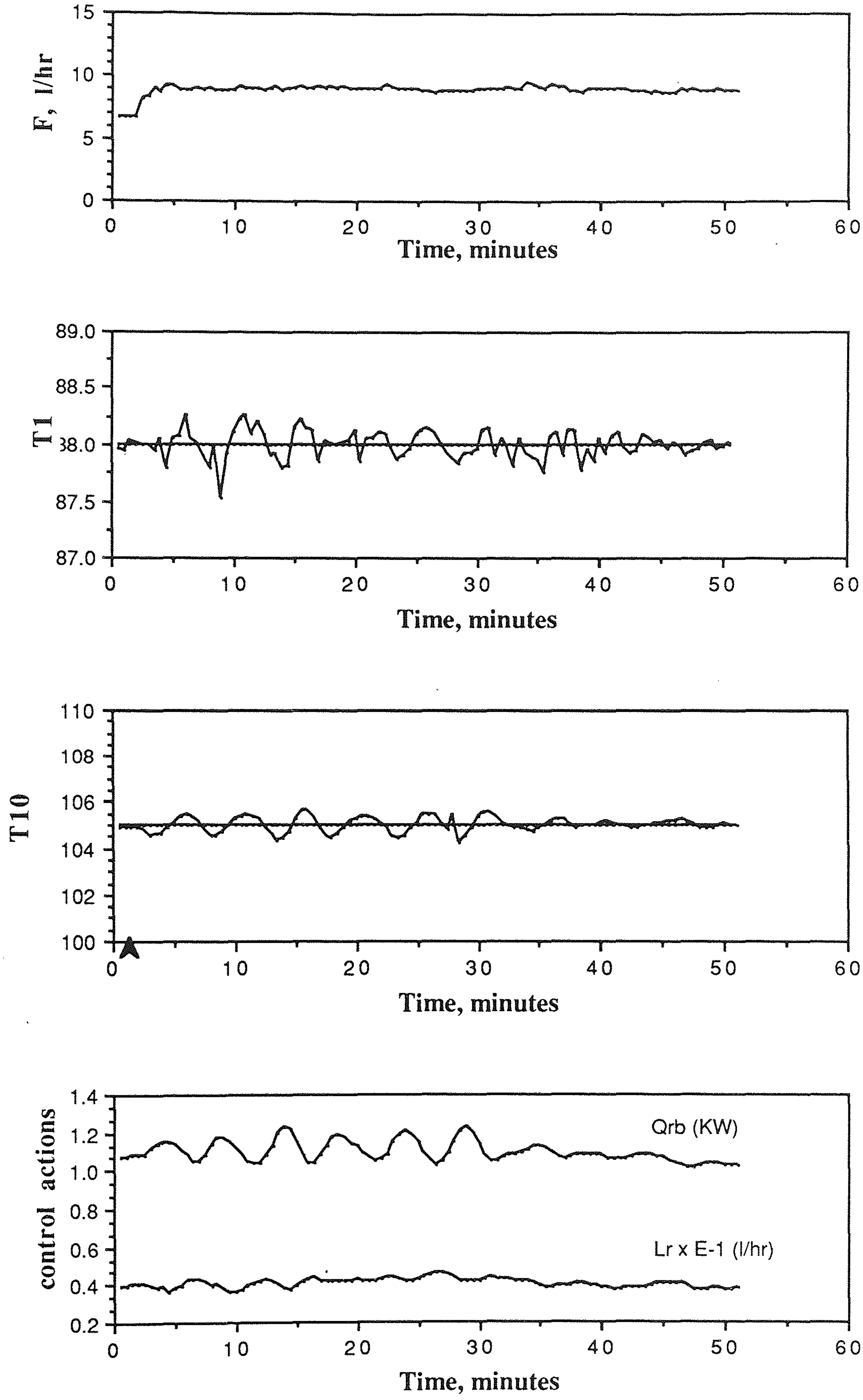


Figure 10.7 IMD3-STC control in presence of feed flow disturbances

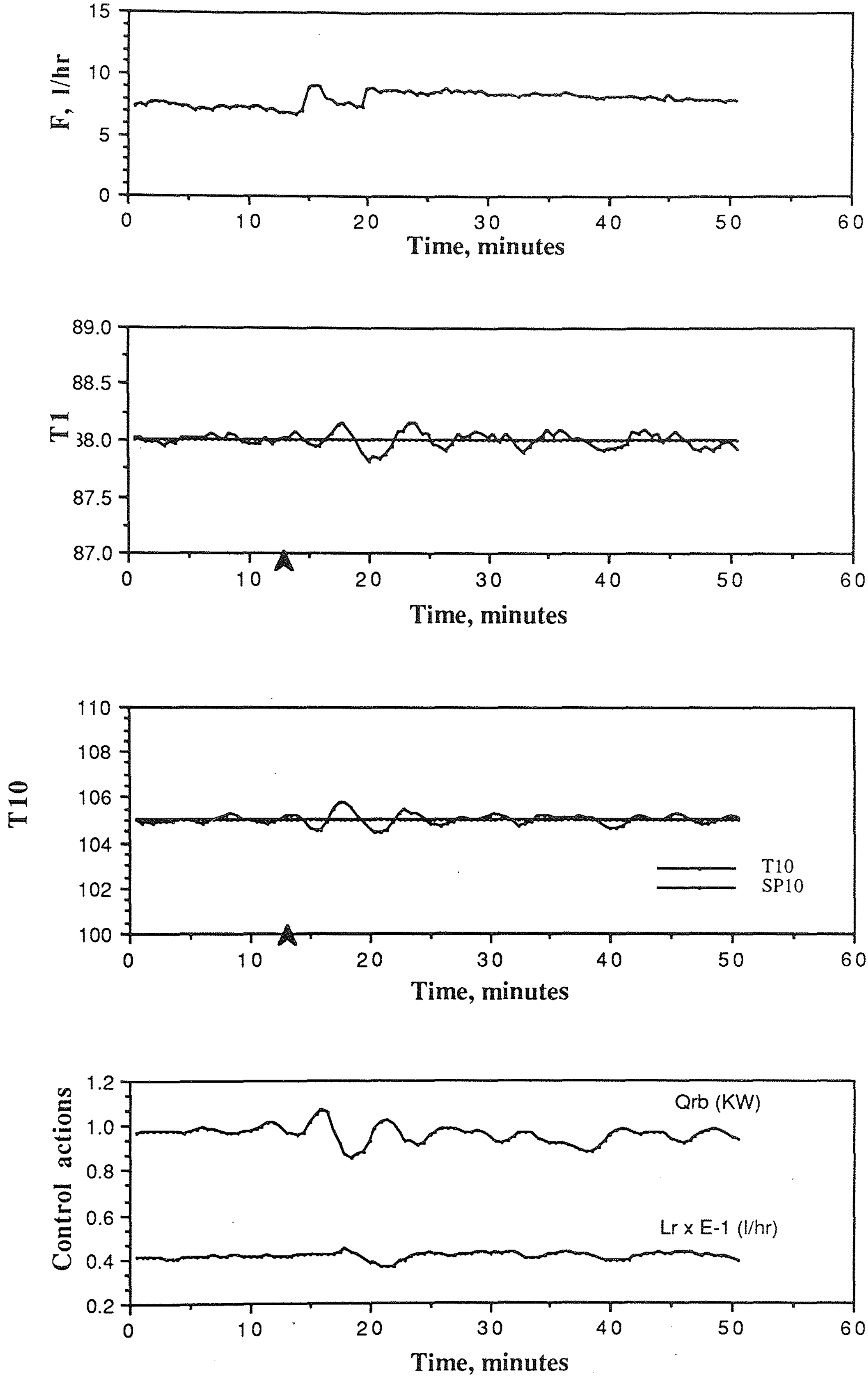


Figure 10.8 IMD1-STC control in presence of feed flow disturbances

(4) Changes in the outflow flowrates and the composition of the products of the column should be smooth and gradual to avoid disturbance of downstream process equipment, so creating unnecessarily high performance control specifications for these units.

11.2 Modelling of the distillation column: The validity of the column simulator

The pilot scale distillation column used in this work has been interfaced with a real-time multitasking microcomputer System96 through an interface box called the Monolog. This arrangement, described fully in Chapter 4, is flexible in that the computer can be used to monitor simultaneously more than one pilot plant. It is appropriate for studying the dynamic behaviour and computer control of distillation columns.

The arrangement is useful for testing the performance and screening control systems for industrial sized versions of distillation columns. One reason for this are the operational problems of the pilot plant distillation column itself. These include the inaccuracies in flowmeter measurements, flowmeter failure, difficulty in calibrating the heat input to the column with the digital signal from the computer and the inability to achieve satisfactory PI control of the reflux drum and reboiler drum liquid levels. These difficulties prevented the long term operation of the column, and made difficulties in modelling the column accurately, but they provided useful situations in which the robustness of the controllers could be tested.

A non-linear dynamic model of the distillation column has been derived as described in Chapter 5. The model, called the column simulator, is based on mass, energy and equilibrium relationships. The assumptions made in deriving the column simulator included constant molar holdup, 90% tray efficiency, adiabatic conditions, and liquid holdups on the trays and in the reboiler and reflux drums are well mixed. The only significant dynamics in the model are due to the dynamics of the component balances of the liquid on each tray.

The column simulator has been tested for validity with experimental data, as discussed in Chapter 5. This model was considered adequate for use in the design, analysis and screening of control systems for use on the experimental column. The validity tests showed that the model adequately modelled the direction of the responses of the tray temperatures, but it exaggerated the speed of response and the process gains of the experimental column.

In summary, the modelling and model verification exercises showed that an adequate non-linear model had been developed satisfactorily with only the tray composition dynamics and the dynamics of the reflux drum and reboiler drum liquid contents in the model for the pilot plant distillation column.

The similarity in some of the observations made from the performance of the controllers on the column simulator and the pilot plant distillation column strengthen the confidence placed on the column simulator as a reasonable good model for the design, analysis and screening of control systems for the pilot plant. For example, the ISV-STC is capable of providing tighter control than the PI and PSV-STC controllers. This indicates that there is no need to improve the model if its purpose is for the design, analysis and screening of control schemes for the pilot plant distillation column. There are, however, several cases where the behaviour of the column simulator and the pilot plant differ significantly when under control, which indicate the need for improving the column simulator once the modelling aspects responsible for the differences are identified. Some such cases are the exeggeration of the speed of response and the gains of the column and the significant errors in the predicted values of the intial steady state values.

11.3 The decoupling and disturbance rejection control scheme

This controller is a linear multivariable controller that allows the specification of closed loop poles to obtain desired responses of the controlled outputs. The objective of the control scheme is to reject the effects of unmeasurable load disturbances on the outputs and provide setpoint tracking while simultaneously effecting dynamic decoupling of the closed loop system. The controller consists of a constant state feedback for load disturbance rejection and a constant precompensator for setpoint tracking. The synthesis procedure for obtaining the state feedback and precompensator matrices, the number of poles that can be assigned to each decoupled input-output channel, and the procedure for computing the minimum number of state variables necessary to achieve complete decoupling, have been described in Section 6.2.

A linearised state variable model of the distillation column, obtained by linearising the column simulator, was used for controller design. In this model, the 10 tray compositions and the compositions in the reflux drum and the reboiler drums were the state variables. The state matrix of this model is tridiagonally dominant, as is usual for a binary distillation column. Two cases were considered; one where it was assumed that the non-dominant terms should be considered to be zero and one where these terms were retained. For the case where the small terms were set to zero, a minimum number of 6 state variables were required to be fedback to achieve decoupling control. When the small terms in the off-tridiagonal where retained, then the synthesis procedure required that all the 12 state variables, the 10 tray compositions and the reflux drum and reboiler drum compositions, should be fedback. The calculated weighting in the state feedback matrix is small compared to the remainder (Table 6.3). To measure all the 12 state variables would be difficult to justify in practice since it means composition analysers would have to be located on each tray, which would be expensive to carry out for small benefit. The off-diagonal terms were therefore eliminated in order to test what would be more practical.

The sensitivity of the results of the synthesis procedure to the small terms in the state matrix is an example of how small model errors can result in the requirement of an unnecessarilly complicated controller structure. It is necessary to identify such cases in practice so as to obtain a control system with a practical controller structure.

Results in Chapter 6 have shown that, for both cases considered above, when the controlled variables are the top tray and bottom tray compositions, two closed loop poles can be assigned freely in the controller. For these poles, each output was thus assumed to follow a first order response.

With the small terms in the system matrix set to zero, the minimum number of state variables which were required to be measured for feedback was affected by the choice of the controlled variables. With the distillate and bottom products compositions selected as the controlled variables, a minimum number of 4 state variables (1st, 2nd, n-1th and the nth state variables) were needed for feedback to achieve complete decoupling control. The coefficients of the state feedback matrix and the precompensator were quite large in magnitude, indicating that the controller may be very sensitive and may therefore result in large control actions and oscillatory behaviour or even instability of the distillation system, as discussed in Section 2.2.4. With the top tray and bottom tray compositions as the controlled variables instead, a minimum of 6 state variables were required to be fedback. This is a significant 50% increase in the number of measurements; a similar increase in the cost if each of the composition analysers costs the same. The coefficients of the state feedback matrix and the precompensator matrix were much smaller, however, than those for when the product compositions were the controlled variables. This indicated a controller that is less sensitive which should produce a more operable system compared to that where controlled variables are the product compositions. Simulation results on the linear model confirmed this, as is shown in Figure 6.1; the feedback controller obtained with the product compositions as the controlled variables always resulted in controller saturation regardless of the pole assignments; the controller with the end tray compositions as the controlled variables gave satisfactory results (see Figures 6.2, 6.3 and 6.6).

This is an interesting result, as it agrees with previously published results on the best choice of locations to take tray composition, or temperature, measurements in a distillation column for the control of the product compositions. The usual aim is to select locations where the corresponding tray compositions would show the satisfactory sensitivity to the control input to obtain the best possible quality of control. Several analysis such as those using techniques like the singular value decomposition (SVD) analysis (Yu and Luyben (137) and modal analysis (Levy et al. (74) and Shimizu & Mah (117)), and simple perturbation tests using models (Deshphande (168)), consistently show that measurements in the locations towards the ends show are better choices than the product compositions.

11.3.1 Load disturbance rejection

The decoupling controller was unable to eliminate offset in the bottom tray composition in the presence of a feed flow disturbance. This was demonstrated by simulation on the linear model. This is consistent with the published results of Shimizu and Matsubara (113), who applied the method on a non-linear model of a 10-stage binary ethylene-ethane distillation column. It also agrees with the results of Takamatsu et al. (130) who applied a state feedback controller based on the geometric approach of Wonham and Morse (178) to a non-linear model of a binary distillation system; the similarities between the disturbance rejection controller used by Takamatsu et al. and the state feedback decoupling controller used in this work have been mentioned in Section 2.6.4.

Discussion of results in Section 6.4.1 has shown that the magnitude of the poles assigned should be made as large and negative as possible to minimise the effects of load disturbances on the controlled outputs. Increasing the magnitude of the poles also helps to further reduce the offset in the bottom tray composition caused by feed flow disturbance, and this is achieved with reduced control effort. The behaviour of the closed loop system under the decoupling controller is similar to the behaviour of a modal controller. As shown by Davison (30), maximising the eigenvalues of the closed loop of the modal controller, through increasing the diagonal elements of the state feedback of the modal controller, minimises the effect of disturbances on the state variables of the system.

11.3.2 Combined feedback and feedforward compensation

Feedforward compensation is widely known to be beneficial when the load disturbances can be measured because the major effects of the disturbances can be suppressed before they significantly affect the system. As discussed in Section 1.1, a combination of feedback and feedforward can be used in order to improve the quality of control, because, ideally, the feedforward would counteract most of the effect of the disturbances leaving the feedback to provide residual control.

In the case of the decoupling and disturbance rejection controller, state feedback alone cannot achieve the complete rejection of the effects of disturbance of the feed flow rate from the bottom tray, or bottom product, composition as has been shown in this work and in Shimizu and Matsubara (113). Shimizu and Matsubara suggested the use of a feedforward compensator, such as the compensator $\mathbf{T_f}$ (Equation 6.11) in order to accomplish this objective. It was expected that the combination of state feedback and this feedforward compensator would improve the performance of the decoupling controller in the presence of feedflow disturbances. Simulations carried out to investigate this possibility showed otherwise. The feedforward compensator on its own was unable to remove completely the effects of either feed flow or feed composition disturbances from the outputs, as steady state offsets occurred in both cases (Figures 6.4 and 6.7.). This demonstrates the mathematical derivation in Shah (207), and also stated in Shimizu and Matsubara (113), that the feedforward compensator will not suppress completely load disturbance effects if the number of state variables is more than the number of controlled outputs.

The performance of the combined state feedback and feedforward compensator was even worse, particularly in the presence of feed composition disturbances (Figure 6.8) where saturation of the controller occurred. This clearly showed that there was no benefit to be gained for load disturbance rejection by combining the feedforward compensator with the state feedback controller of the decoupling and disturbance rejection controller. A possible explanation for the failure of this combination may be that adding the feedforward compensator to the state feedback causes the resulting controller to loose the decoupling properties which the original controller possesses. This occurs because the feedforward compensator is not designed to achieve decoupling control, but to simply reject disturbance effects from the outputs.

11.3.3 Setpoint tracking

The purpose of the $\mathbf{G}$ in the decoupling controller (Equation 6.1) is to equip the controller with the ability to provide setpoint tracking. The precompensator is also designed with the objective of decoupling the system. As mentioned on Section 3.3.1, a diagonal matrix **K*** is a direct function of the precompensator, and guidance on the choice of **K*** was not discussed by previous workers who have applied the technique.

It was found in this work that a strong link exists between the values of the pole assignments and the appropriate values the diagonal elements of **K*** should have to achieve accurate setpoint tracking. The diagonal elements of **K*** must be chosen as $\mathbf{K}^* = -\mathbf{M}_0$ to avoid steady state offsets in the controlled outputs, where $\mathbf{M}_0$ is a diagonal matrix with elements corresponding to the "leading" pole assignments (Sections 6.4.2 and 6.4.5). This is in contrast to the $\mathbf{K}^* = \mathbf{I}$ that could be concluded from previous work (eg. Takamatsu et al. (130), Shimizu and Matsubara (113), Power (209)) as discussed in Section 3.3.1.

It was also found that $\mathbf{K}^* = -\mathbf{M}_0$ must be selected in the case where two closed loop poles were assigned for each output response, indicating that a different number of poles assigned would not dictate a different value of **K***. This could not be proven algebraically as was demonstrated in Section 6.4.5, but it was confirmed by numerically solving for **K*** the steady state gain relationship between the setpoints and the outputs of the closed loop system (Equation 6.23).

11.3.4 Robustness to non-linear effects

It was necessary to test the robustness of the decoupling and disturbance rejection controller to non-linear effects, in order to ascertain whether the technique would perform satisfactorily on the pilot plant distillation column. This was first carried out by applying the controller to a linear state variable model obtained at steady state conditions different from the steady state condition used to design the controller. Large offsets in the top tray and bottom tray compositions resulted, when the model was subjected to feed flow disturbance (see Figure 6.9). The controlled system did not become unstable, which indicated some degree of robustness of the controller to the non-linear effects.

A more rigorous test of the robustness of the controller to non-linear effects was carried out by applying the controller to the column simulator, which has non-linear effects typical of those of the real column. In this case (see Figure 6.23), the performance of the controller was unacceptable as the manipulated inputs saturated even in the absence of load disturbances or setpoint changes. The cause of the bad performance of the controller on the column simulator was probably because the controller matrices had very large elements, so that even effects as small as computer roundoff on the process variables could result in small changes in the control inputs. This could bring non-linear effects into play and, therefore, poor controller performance.

The poor performance of the controller in the face of non-linear effects is consistent with the discussion in Section 2.2.4, that neglecting model uncertainties in controller design can lead to a controller which may be inoperable in practice. The model uncertainty in the case under discussion is due to ~~the~~ neglecting ~~of~~ the non-linear effects of the column simulator, as the state variable model used for controller design is linear. It was also shown in Section 5.10 that the linear state variable model used for the controller design grossly underestimates the gains of the column simulator (Figures 5.14a and 5.14b), also predicts the possibility of a very sensitive controller (unduly large controller coefficients), as a feedback controller is an approximate inverse of the transfer function matrix of the plant which the controller is required to control (Section 2.2.5).

The poor performance of the controller on the column simulator also contrasts with previously published results such as Shimizu and Matsubara (113). They applied the controller to a non-linear model of a 10 stage binary ethylene-ethane distillation column and their results demonstrate that the controller achieved its objectives. This showed that the controller is reasonably robust to the non-linear effects in the distillation column model. A similar conclusion can, however, not be reached in this work, judging from the poor performance on the column simulator even when no disturbances, except computer roundoff, were effecting the column simulator values.

The fast dynamic responses of the tray compositions of the column simulator may also be significant factor responsible for the poor performance of the controller on the column simulator. Rough estimates of the time constants of the tray composition responses in the column simulator show that they are of the order of 2 to 4 seconds. This is very small compared with the time constants of the tray compositions of the distillation column model used by Shimizu and Matsubara (113), which range from 42 to 46 seconds. The poor performance of the control scheme on the column simulator and the acceptable performance of the control scheme on the non-linear model of Shimizu and Matsubara (113), is consistent with the well known fact (see Section 2.2.3) that a slower responding system is more robust to model uncertainties under closed loop control than a faster responding system. From the above arguments, it is expected that the decoupling and disturbance rejection controller would perform better on a non-linear model of an industrial sized version of the pilot plant distillation column, as the dynamic responses would be much slower. However, the poor performance of the controller in presence of non-linearities makes it unlikely that the decoupling controller would be a preferred control scheme over a PI control scheme, for example. This raises the question as to whether the pilot plant distillation column used in this work is useful for testing control systems for distillation columns. The pilot plant is useful in many respects. One is for testing control systems for small scale processes of similar scale as the pilot plant where, for example, the production rate is small but the value of the products are significant. Another is that any control system that operates satisfactorily and is robust under the operating conditions of the pilot plant is certain to operate reliably on an industrial scale version of the column which would have a much slower dynamic response.

Judging from the simulation results on the column simulator, on-line application of the decoupling and disturbance rejection control scheme to the pilot plant distillation system was not practical. A linear state variable model of the column simulator is therefore not suitable for designing a multivariable controller of the form of the decoupling and disturbance rejection controller for use directly on the column simulator, and is, therefore, unsuitable for controller design for the real column. This is unless a method is devised which compensates for the non-linear effects and other model uncertainties in the model at the design stage of the controller. This justify the need for an adaptive form in order to make the method directly applicable on the pilot plant distillation column, and to non-linear systems with significant non-linear effects. These considerations were discussed in Section 6.6.

11.3.5 Comparison with PI control

Simulation results on the linearised model show that for both load disturbances and setpoint changes, the decoupling and disturbance rejection controller offers great advantages over PI control. The decoupling controller provided faster closed loop responses, output responses which did not suffer from closed loop interactions, and the settling times of the closed loop system could be made much shorter without risking system stability by increasing the magnitude of the poles assigned. The exception is that the decoupling controller was unable to eliminate such offsets as the offset in the bottom tray composition from a feed flow disturbance, as it does not have integral action. In the presence of non-linearities, however, the situation is different as has been discussed above. The multiple loop PI controllers were able to provide stable control of the column simulator but the decoupling and disturbance rejection controller could not. There is therefore no benefit to be gained from using the decoupling controller rather than PI for the dual control of the product compositions of the column simulator, and also the pilot plant distillation system.

11.3.6 Addition of integral and derivative modes in to the decoupling and disturbance rejection controller

Integral and derivative modes were added to the decoupling and disturbance rejection controller by following analogy with conventional PI control. The integral mode equipped the controller with the ability to remove the offset which is caused by various effects such as feed flow disturbances on the bottom tray composition, non-linear effects and the wrong choice of $\mathbf{K}^*$ in the precompensator. Each mode requires only m extra differential equations to be solved so that the increase in computational overheads is small, where m is the number of controlled outputs. The appropriate integral and derivative times were selected by classical design techniques such as the Cohen and Coon method used to assist the tuning of the conventional PI controllers applied in this work. Adjustment of the pole assignments of the controller adjusts its proportional action so that the appropriate integral time must be carefully selected in conjunction with the pole assignments.

The discussion of the results in Section 6.7 has shown that the controller with the integral mode included is able to remove offset in the bottom tray composition due to feed flow disturbance and the offset due to the wrong choice of $\mathbf{K}^*$. As is expected when integral action is applied to a controller with proportional action only, the integral mode increased the sensitivity of the closed loop system as oscillatory closed loop responses were produced. The derivative mode was able to improve the robustness of the controller with integral mode, as is expected of a derivative mode to affect a proportional-plus-integral controller.

The integral mode was unable to remove offset due to non-linear effects and, in fact, poorer performance resulted compared with when the original state feedback controller alone was used (see Figure 6.27). This has been attributed to two reasons, one is the greatly increased sensitivity of the closed loop system with the introduction of the integral mode, the other is the sensitivity of the original state feedback controller to non-linear effects as discussed earlier. These two factors combined to produce an inoperable control system in the presence of non-linear effects of the column simulator.

The successful operation of the decoupling and disturbance rejection controller with the integral and derivative modes included is an interesting result. It demonstrates how the capabilities of multivariable controllers which have the form of a state feedback for disturbance rejection and a precompensator for setpoint tracking can be improved by following the simple approach used in this work to include integral and derivative modes into the control law. This has been achieved with minimal extra computational overheads in implementing the controller. The integral and derivative times for each "control loop" can be selected independently using conventional SISO controller design techniques which are easy to use.

11.4 The off-line Kalman filtering studies

Estimation studies were carried out in off-line mode using experimental data obtained from the pilot plant distillation column. A Kalman filter which uses a linear state variable model of the column simulator was assessed for the purpose of estimating the tray compositions of the distillation column from available process measurements. The objective was to enable the indirect, or estimator aided, control of the products of the pilot plant distillation column from measured values of some tray temperatures and inputs into the column. A total of 5 tray temperatures (1st, 2nd, 7th, 9th and 10th tray temperatures) and 4 inputs (reflux flow, reboiler heat input, feed composition and feed flowrate) were supplied to the filter.

The discussion of the estimation studies in Chapter 7 showed certain weaknesses in the Kalman filter. The filter produced unstable estimates of the tray compositions whose tray temperatures were not supplied to the filter. The estimates of the composition of the other trays tracked their true values, but the filter could not remove the biases in these estimates. This is a limitation to application of estimator aided control of the products of the distillation system since the filter must be able to remove biases from the estimates it produces for it to be considered practical for on-line application.

The cause of these weaknesses is primarily the gross inaccuracy in the filter model used, as well as the long sampling interval of 30s. The model inaccuracies mainly stem from the filter model's underestimation of the gains of the column simulator which arises mainly from the neglect of non-linear effects of the column simulator. The linearised state variable is therefore also unsuitable as the filter model in a Kalman filter which is to be used in design of an estimator aided control policy for the distillation column.

Extra tray temperature measurements and a more accurate filter model are required to combat these weaknesses of the filter. In the case of a more accurate model, the column simulator would be a good choice. All these would, however, incur a considerable increase in program memory requirement and the filter execution time, both of which are already considerable. The time required to complete a filter cycle consisting of estimation and prediction of the state variables was 72 seconds with the 5 tray temperatures and 4 inputs supplied to the filter; this is more than twice the measurement sampling interval of 30s which is also the recommended control interval.

The long filter cycle time, the considerable program memory requirement and the inability of the filter to produce stable and unbiased estimates of all the tray compositions, all prevented the on-line application of the Kalman filter. The estimator aided control of the product composition of the distillation column, using, for example, the EAFF of Daie (26), could therefore not be carried out.

11.5 Evaluation of the self tuning controllers on the column simulator

The single loop self tuning controller (SV-STC) and multiple loop self tuning controllers (MD1-STC, MD2-STC and MD3-STC), described fully in Chapter 8, were applied on the column simulator to assess their practicality for on-line application to the pilot plant distillation column. The self tuning controllers are simple in their design due to the assumptions made in their design.

11.5.1 Single loop top tray temperature control

The results in Section 9.3 has shown that the positional and incremental forms of SV-STC, PSV-STC and ISV-STC, gave significantly better regulatory and servo performances than the PI controller in the control of the top tray temperature. The PSV-STC improved on the IAE by 40% and the ISV-STC by 48% in the presence of unmeasured load disturbances (feed flow and feed composition disturbances) and setpoint changes. These improvements would, in practice, represent a significant reduction in the off-specification of the top product of the distillation column. The regulatory and the servo performances of the ISV-STC was also shown to be better than the PSV-STC (Figure 9.5).

The circumstances under which each one of the single loop controllers, PI, ISV-STC and PSV-STC, would be favoured for the other two would be different. If, for example, the products and the outflows of the column are required to change smoothly so as not to greatly disturb downstream processing units, the PI controller would be the most appropriate choice of the three because the closed loop response it provides is smooth and of first order type closed loop response. If closer control is required, and fast and large changes in the manipulated variables are acceptable, then either PSV-STC or ISV-STC would be more appropriate to PI; since the ISV-STC is clearly better than PSV-STC, it is the preffered choice.

11.5.2 Simultaneous control of the top and bottom tray temperatures

Multiple loop PI controllers performed satisfactorily in the face of large unmeasured load disturbances and to setpoint changes, but were slow in returning the outputs to their desired values. Simplified decoupling, using rough estimates of steady state decouplers obtained using steady state gains from the step response tests on the column simulator, deteriorated the dynamic responses of the controlled system for both setpoint tracking and regulatory control; the performance of the multiple loop PI controllers were usually degraded by more than 20% with the steady state simplified decouplers (see Table 9.3). Since the steady state decouplers were estimates and therefore in error, the significant degradation in the performance of the multiple loop PI controllers was not suprising, as the column simulator is a non-linear system.

For either regulatory or servo control, the major benefit that is gained from using self tuning control (MD1-STC, MD2-STC and MD3-STC) is the much tighter control it offers compared with PI control, except for the case of PMD3-STC which was worse than PI. These improvements were more significant in the case of servo control than for regulatory control (see Table 9.3). Tighter control is, however, achieved with larger and faster control actions so that there is greater possibility of overshoot of the setpoint and oscillatory closed loop behaviour.

The comparison of PMD3-STC and IMD3-STC show clearly the benefit of using an incremental self tuning controller rather than a positional one. As explained in Section 9.4, the PMD3-STC does not provide decoupling so that interaction effects can be considered as unmeasured/unknown load disturbances. In Section 9.1, it was shown, using a simple first order linear model, that a positional STC is not able to compensate adequately for unmeasured/unknown load disturbances as it relies on accurate estimates of the disturbance effects, while the incremental form does not suffer from such problems as the control law provides an implicit estimate of the disturbance effects. It is this characterstic of an incremental STC that is considered to be the main reason why IMD3-STC performed better than PMD3-STC in both servo and regulatory control of the column simulator as was shown in Section 9.4.

11.5.2 The performance of the parameter correction methods

Self tuning control has been shown to offer significant benefits in the control of the column simulator. There was, however, a risk that some of the estimates of the controller parameters could attain bad values if, for example, insufficient excitation of the closed loop system resulted. As discussed in Section 2.9.7, this is a limitation to the application of adaptive control to steady state chemical processes, since at or near a steady state variation of the process variables would be relatively small. The problem would become more severe as the number of parameters that need to be estimated increases, since then stronger conditions of excitation are required.

When the estimates of the controller parameters attain bad values, poor controller performance or even instability of the closed loop system may result. An example of such as case was demontrated using IMD1-STC for servo control (Figure 9.28). Attempts were made to improve the performance of the IMD1-STC in these situations by the inclusion of the simplified parameter correction (SPC) and the parameter correction (PC) methods, both of which are variants of the algorithm suggested by Ossman and Kamen (94), as desribed in Section 8.3. The objective of each method is to force the estimates of the controller parameters into known bounds where it is known that acceptable control would result.

Simulation experiments were designed to assess the benefits that these two correction algorithms could offer in two situations:

a) one where the bounds of only the two diagonal elements of $\mathbf{G}(z^{-1})$ were specified, and

b) one where the bounds of all the four $\mathbf{G}(z^{-1})$ elements were specified.

Rough estimates of the bounds of the parameter estimates were chosen to represent the fact that the column simulator is a non-linear system so that the values of these parameters are, therefore, not accurately known. The width of the bounds were chosen to be wide to reflect these uncertainties (see Table 9.4). In both cases, the SPC method offered great improvements in the performance of IMD1-STC (see Table 9.5) with the improvement being higher for the second case. The PC method, on the other hand, gave a poorer performance (a 65% deterioration in the IAE) for the first case and an improvement of 20% in the second case, but even this improvement was much lower than that obtained when the SPC was used. The PC could not offer better improvements because, after the initial period of effective correction of the parameter estimates, the magnitude of the covariance matrix became very small giving negligible correction. This is a limitation in the application of the PC, a limitation that could be remedied by devising a means to maintain a large enough covariance matrix in the estimation algorithm to ensure effective correction at all times. An example of such a method was demonstrated (Figures 9.38 and 9.39) using a variable forgetting algorithm in Equation 9.3 that keeps the trace of the covariance matrix at a required level.

The improvements obtained by combining the SPC with IMD1-STC in the situations discussed above demonstrate the promise of the approach of adding a parameter correction algorithm in the estimator of a self tuning control algorithm. It confirmed that the SPC is a useful way of preventing the controller parameters from converging to wrong values and so retain satisfactory controller performance. As significant improvements were obtained even when the bounds of only the two diagonal elements of $\mathbf{G}(z^{-1})$ were specified, the conjecture made in Section 3.3.3 is also validated. This is that, when the problem of parameter estimates attaining bad values arise, it is possible to force only a subset of the parameters of the self tuning controller to their correct ranges and still retain satisfactory and robust controller performance.

11.6 Computer control of the pilot plant distillation column

Single loop top tray temperature control was successfully carried out on the experimental distillation column using PSV-STC, ISV-STC and PI control. The results demonstrate the capabilities of these algorithms on a real system and they support some of the observations made from the simulation studies. One of these is that the ISV-STC gives faster closed loop response and is capable of providing tighter control than both the PI and the PSV-STC. Also the parameter estimates of the ISV-STC changed significantly after setpoint and load dsiturbance changes. The performances of the multiple loop PI, IMD3-STC and IMD1-STC were also evaluated on the pilot plant. The results demonstrate that the IMD3-STC would not be preferred to the other two for the dual composition control of the pilot scale distillation column since control loop interactions caused significant oscillatory behaviour of outputs. The IMD1-STC, which provides decoupling of the control loops, also gave oscillatory responses of the outputs but was much better than IMD3-STC as the oscillations diminished quickly compared with that of IMD3-STC. On-line experiments also demonstrated the poor behaviour of the multiple loop PI controllers when steady state decoupling was included. On the basis of the experimental results, it was concluded in Chapter 10 the ISV-STC should be used for single loop control and IMD-STC for multiple loop control.

The time required to complete each self tuning control cycle consisting of data logging, controller parameter estimation and the calculation of the controller output, was usually between 12 and 15 seconds; the corresponding time was usually about 8 seconds for PI control. These times are within the sample time of 30 seconds, so that control actions were always effected well inside the interval. There is, however, a significant delay compared to the ideal of control action occurring at the sample instant. This delay did not appear to limit significantly the performances of the controllers. The controllers have also been shown to be robust against saturation of the reflux control input, particularly in the case of IMD3-STC in Figure 10.23, where saturation occurred several times in the duration of the experiment.

In summary, although the computational requirements of the self tuning controllers are significantly larger than for PI, the self tuning controllers were reliable and robust to control input uncertainties and controller saturation. They have been demonstrated to be able to give tight control of the products of the distillation column as well as adapt to very large disturbances on the pilot plant column.

11.8 Summary of conclusions

The following conclusions can be drawn from this work:

1) The pilot plant binary distillation column interfaced with a real-time multi-tasking microcomputer (System 96) is suitable for studying the dynamic behaviour and computer control of distillation columns. The column had some operational problems such as the frequent failure of flowmeters and the reflux flow vale failure, which are typical of an industrial environment and, therefore, suitable for testing the robustness and reliability of control systems

2) A non-linear model of the pilot plant distillation column has been derived based on mass and energy and vapour-liquid equilibrium relationships. The model satisfactorily reflect the open loop and the closed loop behaviour of the real column under PI and self tuning control in several situations. The model has been useful in the design, analysis and screening of control systems for the real column.

3) A disturbance rejection and decoupling control scheme which is a state variable feedback control technique based on a linearised model of the column simulator was designed for the column. The controller has been shown to be intolerant to non-linear effects of the column simulator and is therefore not viable for on-line application on the pilot plant.

4) When applied on the linearised model of the simulator, the combination of the feedforward compensator of Shah (207) with the state feedback of the

decoupling and disturbance rejection controller results in poor control for load disturbance rejection.

5) It has been established in this work that, in the decoupling and disturbance rejection controller, $\mathbf{K}^* = -\mathbf{M}_0$ must be satisfied in order to avoid steady state offsets when using the controller for setpoint tracking.

6) Integral and derivative modes have been successfully added to the decoupling and disturbance rejection control scheme by analogy with conventional PI control. The integral and derivative times are easy to choose using simple classical controller design techniques as guidance and the added computing effort is minimal. The integral and derivative modes can be readily added to controllers having the same form (state feedback and a precompensator) as the decoupling and disturbance rejection controller.

7) Extensive simulated and experimental studies confirm that self tuning control offers significant improvements over PI control for both single temperature control and dual temperature control of the distillation column. These improvements are in terms of increased speed of response, tighter control and better adaptation to changes in operating conditions.

8) The SPC algorithm is a simple approach to preventing the controller parameters from attaining bad values and so causing bad control in an adaptive control system using a recursive parameter estimation technique. Simulation results on a non-linear model of the distillation column using the implicit form of self tuning control have shown that SPC offers significant improvement even when only two of the controller parameters were prevented from attaining bad values.

11.9 Recommendations for further work

The areas were work is suggested for further research are as follows:

1) In the case of the disturbance rejection and decoupling control scheme, more work is suggested to establish that $\mathbf{K}^* = -\mathbf{M}_0$ for linear multivariable

systems on which the decoupling and disturbance rejection control scheme can be applied.

2) An adaptive form of the decoupling and disturbance rejection control scheme should also be devised in order enable the approach to be applied directly to non-linear systems. This would increase the possibility of applying the control scheme directly to the non-linear distillation column used in this work, and to non-linear systems in general.

3) The Estimator Aided Feedforward (EAFF) control scheme of Daie (26) should be tested on the column to verify the improvements obtained by Daie by simulation. The performance of the Kalman filter in estimates of the tray compositions should be improved and the computational requirement of the Kalman filter should be reduced to enable on-line application to be practical. As mentioned in Chapter 7, the model reduction procedure of Cho and Joseph (17, 18, 19) could be used to reduce the equations of the column simulator and therefore reduce the order of the Kalman filter in the EAFF scheme. These workers have shown that it is possible to reduce the non-linear equations of a distillation column model by a factor of 4 and the reduced model would still retain reasonably good accuracy. A faster more powerful microcomputer than the System96 is also needed for increased computational speed such that the filter cycle time can be reduced significantly to well below the sampling interval of 30 seconds

4) The SPC algorithm has been applied in this work the implicit self tuning control algorithms where the controller parameters are estimated directly. The SPC method should be applied with an explicit form of the self tuning controllers where the system parameters are first estimated and the used to calculate the controller parameters. This would ascertain whether the SPC would offer similar improvements that were obtained in this work.

5) The application of the simplified parameter correction, SPC, method is recommended to an experimental system were the number of parameters

are large and difficulty is encountered in estimating all of them. This would demonstrate if the algorithm offers improvements similar to the improvements obtained in this work by simulation on the column simulator. The use of the SPC to aid or speed up the initial tuning of self tuning controller parameters is also worthwhile investigating.

6) As mentioned earlier in the thesis, the column was not operated for periods longer than 80 minutes at any one time due to practical problems on the pilot plant. Application of the self tuning algorithms, and indeed any control algorithm that is investigated, should be tested for longer periods of time, say 8 to 12 hours or even days, as such investigations will be more representative of industrial operation. This means the laboratory distillation process needs to be modified to be able to handle longer periods of operation.

References

(1) Arkun,Y., Manousiouthakis, B., Palazoglu, A. and Guruswamy, V., "Computer aided analysis and design of robust multivariable control systems for chemical processes", Computers and Chemical Engineering 1985, **9**(1), 27.

(2) Asbjornsen, O.A., "Challenges in mordern process control", Computers and Chemical Engineering, 1984, **8**(5), pp. 275.

(3) Agarwal, M. and Seborg, D.E., "A multivariable non linear self tuning controller", AIChE Journal, 1987, **33**(8)

(4) Astrom, K.J, "Theory and applications of adaptive control : A survey", Automatica, 1983, **19**(5), pp. 471.

(5) Astrom K.J., "A Commentary on the Rohrs et al. Paper "Robustness of continuous - time adaptive control algorithms in the presence of unmodelled dynamics", IEEE Transactions on Automatic Control, 1985, AC - **30**(9), pp. 889.

(6) Astrom, K.J., Hagander, P. and Sternby, J., "Zeroes of Sampled Systems", Automatica, 1984, **20**(1), pp. 31.

(7) Anderson, B.D.O., "Adaptive systems, lack of persistency of excitation and bursting phenomena", Automatica, 1985, **21**(3), pp. 247.

(8) Anderson, B.D.O. and Johnson, C.R., "Exponential convergence of adaptive identification and control algotrithms", Automatica, 1982, **18**(1), pp. 1.

(9) Astrom, K.J. and Wittenmark, B., "Self tuning controllers based on pole-zero placement", IEE Proceedings, 1980, **127**(3) Part D, pp. 121.

(10) Beaverstock, M.C. and Harriot, P.L., "Experimental closed loop control of distillation columns", Industrial and Engineering Chemistry Process Design and Development, 1973, **12**(4).

(11) Berger, C.S., "Sef-Tuning control of offset using a moving average filter", IEE Proceedings, 1986, **133**(4) Part D, pp. 184.

(12) Boland, F.M. and Giblin, J., "Non-linear control of non-minimum phase systems", IEE Proceedings., 1982, **129**(4) Part D, pp. 118.

(13) Berger, C.S., "Robust controller design by minimisation of the variation of the coefficients of the closed-loop characteristic equation", IEE Proceedings, 1984, **131**(3) Part D, pp. 103.

(14) Badr, O. Rey, D. and Laded, P., "Adaptive dual control for a multivariable distillation processes", Digital Computer Applications to Process Control, Van Nauta-Lemke eds. IFAC and North Holland Publishing Company , 1977, pp. 141.

(15) Bell, D. and Griffin, A.W.J., "Modern control theory and computing", 1969, McGraw-Hill, London.

(16) Brierly, R.J.P., " DistPack : Using a combination of algorithms to solve difficult distillation and absorption problems", Distillation '79 IChemE Symp. Ser., No 56, 1979, 2.6 / 89.

(17) Cho, Y.S. and Joseph, B., "Reduced-Order steady state and Dynamic models for separation processes - Development of the Model Reduction procedure", AIChE Journal,1983, **29**(2), pp. 261 - 269.

(18) Cho, Y.S. and Joseph, B., "Reduced-Order steady state and Dynamic models for separation processes - Application to non-linear multicomponent systems", AIChE Journal, 1985, **29**(2), pp. 270-275.

(19) Cho, Y.S. and Joseph, B., "Reduced-Order models for separation column - Application to columns with multiple feeds and sidestreams", Computers and Chemical Engineering, 1984, **8**(2), pp. 81 - 90.

(20) Clarke, D.W, "Self-tuning control of non minimum phase systems", Automatica, 1984, **20**(5), pp. 517.

(21) Clarke, D.W., Hodgson, A.J.F. and Tuffs, P.S., "Offset problem and k-incremental predictors in self-tuning control", IEE proceedings, 1983, **130**(5) Part D, pp. 217.

(22) Clarke, D.W. and Gawthrop, P.J., "Implementation and application of microprocessor-based self-tuners", Automatica, 1981, **17**(1), pp. 233.

(23) Chien, I.L., Seborg, D.E. and Mellichamp, D.A., "Self tuning control with decoupling", AIChE Journal, 1987, **33**(7), pp. 1079.

(24) Chien, I.L., Mellichamp, P. and Seborg, D.E., "Multivariable self tuning control strategy for distillation columns", Industrial Engineering Chemistry Process Design and Development, 1986, **25**(3), pp. 595.

(25) Coppus, G.W.M., Shah, S.L. and Wood, R.K., "Robust multivariable control of a binary distillation column", IEE Proceedings, 1982, **29**(1) Part D, pp. 21.

(26) Daie, S., "Computer Control of Chemical Process Plants with special reference to Distillation", Ph.D Thesis, 1980, University of Aston in Birmingham, Gosta Green, Birmingham, U.K.

(27) Dexter, A.L., "Self tuning control algorithm for single-chip microcomputer implementation", IEE Proceedings, 1983, **130**(5) Part D, pp. 255.

(28) Information Display Products, "Plot 10 User's Manual", 1972,Tektronix Inc., P.O. Box 500, Oregun, U.S.A.

(29) Microware System Coporation, "System96 User's Manual", 1983, Microware System Coporation, Des Moines, Iowa, U.S.A.

(30) Davison, E.J., "Control of a distillation column with pressure variation", Transactions of the Institution of Chemical Engineers, 1967, 45, pp. T229.

(31) Dahlqvist, S.A., "Control of a distillation column using self tuning regulators", The Canadian Journal of Chemical Engineering, 1981, **59**, pp. 118.

(32) Furzer, I.A., "A new rapid algorithm for the simulation of distillation column ", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, B529.

(33) Foss, A.S., "Critique of chemical process control theory", AIChE Journal, 1973, **19**(2), pp. 209.

(34) Fortescue, T.R., Kershenbaum, L.S. and Ydstie, B.E., "Implementation of self tuning regulators with variable forgetting factors", Automatica, 1981, **17**(), pp. 831.

(35) Fuchs, J.J, "Recursive least-squares algorithm revisited", IEE Proceedings. 1981, **128**(2) Part D, pp. 74.

(36) Gunn, D.J., Ali, S.A. and Rashid, K., "A stable accurate method for the dynamic simulation of Distillation", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. B25.

(37) Gallun, S.E. and Holland, C.D., "Gears procedure for the simultaneous solution of differential and algebraic equations with application to unsteady state distillation problems", Computers and Chemical Engineering, 1982, **6**(3), pp. 231.

(38) McDermott, P.E, Mellichamp, D.A. and Rinker, R.G., "Pole-Placement selftuning control of a fixed - bed autothermal reactor: Multivariable Variable Control", AIChE Journal, 1986, **32**(6), pp. 1015.

(39) Goodwin, G.C. and Sin, K.S., "Adaptive Filtering Prediction and Control", 1984, Prentice-Hall.

(40) Measurement Systems Limited, "Monolog User Manual", Measurement Systems Limited, Mill Reef House, 9-14 Cheap Mill street Newbury, Berkshire, U.K.

(41) Guilandoust, M.T, Morris, A.J. and Tham, M.T., "Evaluation and control of distillation product composition using tray temperature measurements", IFAC Conference DYCORD '86, Bournemouth, Dec 1986.

(42) Measurement Systems Limited, "Monolog Link software User Manual", Measurement Systems Limited, Mill Reef House, 9-14 Cheap Mill street Newbury, Berkshire, U.K.

(43) Gawthrop, P.J., "Self tuning PID controllers : Algorithms and Implementation", IEEE Transactions on Automatic Control, 1986, AC-**31**(3), pp. 201.

(44) Hadju, H., Folders, P. and Mizsey, P., "New interation effects in distillation column dynamics", Distillation '79 IChemE Symp. Ser., No 56, 1979, 2.6/67.

(45) Kinoshita, M., Hashimoto, I. and Takamatsu, T., "A new simulation procedured for multicomponent distillation column processing non-ideal solutions or reactive solutions", Journal of Chemical Engineering of Japan, 1983, **16**(5), pp. 370.

(46) Kinoshita, M., Hashimoto, I. and Takamatsu, T., "A simulation procedure for multicomponent distillation column within which three phases of vapour and two partially immiscible liquids are present", Journal of Chemical Engineering of Japan, 1983, **16**(3), pp. 513.

(47) Hammarstrom, L.G., Waller, K. V. and Fagervik, K. C., "On modelling accuracy for multivariable distillation control", Chemical Engineering Communication, 1982, **19**, pp. 77 - 90.

(48) Holland, C.D and Liapis, A.I, "Computer methods for solving dynamic separation problems", 1983, McGraw-Hill.

(49) Hamilton, J.C., Seborg, D.E. and Fisher, D.G., "An experimental evaluation of Kalman Filtering", AIChE Journal, 1973, **19**(5), pp. 901.

(50) Hesketh, T., "State-space pole-placing self tuning regulator using input-output values", IEE Proceedings, 1982, **129**(4) Part D, pp. 123.

(51) Hutchinson, J.F. and McAvoy, T.J., "On-line control of a non-linear multivariable process", Industrial and Engineering Chemistry Process Design and Development, 1973, 12(3) pp. 226.

(52) Cheng, T.C. and Ward, T.J., "Non-interacting control", Industrial and Engineering Chemistry Process Design and Development, 1975, 14(2), pp. 193.

(53) Treybal, R.E., "Mass Transfer Operations", 1980, 3rd edition, McGraw-Hill, Kogakusha.

(54) Isermann, R., "Parameter adaptive control algorithms : A Tutorial", Automatica, 1982, **18**(5), pp. 513.

(55) Isermann, R., "Practical aspects of process identification", Automatica, 1980, **16**, pp. 513.

(56) Burden, R.L., Faires, J.D. and Reynolds, A.,"Numerical Analysis", Prindle, Weber and Schmidt, Boston, Massachussets, 2nd edition, 1981.

(57) Jafarey, A. and McAvoy, T.J., "Steady state feedforward control algorithms for reducing energy costs in distillation", ISA Transactions, 1980, **19**(4), pp. 89.

(58) Jerome, N.F and Ray, W.H., "High performance multivariable control strategies for system having time delays", AIChE Journal, 1986, **32**(6), pp. 914.

(59) Jensen, N., Fisher, D.G. and Shah, S.L., "Interaction analysis in multivariable control systems", AIChE Journal, 1986, **32**(6), pp. 959.

(60) Jung, H.Y. and Lee, W., "Modified derivative decoupling control of non linear multivariable process", Journal of Chemical Engineering of Japan, 1984, 17(1), pp. 26.

(61) Kumar, S., Wright, J.D. and Taylor, P.A., "Modelling and Dynamics of and Extractive Distillation Column", Canadian Journal of Chemical Engineering, 1984, **62**, pp. 780.

(62) Kisakurek, B., "A predictive model for dynamic distillation ", Chemical Engineering Communication, 1983, **20**, pp. 63-79.

(63) Toivonen, H.T., "Variance constrained self tuning control", Automatica, 1983, **19**(4), pp. 415.

(64) Fletcher, J.P., "A method for the rigorous calculation of distillation columns using a generalised efficiency model ", Institution of Chemical Engineers, Symposium Series, No. 104, 1987, pp. A437.

(65) Keyser, R.M.C. and Van Cauwenberghe, A.R., "A self-tuning multistep predictor application", Automatica, 1981, **17**(1), pp. 167.

(66) Kam, W.Y., Morris, A.J., Tham, M.J. and Jones, R.W., "A comparison of several multivariable parameter adaptive controllers", IFAC Conference on Digital Computer Applications to Process Control, Vienna, 1985.

(67) Kam, W.Y., Tham, M.T., Jones, R.W., Warwick. K. and Morris, A.J., "Extended generalised minimum variance multivariable self-tuning control", American Control Conference, Boston, 1985.

(68) Kam, W.Y., Tham, M.T., Morris, A.J. and Warwick. K., "Multivariable self -tuning : Structure, Identification and Control", IFAC Symposium on Identification and System Parameter Estimation, York, 1985.

(69) Kurz, H. and Goedecke, W., "Digital parameter - adaptive control of processes with unknown dead time", Automatical, 1981, **17**(1), pp. 245.

(70) Kiovo, H.N., "A multivariable self - tuning controller", Automatica, 1980, **16**( ), pp. 351.

(71) Luyben, W.L., "Distillation Decoupling", AIChE Journal, 1970, **16**(2), pp. 198.

(72) Carnahan, B., Luther, H.A. and Wilkes, J. O.,"Applied Numerical Methods", John Wiley, New York, 1979.

(73) Luyben, W.L., "Steady state conservation aspectes of distillation column control system design", Industrial and Engineering Chemistry Fundamentals, 1975, **14**(4), pp. 321.

(74) Levy, R.E., Foss, A.S. and Grens, E.A., "Response modes of a binary distillation column", Industrial and Engineering Chemistry Fundamentals, 1969, **8**(4), pp. 765.

(75) Lau, H., Alvarez, J. and Jensen, K.F, "Synthesis of control structures by Singular Value Analysis: Dynamic Measures of Sensitivity and Interaction", AIChE Journal, 1985, **31**(3), pp. 427.

(76) Luyben, W.L., "Feedforward control of distillation columns with intermediate feedback control trays", Chemical Engineering Science, 1965, **24**, pp. 997 - 1007.

(77) Lai, T.L. and Wei, C., "Extended Least Squares and their applications to adaptive control and prediction in linear systems", IEEE Transactions on Automatic Control, 1986, AC - **31**(10), pp. 898.

(78) Latawiec, K. and Chyra, M., "On low frequency and long run effects in self tuning control", Automatica, 1983, **19**(4), pp. 419.

(79) Lozano - Leal, R., "Convergence analysis of recursive identification algorithms with forgetting factor", Automatica, 1983, **19**(1), pp. 95.

(80) Leis, J.R. and Kramer, M.A., "Sensitivity analysis of differential and algebraic equations", Computers and Chemical Engineering, 1985, **19**(1), pp. 93.

(81) Liu, S.L., "Non-interacting process control", Industrial and Engineering Chemistry Process Design and Development, 1967, 6(4), pp. 460.

(82) Marchetti, J.L., Benallou, A., Seborg, D.E. and Mellichamp, D.A., "A pilot-scale distillation facility for digital computer control research", Computers and Chemical Engineering, 1985, **9**(3), pp. 301.

(83) Mijares, G., Holland, C.D, McDaniel, R., Dollar, C.R. and Gallun, S.E, "Analysis and Evaluation of the Relative Gains for Non-Linear systems", Computers and Chemical Engineering, 1985, **9**(1), pp. 61.

(84) McDermott, P.E, Mellichamp, D.A. and Rinker, R.G., "Pole-Placement selftuning control of a fixed - bed autothermal reactor: Single Variable Control", AIChE Journal, 1986, **32**(6), pp. 1004.

(85) Morris, A.J, Nazer, Y., Wood, R.K. and Lieuson, H., "Evaluation of self-tuning controllers for distillation column control", In Iserman, R.

and Kaltenecker, H. (eds), Reprint of IFAC Conference on Digital Computer Applications to Process Control, 1981, pp. 345.

(86) Morris, A.J, Nazer, Y. and Wood, R.K., "Multivariate self-tuning process control", Optimal control Applications and Methods, 1982, **3**, pp.363.

(87) Montague, G.A., Morris, A.J., Wright, A.R., Aynsley, M. and Ward, A., "Parameter adaptive control of the Fed-Batch Penicillin Fermentation", IFAC Conference, B10 '85, Holland.

(88) Montague, G.A., Morris, A.J. and Tham, M.T., "Performance of three multivariable self tuning controller design techniques", 25-th IEEE Conference on Decision and Control, Athens, Greece, Dec 10-12, 1986.

(89) Martin-Sanchez, J.M. and Shah, S.L., "Multivariable adaptive predictive control of a binary distillation column", Automatica, 1984, **20**(5), pp. 607.

(90) Meyer, C., Seborg, D.E. and Wood, R.K, "An experimental application of time delay compensation techniques to distillation column control", Digital Computer Applications to Process Control, IFAC and North Holland Publishing Company, 1977, pp. 439.

(91) Morari, M. and Fung, A.K.W., "Non-linear inferential control", Computers and Chemical Engineering, 1982, **6**(4), pp. 271.

(92) Standart, R., Chemical Engineering Science, 1965, **20**, pp. 611.

(93) Ohmura, S. and Hirata, M., " A new Distillation calculation method for utilising salient features of both short-cut and tray-by-tray methods", Distillation '79 IChemE Symp. Ser., No. 56, 1979, 2.6/51.

(94) Ossman, K.A. and Kamen, E.W," Adaptive regulation of MIMO linear discrete - time systems without requiring persistent excitation", IEEE Transactions on Automatic Control, 1987, AC - **32**(5), pp. 397.

(95) Ogunnaike, B.A. and Ray, W.H., "Computer-aided multivariable control system design for processes with time delays", Computers and Chemical Engineering, 1982, **6**(4), pp. 311.

(96) Ogunnaike, B.A., Lemiare, J.P, Morari, M. and Ray, W.H., "Advanced Multivariable control of a pilot-plant distillation column", AIChE Journal, 1983, **29**(4), pp. 632.

(97) Ogunnaike, B.A. and Ray, W.H., "Incomplete state feedback for time delay systems : Observer applications in multidelay compensators", AIChE Journal, 1984, **30**(5), pp. 717.

(98) Marlin, T.E, McAvoy, T.J., Marino-Galarraga, M. and Kapoor, N., "A short-cut method for process control and operability analysis", Proceedings of the Third International Conference on Chemical Process Control, Asilomar, California, Jan 12-17, 1986, pp. 369.

(99) Payne, S.G., "The application of on-line estimation to a double effect evaporator", Ph.D Thesis, 1974, University of Aston in Birmingham, Gosta Green, Birmingham, U.K.

(100) Shinnar, R., "Impact of model uncertainties and non-linearities on modern controller design : Present structure and future goals", Proceedings of the Third International Conference on Chemical Process Control, Asilomar, California, Jan 12-17, 1986, pp. 53.

(101) Peterka, V., "Predictor-based self tuning control", Automatica, 1984, **20**(1), pp. 39.

(102) Palmor, Z.J. and Powers, D.V., "Improved dead-time compensator controllers", AIChE Journal, 1985, **31**(2), pp. 215.

(103) Palmenberg, R.E. and Ward, T.J., "Derivative decoupling control", Industrial and Engineering Chemistry Process Design and Development, 1976, **15**(1), pp. 41.

(104) Ray, W. H.,"Multivariable Process Control - A Survey", Computers and Chemical Engineering, 1983, **7**(4), pp. 367.

(105) Rosenbrock, H.H., "Distinctive problems in process control", Chemical Engineering Progress, 1962, **58**(9), pp.43.

(106) Rosenbrock, H.H., "The transient behaviour of distillation columns and heat exchangers: A historical and critical review", Transactions of the Institution of Chemical Engineers, 1962, **40**, pp. 376.

(107) Ranzi, E., Faravelli, T., Rovaglio, M., Donenichini, R. and Biardi, G., "A general program for dynamic simulation of multicomponent distillation columns", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. A87.

(108) Rohrs, C.E, Valavani, L., Athans, M. and Stein, G., "Robustness of continuous - time adaptive control algorithms in the presence of unmodelled dynamics", IEEE Transactions on Automatic Control, 1985, AC - **30**(9), pp. 881.

(109) Rosenbrock, H.H., "The control of distillation columns", Transactions of the Institution of Chemical Enginners, 1962, **40**, pp. 35.

(110) Shah, M.K. and Luyben, W.L., " Control of a binary distillation column using non-linear composition estimators", Distillation '79 IChemE Symp. Ser., No 56, 1979, 2.6/1.

(111) Scali, C., Bambrilla, A. and Morari, M., " Controller design for a multicomponent distillation unit in the presence of uncertainty", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. B63.

(112) Stathaki, A., Mellichamp, D.A. and Seborg, D.E., " Dynamic Simulation of a Multicomponent Distillation Column with Assymetreic Dynamics", Canadian Journal of Chemical Engineering, 1985, **63**, pp. 510.

(113) Shimizu, K. and Matsubara, M., " Decoupling and Disturbance rejection control of a binary distillation column with pressure variation", Journal of Chemical Engineering of Japan, 1984, **17**(3), pp. 262.

(114) Shimizu, K. and Matsubara, M.,' Triangular decoupling for a binary distillation column under pressure control", IEEE Transactions on Automatic Control, 1985, AC - **30** (5), pp. 474.

(115) Shaffii, A.F., " The use of microprocessors in the control of chemical plant with special reference to the use of distributed processors", Ph.D Thesis, 1983, University of Aston in Birmingham, Gosta Green, Birmingham, U.K.

(116) Stephanopoulos, G., "Chemical Process Control : an Introduction to Theory and Practice", 1984, Prentice-Hall, New Jersey.

(117) Shimizu, K. and Mah, R.S.H., " Dynamic characteristics of binary SRV distillation systems", Computers and Chemical Engineering, 1983, **7**(2),pp. 103.

(118) Shimizu, K. and Mah, R.S.H., "Pole Assignment in binary SRV distillation", Computers and Chemical Engineering, 1983, **7**(2),pp. 123.

(119) Skogestad, S. and Morari, M., "A systematic approach to distillation column control", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. A71.

(120) Samson, C. and Fuchs, J.J., "Discrete regulation of not - necessarilly minimum phase systems", IEE Proceedings, 1981, **128**(3) Part D, pp. 102.

(121) Sanoff, S.P. and Welstead, P.E., Comments on " Implementation of self tuning regulators with variable forgetting factors", Automatica, 1983, **19**(3), pp. 345.

(122) Samson, C., "Stability Analysis of Adaptively controlled systems subject to bounded disturbances", Automatica, 1983, **19**(1), pp. 87.

(123) Sharaf, M.M and Hogg, B.W., "Evaluation of on-line identification methods for optimal control of a laboratory model turbogenerator", IEE Proceedings, 1981, **128**(2) Part D, pp. 65.

(124) Sen de-la, M., "A model reference adaptive control system for discrete multivariable bilinear systems with interconnected subsystems", IEE Proceedings, 1986, **133**(4) Part D, pp. 165.

(125) Shacham, M., "Comparing software for the solution of systems of non-linear algebraic equations arising in chemical engineering", Computers and Chemical Engineering, 1985, **9**(2), pp. 103.

(127) Schuil, J.A. and Bool, K.K., "Three phase flash distillation column", Computers and Chemical Engineering, 1985, **9**(3), pp. 295-300.

(128) Takamatsu, T., Kinoshita, M., "A simulation procedure for multicomponent distillation columns incorporating plate efficiencies", Journal of Chemical Engineering of Japan, 1985, **18**(1), pp. 78.

(129) Takamatsu, T, and Kawachi, K., "A minimum number of feedback state variables for the decoupling control of a binary distillation column", Journal of Chemical Engineering of Japan, 1978, **11**(3), pp. 179.

(130) Takamatsu, T, Hashimoto, I, and Nakai, Y., "A geometric approach to multivariable control system design of a distillation column", Automatica, 1979, **15**, pp. 387.

(131) Tham, M.T., Morris, A.J. and Montague, G.A., "Multivariable self tuning control for distillation terminal composition regulation", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. A13.

(132) Tham, M.T., Vagi, F., Morris, A.J. and Wood, R.K., "Simultaneous adaptive regulation of terminal compositions in binary distillation", IFAC Conference DYCORD '86 Bournemouth, December 1986.

(133) Tade, M.O., Bayoumi, M.M. and Bacon, D.W., "Adaptive decoupling of a class of multivariable dynamic systems using output feedback", IEE Proceedings, 1986, **133**(6) Part D, pp. 265.

(134) Wahl, E.F and Harriot, P., "Understanding and prediction of the dynamic behaviour of distillation columns", Industrial and Engineering Chemistry Process Design and Development, 1970, **9**(3), pp. 396.

(135) Cairns, B.P. and Furzer, I.A., "Three phase flash distillation Experimental results", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. B505.

(136) Xianya, X. and Evans, R.J., "Adaptive control of discrete-time time varying systems with unknown deterministic disturbances", IEE Proceedings, 1984, **131**(3) Part D, pp. 81.

(137) Yu, C.C, and Luyben, W.L, "Control of multicomponent distillation columns using rigorous composition estimators", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. A29.

(138) Yue, P.L. and Billing, P.J., "Microcomputer simulation of distillation column operation and control", Distillation and Absorption, IChemE Symp. Ser., No 104, 1987, pp. B1.

(139) Zarrop, M.B., "Variable forgetting factors in Parameter estimation", Automatica, 1983, **19**(3), pp. 295.

(140) Seborg, D.E., Edgar, T.F. and Shah, S.L., "Adaptive control strategies for process control - A survey", AIChE Journal, 1986, **32**(6), pp. 881.

(141) Friedland, B. and Grabousky, S.M., "Estimating sudden changes in biases in linear dynamic systems", IEEE Transactions on Automatic Control, 1982, AC-**27**(1), pp. 237.

(142) Shinskey, F.G., "Distillation Control", 1984, 2nd ed., McGraw-Hill, New York.

(143) Hsu, L. and Costa, R., "Bursting phenomena in continuous time adaptive systems with a σ-modification", IEEE Transactions on Automatic Control, 1987, AC-**32**(1), pp. 84.

(144) Ray, W.H., "Advanced Process Control", McGraw-Hill, New York, 1981.

(145) Bozic, S.M., "Digital and Kalman Filtering : An introduction to discrete-time filtering", Edward Arnold, London, 1979.

(146) Ydstie, B.E., Kershenbaum, L.S. and Sargent, R.W.H., "Theory and application of extended horizon self-tuning controller", AIChE. Journal, 1985, **31**(11), pp. 1771.

(147) Lozano-Leal, R. and Goodwin, G.C., "A globally convergent adaptive pole placement algorithm without a persistency of excitation requirement", IEEE Transactions on Automatic Control, 1985, AC-**30**(8), pp. 795.

(148) Benson, R.S., "The state of chemical process control: An industrialist's view", Chemical Engineering Research and Design, 1987, **65**(6), pp. 451.

(149) Russel, L.W. and Perkins, J.D., "Towards a method for diagnosis of controllability and operability problems in chemical plants", Chemical Engineering Research and Design, 1987, **65**(6), pp. 453.

(150) Morari, M., "Robust Process Control", Chemical Engineering Research and Design, 1987, **65**(6), pp. 462.

(151) Ydstie, B.E., "Adaptive Process Control", Chemical Engineering Research and Design, 1987, **65**(6), pp. 486.

(152) Levien, K.L. and Morari, M., "Internal Model Control of coupled distillation columns", AIChE. Journal, 1987, **33**(1), pp. 83.

(153) Skogestad, S. and Morari, M., "Control configuration selection for distillation columns", AIChE. Journal, 1987, **33**(10), pp. 1620.

(154) Astrom, K.J. and Wittenmark, B., "On self-tuning regulators", Automatica, 1973, **9**, pp. 185.

(155) Peterka, V., "A square root filter for real-time multivariable regression", Kybernetika, 1975, **11**, pp. 53-57.

(156) Clarke, D.W. and Gawthrop, P.J., "Self-Tuning Controller", Proceddings of IEE, 1975, **122**(9), pp. 929.

(157) Clarke, D.W. and Gawthrop, P.J., "Self-Tuning Controller", Proceedings of IEE, 1979, **126**(6), pp. 633.

(158) Kershenbaum, L.S. and Fortescue, T.R., "Implementation of on-line control in chemical process plants", Automitca, 1981, **17**(7), pp. 777.

(159) Morari, M., "A general framework for the assessment of dynamic resilience", Chemical Engineering Science, 1983, **38**, pp. 1881.

(160) Klema, V.C. and Laub, A.J., "The singular value decomposition : Its computation and some applications", IEEE Transactions on Automatic Control, 1980, AC-**25**, pp. 164.

(161) Shinskey, F.G., "Distillation Control", 1977, McGraw-Hill, New York.

(162) Dahlqvist, S.A., "Control of the top and bottom products compositions in a pilot sclae distillation column", Distillation '79, IChemE Symp. Ser., 1979, No 56, pp. 2.6/25.

(163) Kalman, R.E., "Topics in mathematical systems theory", McGraw-Hill, London, 1969.

(164) AIChE, "Bubble Tray Design Manual : Prediction of Fractionation Efficiency", 1958, Ammerican Institution of Chemical Engineers, New York.

(165) Arkin, Y., "Dynamic process operability : Important problems, recent results and new challenges", Proceedings of the Third International Conference on Chemical Process Control, Asilomar, California, Jan 12 - 17, 1986, pp. 323.

(166) Astrom, K.J., "Adaptation, auto-tuning and smart controls", Proceedings of the Third International Conference on Chemical Process Control, Asilomar, California, Jan 12-17, 1986, pp. 427.

(167) Shinskey, F.G., "An expert system for the design of distillation controls", Proceedings of the Third International Conference on Chemical Process Control, Asilomar, California, Jan 12-17, 1986, pp. 895.

(168) Desphande, P. B., "Distillation Dynamics and Control", Edward Arnold, London, 1985.

(169) Rys, R.A., "Advanced control techniques for distillation columns", Chemical Engineering, December 1984, **10**, pp. 75.

(170) Rinard, I.H., "A road map to control system design", Chemical Engineering, November 1982, **29**, pp. 47.

(171) Pollard, A., "Process control", Heinemann, 1981.

(172) Smith, O.J.M., "Closer control of loops with dead-time", Chemical Engineering Progress, 1957, **53**(5), pp. 217.

(173) Smith, O.J.M., "A controller to overcome dead-time", ISA Journal, 1959, **6**(2), pp. 28.

(174) Palmor, Z.J., "Stability of Smith dead-time compensator controllers", Int. Journal of Control, 1980, **32**(6), pp. 937.

(175) Sin, K.S. and Goodwin, G.C.,"Stochastic adaptive control using a modified least squares algorithm", Automatica, 1982, **18**(3), pp. 315-321.

(176) Dumont, G.A., "On the use of adaptive control in the process industries", Proceedings of the Third International Conference on Chemical Process Control, Asilomar, California, Jan 12-17, 1986, pp. 467.

(177) Doyle, J.C. and Morari, M., "A unifying framework for control systems design under uncertainty and its implications for chemical process control", Proceedings of the Third International Conference on Chemical Process Control, Asilomar, California, Jan 12-17, 1986, pp. 5.

(178) Wonham, W.M. and Morse, A.S., "Decoupling and pole assignmenet in linear multivariable systems: A geometric approach.", SIAM Journal of Control, 1970, **8**, pp. 1.

(179) Falb, P.L. and Wolovich, W.L., "Decoupling in the design and Synthesis of multivariable control systems", IEEE Transactions on Automatic Control, 1967, AC-**12**(6), pp. 651.

(180) Bachman, K.C., Zimmerli, A. and Simons, E.L., "Vapour Liquid Equilibria at sub-atmospheric pressures", Industrial and Engineering Chemistry, 1950, **42**(12), pp. 2569.

(181) Ogbonda, J.E., "Dynamic Simulation of Chemical Plant", Ph.D Thesis, 1987, University of Aston in Birmingham, Gosta Green, Birmingham , U.K.

(182) Fuller, M., "The control of a distillation column using the SEED 19 microcomputer", 1st Year Report, 1984, University of Aston in Birmingham, Gosta Green, Birmingham , U.K.

(183) Kocak, M.C, " Dynamic Simulation of Chemical Plant", Ph.D Thesis, 1980, University of Aston in Birmingham, Gosta Green, Birmingham, U.K.

(184) Kreisselmeier, G., "An approach to stable indirect adaptive control", Automatica, 1985, **21**, pp. 425-431.

(185) Jordan, A.P.H., "The optimal control of production quality switching in distillation column", Ph.D Thesis, 1972, University of Manchester, Manchester, U.K.

(186) Gunderlach, V.G., "Trends in computer-aided on-line measurement methods", Computers and Chemical Engineering, 1984, **8**(5), pp. 225.

(187) Gerald, C.F., "Applied Numerical Analysis", Addison-Wesley, 1977.

(188) Coughanowr, D.R. and Koppel, L.B., "Process systems analysis and control", McGraw-Hill, 1965.

(189) Shimizu, K., Holt, B.R., Morari, M. and Mah, R.S.H., "Assessment of control structures for binary distillation columns with secondary reflux and vapourisation", Industrial and Engineering Chemistry Process Design and Development, 1985, **24**(3), pp. 852.

(190) Dumont, G.A., "Self tuning control of a chip refiner motor load", Automatica, 1982, **18**(3), pp. 307.

(191) Ziegler, J.G. and Nichols, N.R., Optimum settings for automatic controllers", Transactions of ASME., 1942, **64**, pp. 759.

(192) Thurston, C.W., "Computer aided design of distillation column controls", Hydrocarbon Processing, July 1982, pp. 125.

(193) Nawari, M.O, "Mathematical modelling and Optimal control of chemical processes", Ph.D Thesis, 1984, University of Aston in Birmingham, Gosta Green, Birmingham , U.K.

(194) Witcher, M.F. and McAvoy, T.J., "Interacting control systems : steady state and dynamic measurement of interaction", ISA Transactions, 1977, **16**(3), pp. 35.

(195) Crossley, T.R. and Porter, B., "Synthesis of aircraft modal control systems having real of complex eigenvalues", Aeronautical Journal, 1969, **73**, pp. 138.

(196) Morgan, B.S., "The synthesis of linear multivariable systems by state - variable feedback", IEEE Transactions on Automaticc Control, 1964, AC - **9**, pp. 405.

(197) Landau, I.D., "A survey of model reference adaptive techniques : Theory and Applications", Automatica, 1974, **10**, pp.353.

(198) Box, G.E. and MacGregor, J.F., "The analysis of closed loop dynamic-stochastic systems", Technometrics, 1974, **16**, pp. 391.

(199) Stanley, G.T. and McAvoy, T.J., "Dynamic energy conservation aspects of distillation control", Industrial and Engineering Chemistry Fundamentals, 1985, **24**(4), pp.439.

(200) Garcia, C.E and Morari, M., "Internal Model Control. 1. A unifying review and some new results.", Industrial and Engineering Chemistry Process Design and Development, 1982, **21**(2), pp. 308.

(201) Garcia, C.E and Morari, M., "Internal Model Control. 2. Design procedure for multivariable systems", Industrial and Engineering Chemistry Process Design and Development, 1985, **24**(2), pp. 472.

(202) Garcia, C.E and Morari, M., "Internal Model Control. 3. Multivariable control law computation and tuning guidelines", Industrial and Engineering Chemistry Process Design and Development, 1985, **24**(2), pp. 484.

(203) Cohen, G.H. and Coon, G.A., "Theoretical investigations of retarded control", Transactions of ASME., 1969, **75**, pp. 827.

(204) Grosdidier, P. and Morari, M. "Closed loop properties from steady state gain information", Industrial and Engineering Chemistry Process Design and Development, 1985, **24**(2), pp. 221.

(205) Ryskamp, C.J., "New strategy improves dual composition control", Hydrocarbon Processing, 1980, **59**(6), pp. 51.

(206) Astrom, K.J. and Wittenmark, B., "Computer controlled systems: Theory and design", Prentice-Hall, New Jersey, 1984.

(207) Shah, S.L., "Comment on 'Disturbance rejection in linear systems'", International Journal of Systems Science, 1976, **7**(4), pp. 479.

(208) Preuss, H. P., "Perfect steady-state tracking and disturbance rejection by constant state feedback", International Journal of Control, 1982, **35**(4), pp.75.

(209) Power, H.M., "Simplification and extension of the Falb-Wolovich decoupling theory", Intenational Journal of Control, 1977, **25**(5), pp.805.

(210) Kimura, H., "Perfect and sub-perfect regulation in linear multivariable systems", Automatica, 1982, **18**(2), pp.125.

(211) Anderson B.D.O and Moore, J.B., "Linear Optimal Control", Prentice-Hall, 1971.

(212) Tham, M.T, Morris, A.J and Montague, G.A., "Generalisation of GMV selftuning controller sysnthesis", 1988, IFAC Identification '88, Beijing, August 27-31.

(213) Tham, M.T, Morris, A.J and Montague, G.A., "Self-tuning Process Control: a review of som algorithms and their application", 1987, American Control Conference, Minneapolis, USA.

List of Symbols

a	scalar, leakage term in Equation 2.93
A,B,C,D	constant system, input, measurement and disturbance matrices
A*, B*	matrices Equations 6.7 and 6.8
$A(z^{-1})$, $B(z^{-1})$, $C(z^{-1})$	polynomials of an SISO system described in discrete form (Equation 8.1)
$\mathbf{A}(z^{-1})$, $\mathbf{B}(z^{-1})$, $\mathbf{C}(z^{-1})$	polynomials matrices of an MIMO system described in discrete form (Equation 8.3)
$E(z^{-1})$, $F(z^{-1})$, $G(z^{-1})$	polynomial of a single loop self tuning control law, Equations 8.12 and 8.13
$\mathbf{E}(z^{-1})$, $\mathbf{F}(z^{-1})$, $\mathbf{G}(z^{-1})$	polynomial of a multivariable self tuning control law, Equation 8.25
d_i	decoupling index for the i-th "control loop", Equation 6.6
d, **d**	scalar and vector of constant offset term in Equation 8.1 and 8.3, respectively
ep(t)	prediction error at time t
ε	
e	scaler, deadzone in Equation 2.93
Eo	a measure of information content in the recursive parameter estimator, Equations 8.42 and 8.43
E	expectation operator
F	Feed flow
F	constant state feedback in Equation 6.1
G	precompensator in Equation 6.1
Gp	process model
Gc	controller
h	liquid enthalpy
h(**x**)	non-linear measurement matrix
H	matrix containing number of states to be measured
I	Identity matrix
J	Jacobian matrix
$\underline{K}$(t), $\underline{P}$(t), $\underline{Q}$(t),$\underline{R}$(t),$\underline{M}$(t)	the gain, covariance, system noise covariance, measurement noise covariance, and linearised measurement marices for the Kalman filter
K(t), PP(t)	the gain and covariance matrice of RLS estimator
Kp	process gain
Kc	proportional gain of a PI controller
K*	diagonal matrix in definition of precompensator **G**,
L_{od}	observability matrix for the Kalman filter, Equation 2.129

M_k	diagonal matrix containing pole assignments for the decoupling and disturbance rejection controller
No	nominal asymtotic memory length in Equation 8.47
$P(z^{-1})$, $Q(z^{-1})$, $R(z^{-1})$	output weighting, control weighting and setpoint filter polynomials for an SISO self tuning controller
$\mathbf{P}(z^{-1})$, $\mathbf{Q}(z^{-1})$, $\mathbf{R}(z^{-1})$	output weighting, control weighting and setpoint filter matrix polynomials for an MIMO self tuning controller
Q	diagonal matrix of singular value
$\mathbf{R}$	state feedback in Equation 6.39
T_j	temperature of the j-th tray
$\mathbf{T_f}$	feedforward compensator
$\mathbf{T_I}$, $\mathbf{T_D}$	diagonal matrices containing integral and derivative times
u, w, y	scalar input, setpoint and output
u, **w**, **x**, **y**, **zd**	input, setpoint, state, output and disturbance vectors
v_i, w_i	i-th left and right eigenvectors
V, W	matrix of left and right eigenvectors
V, *W*	matrices of the left and right singular vectors respectively
W_c, W_o	controllability and observability matrices in Equations 2.27 and 2.28
X	state vector of kalman filter
Y	
z	discrete polynomial, z-transform operator
nm, m, n, nd	number of outputs, inputs, states and suturbances
Lr, Qrb, xf	reflux, reboiler heat input and feed composition
X*	prediction of **X**
X^	estimate of **X**
x*	prediction of **x**
x^	estimate of **x**
Ø(t)	data vector of RLS
θ(t)	parameter vector of RLS
μ	scalar defining rate of correction in the SPC algorithm, Equation 8.54
α	scalar number in Equation 8.49
α	scalar defining rate of correction in the PC algorithm, Equation 8.52
β	
ψ	auxillary output
λ_i	i-th eignenvalue
σ	singular value
π_{ij}	relative gain between i-th output and j-th input

η_{ij}	activation of a mode i by input j
ρ	damping factor in Equation 5.19
τ_i, τ_p, τ_D	integral time, time constant and derivative time
ϖ	scaling factor in the "proxy of residuals" method, Equation 2.76
ζ	poles
$\xi(t)$	noise affecting an SISO system
χ	
$\gamma(.)$	condition number
κ_i	ith diagonal element in a modal controller K, in Equation 2.35
$\upsilon(t)$	forgetting factor at time t
ς^i_k	k-th pole for i-th output response
Θ	
Γ, Ω	right and left eigenvectors of system matrix A
Φ	transition matrix
Δ	deviation from steady state
Δ_1	difference operatoer, $1- z^{-1}$
Δ_k	k difference operatoer, $1- z^{-k}$
$\|\Lambda\|$	norm of RGA matrix
Λ	diagonal matrix of eigenvalues

APPENDIX A1

Flowmeter and control valve specifications; Functions of the Monolog software, Master.

A1.1 Flowmeter and control valve specifications

A1.1.1 Flowmeter specification

Type	**Standard**	**High Flow**
Electrical Specifications		
Power Supply	4.5 - 24 volts (dc)	
Current Consumption	20 milliamps (max)	
Output	Square wave pulse from open collector, TTL,CMOS and LSI compatible	
Output High	supply (4.5 - 24 V (dc))	
Output Low	100mV Type	
Frequency	24Hz at 10 l/hr	31Hz at 30l/hr
	52Hz at 20 l/hr	375Hz at 300 l/hr
Accuracy		
Signal Reproductivity	±1%	
Sensor to sensor variations	±3%	
Linearity	±3%	

Type	Standard	High Flow
Mechanical Specification		
Weight	70grammes	
Construction	Acetal body and rotor, 316 stainless steel shaft ceramic magnets.	
Metering principle	velocity counter	
Flow range	3.0 - 100 l/hr	10 - 500 l/hr
Temperature range	-25°C to +120°C	
Pressure drop	0 - 1 Bar	
Viscosity range	0.8 - 10 cSt	
Bursting pressure	30 Bar	
Operating pressure	10 Bar	
Flow direction	Both	One way as indicated

A1.1.2 Control Valve Specifications

Valve : Miniature Air-Operated Control Valve,

'M' Valve for liquid Gases abd Steam in 1/4" and 1/2" pipes.

Standard Specification	
Body	Wrought 316 Stainless Steel
Air Motor Housing	Die-cast aluminium, Epoxy coated finish
Bellofram	Nitrile
Packing	PTFE Braided
Connections	NPT Screwed
Stem/Plug/Seat	All 316 Stainless steel
MVO	Closes with lack of air (Used in this work**)
MVC	Opens with lack of air
Working limits	350 bar without shock (at 20 °C); 200 °C
Characteristics	Linear with rangeability of 40:1
Pressure range	3 - 15 psig (o.2 - 1bar) for operation
Connection	1/4 NPT; J trim, Cv max = 0.05, Kv max 0.04 ***

The current to pneumatic converter specification	
Type 100D microprocessor compatible digital to pneumatic converter.	
Pressure range	up to 15 psig.
Digital input	8 or 10 bit binary number, active
Curent input	4-20 mA
Output	3 - 15 psig

A1.2 Functions of the Monolog

The Monolog is a front end processors which recieves commands from the host computer, executes the command and reports results if appropriate. The description of the Monolog has been given in Chapter 4 Section 4.5.

The Monolog **commands** are:

1. Read and report analogue channels once
2. Read and report analogue channels continuously
3. Halt reading analogue channels
4. Read digital port
5. Set digital port
6. Set analogue output
7. Set floating point data format
8. Set alarm reporting on digital input ports.

Command 1. Read and report analogue channels once: The analogue readings are performed with a 16 bit precision A/D conversion. Monolog adopts default values of parameters are not specified. The full command string to commence analogue readings is are 1, F, L, M, R, S, C, D, W. This command string may be truncated at any parameter and values not set assume default values.

The parameters that can be set are as follows;

	Range	Default
F: First channel number	1 - 256	0
L: Last channel number	1 - 256	0
M: Mode of operation	0-14	0
R: Range of readings	0 - 5	0
S: Speed of operation	0 - 1	0
C: Cycles of readings	1- 32765	1
D: Delay between readings	0 - 255	0
W: Wait for synchronisation	0 - 2	0

The F and L channels specify which channels are to be read sequentially. If L<F the tha channels will be scanned from F to 256, then from 1 to L.

The mode, M, values of 0 - 14 have the following functions

Mode	Result in
0 Skip channels (returns code $2 \times 10^{+10}$)	
1 DC voltage measurement, 0 - 10 volts	microvolts
2 Thermocouple type K	°C
3 Thermocouple type J	°C
4 Thermocouple type T	°C
5 Thermocouple type S	°C
6 Thermocouple type E	°C
7 PRT type PT100	°C
8 Cold junction temperature	°C
9 Resistance	milliohms
10 Strain gauge full bridge	microstrain
11 4 - 20 mA transducer	%
12 DC current	microamps
13 Voltage measurement	LSBs
14 DC voltage measurement	millivolts
The units given above are the engineering units that can be handled by Monolog	

The range parameter, R, applies to the range of voltage measured. The ranges are

0 Autorange
1 Fix last used range
2 20mV
3 150mV
4 1.5V
5 12V

The speed parameter,S, denotes the speed of scanning the channels

0 16 channels / second - 16 bit resolution

1 160 channels persecond - 12 bit resolution

A cycle count, C, from 1 to 32765 is permitted. The scan from F to L channels is repeated "C" times

The delay, D, allows reading rate to be controlled over a range by inserting up to 255 fixed increments of delay between each analogue to digital conversion cycle. The delay for each increment is given on the table below

Resolution(bits)	Delay increment (ms
16	4.660
12	1.379

The wait facility, W, allows the Monolog to remain quiescent until a measurement cycle is initiated by a signal at TPI on the A/D converter board. The modes are

0 proceed with no delay between scans

1 Commence scan on positive transition at TPI

2 Read next channel on a positive transition at TPI

Command 2. Read and report analogue channels continuously: The function of command 2 is to initiate an analogue scan sequence, report the values measured and then repeat. The full command string is 2, F, L, M, R, S, C, D, W, which are exactly the same as with command 1 except the 2. This command string, or a truncated form, is used for on-line applications on the column.

Command 3. This terminates the read and report sequence of command 2. The format is 3; with no parameters.

Command 4. Read digital port : This command reads one * bit digital port

Format : 4, n

where n is the port number in the range 2 - 15. Port one is on the Single Board Computer in the Monolog and ports 2 to 15 are on General Purpose Interface (GPIF) parallel interface boards.

Command 5. Set Digital Port : This command sets the output value of one 8 bit digital port.

Format : 5, n, v

where n is the port number range 1 to 15 and v is the value range 0 to 255.

Command 6. Set analogue output : This sets the output of a 12 bit digital to analogue (D/A) converter.

Format : 6, n, v

where n is the D/A converter port number range 1 to 4 and v is the value in the range 1 to 1023. There are 4 ports on each D/A converter card

Command 7. Set floating point data format : This sets the resolution of the floating point data format.

Format : 7, n

where n is the number of digits after the decimal point and ranges from 0 to 7. The default is 2 and this was used in this work.

Command 8. Set a port for alarm reporting :

Formant : 8, n, v

where n is the port number range 1 to 15 and v is the value of the port bits that require alarm reporting; range 0 - 255.

A1.2.1.How to use Master

The **Master** is a subroutine written in 6809 assembler language and can be called from any user written Basic09 program. The calling format is

RUN Master (function, status,device,path,errors,buffer)

The function is the parameter corresponding to what the Monolog is required to do; status is the state of the Monolog hardware and device is the device descriptor; path is the input and output path through which data is retrieved and sent; errors report the errors encountered during operation; and buffer essentially holds the data acquired from the experimental column. Since the Monolog itself does not interact with the user, status and errors are the means by which an indication of the state of the hardware at any time can be known.

The meanings of the parameters passed to the **Master** are as follows :

function : integer variable

Specifies the operation to be performed on the Monolog link

The possible operations are as follows;

Value of **function**	**Operation**
1	Open and initialise path number for link to Monolog
2	Close link to Monolog
3	Reset Monolog to a known state
4	Send a command
5	Send data
6	Read Data

status : integer variable

Gives the last known state of the Monolog link after each request to **Run Master**. **Status** is set to zero at start of program execution and must not be modified by the user during execution. The possible values of this parameter are as follows;

Value of **status**	**Meaning**
0	Link to Monolog closed or not in use
1	Link error has occurred
2	Link open and initialised
3	Monolog reset to known state
4	Command sent to Monolog
5	Data sent to Monolog
6	Data read from Monolog

device : integer variable

Used internally by **Master** program to hold device data. It is set to zero at start of execution and must not be modified during execution of program.

path : integer variable

Used by **Master** program to hold the path number allocated for the link opened to the Monolog. It is set to 0 at start of program and must not be modified during execution.

errors : Integer variable

This parameter is set whenever the **Slave**, **Master** or 0S9 detects an error during communication. **Status** is also set to 1 whenever an error is detected.

Possible errors values in addition to the standard OS9 errors are

Values of errors	Meaning
0	No errors
2	illegal function
3	wrong number of parameters
4	illegal command
5	input time out
6	no acknowledgement recieved
7	buffer overflow on input
8	no STX recieved
9	data overrun on input
10	command rejected
11	wrong parameter type or size

The input and output timeouts provided by the Monolog routines are limited in duration to a maximum of 1.5 seconds. Therefore, if the time taken for the readings exceeds this timeout it is neccessary to put a retry loop in the program that retrieves data.

buffer : string variable

This is a variable length string buffer whose use and contents depends on the **function** requested. The possible contents for each function are:

Function	Contents of **buffer**
Open Link	Contains text strings equivalent to the device name i.e: **buffer** := "/M0"
Reset Link	Set to spaces i.e: **buffer** :=" ".
Send command	Contains the text string representing the **command** to be sent to the Monolog. Each field separated by a single space character and terminated by a carriage return character i.e : **buffer** := STR£(command) + " " + STR£(F) + " " + STR£(L) + " " + STR£(M) + " " + STR£(R) + " " + STR£(S) + " " CHR£(13) Minimum size of buffer = 2 bytes Maximum size of buffer = 32 bytes
Send data	Not yet available
Read data	Contains the data recieved in text string form with each value seperated by a single space and terminated by a carriage return. The data recieved is dependent upon the last **command** sent.
Close link	Set to spaces i.e : **buffer** = " "

More details on the Monolog can be found in the Monolog user manual (40)

A1.2.3 Functions of the programs for on-line data logging and control

Module	Function
Get-Data	Retrieves process measurements from the experimental rig through the Monolog using **Master**.
Log-New	Main program for data-logging and storage. Calls **Get-data**
Drive-Valve	Uses user information to send control actions to the column
Valve-out	Implements the control actions in **Drive-Valve**

A1.3 The startup and shut down procedures of the column

A1.3.1 Process Startup

1) Switch on mains power and power supply to the computer, Monolog, and the instrumentation of the column. Open air supply from the compressed air in the department and ensure air supply to the control valves is 20 psig by adjusting the pressure regulator.

2) Open cooling water supply for the condenser and the heat exchanger for the cooling of the bottoms.

3) Fill up a feed tank with T/T liquid mixture of required composition and ensure the mixture is well mixed in order that constant feed composition is maintained during operation, as composition analysers were not fitted on the column. Mixture between 30/70 %w/w to 50/50 %w/w were usually used.

4) Load all the necessary data acquisition programs, including user written real-time programs like Log-New-store for storing step response data. Ensure enough

memory space in the floppy disk is available for data storage. A typical set of data stored at each sampling interval is shown on Table 4.5. The measurement sampling interval used was always 30 seconds

5) Run the program **Log-new** which displays information on the VDU

6)Fill the reboiler drum with feed to required level. Switch on the firerod heater to required setting. Heat input of between 0.7 and 1.25 KW was usually used. As vapour is produced the reboiler liquid level readings will initially rise due to the effect of increased pressure at the base of the column from vapour entry. Experience showed the the extra height indicated within the range is around 8cm ± 2cm, and this is subtracted from the readings supplied to the computer. The column is charged with more feed as more liquid is vaporized to ensure the liquid level stays within range.

7) When enough liquid has been condensed in the reflux drum, the column is then operated at total reflux for about 10 to 15 minutes. The feed valve is then opened to deliver the required feed flow into the column, the distillate valve is opened and usually set at 40% opening, and two position control of the reboiler liquid level is introduced (high opening 60% low opening 10 %).

8) After steady state, or near steady, is reached, usually around 20 minutes, then step response tests or control studies can commence.

A1.3.2 Process Shutdown

1) Switch of firerod heater and then all the delivery pumps

2) Switch off air supply to the valves

3) Switch of Monolog and The computer

4) Switch off cooling water supply after 30 minutes

5) Allow the contents of the reboiler to cool down to around 30 °C, usually about 3 hours.

6) Open air supply to the valves and pump out the liquid to the product or storage tank.

7) Switch off are supply and the pump

APPENDIX A2

Functions of the program modules of the steady state and the dynamic model of the distillation column

Appendix A2.1 Software for the steady state simulation of the binary Trichloroetylene and Tetrachloroethylene distillation system using the method of Kinoshita et al.(45)

Key Modules	Functions
normalise_XY	Normalises the liquid or vapour compositions
vl-enth	Computes the vapour and liquid enthalpies
bubpt	performs the bubble point calculations to determine equilibrium temperature
gauss-emcp	performs Gaussian elemination withmaximal pivoting algorithm given in Burden et. al. (56) to solve $- \mathbf{J}\, \delta \mathbf{x}^{r} = \mathbf{f}^{r}$ for $\delta \mathbf{x}$
model_nr_xallemv	key module that formulates the equations for the steady state simulations
Simulate_nr_xallemv	main calling program; accepts initial values, computes the jacobian matrix, finds the damping coefficient and outputs information to the VDU and files.

Appendix A2.2 Software for Dynamic Simulation of the distillation column

Key Modules	Functions
Dynmodel	performs the dynamic simulation
vapour	calculates vapour and liquid enthalpies, and the vapour and liquid flow inside the column
bubpt	performs bubble point calculations to obtain equilibrium tray temperatures on a tray
input-data	obtains input data from a file.
lamda	computes latent heat of vapourisation of binary mixture
dens	computes density of mixture
enth	computes enthalpy of pure liquid.

The computer programs for both steady state and dynamic models were written using mass fractions of the vapour and liquid compositions rather their mole fractions. The equations as presented above are presented in form of mole fraction to avoid confusion. The necessary conversions from mass to mole fractions and vice versa were done in the computer programs.

A2.2.1 Settings for PI and PID controllers using the Cohen and Coon equations (Stephanopoulos (116))

The setting are based on the assumption that the process is first order with a dead time g in Laplace domain, as

$G(s) = Ke^{-\tau_d s}/(\tau_p s+1)$

K process gain, τ_p is process time constant and τ_d is the process time delay

PI

$Kc = (0.9 + \tau_d/(12\tau_p))\tau_p/(K\tau_d)$

$\tau_i = \tau_d(30.0 + 3\tau_d/\tau_p)/(9 + 20\tau_d/\tau_p)$

PID

$Kc = (4/3 + \tau_d/(4\tau_p))\tau_p/(K\tau_d)$

$\tau_i = \tau_d(32.0 + 6\tau_d/\tau_p)/(13 + 8\tau_d/\tau_p)$

$\tau_D = 4\,\tau_d\,/(11 + 2\tau_d/\tau_p)$

Kc is the controller gain, τ_i is the integral time and τ_D is the derivative time

The PID controller in contrinuous time form is

$$u = Kc\left(e(t) + \frac{1}{\tau_i}\int_0^t e(t)dt + \tau_D\frac{de(t)}{dt} \right)$$

where $e = y_s - y$. The y is the controlled output and the s denotes setpoint.

To implement the controller by computer control the control law above must be represent discrete form of in form of difference equation. First order differencing is assumed in this v In this case the PI control law is

$$u(k) = Kc\left(e(k) + \frac{\Delta Tc}{\tau_i}\sum_{i=0}^{i=k} e(i) + \tau_D\frac{e(k)-e(k-1)}{\Delta Tc} \right)$$

where k is the sampling instant. This equation is the positional form of the PI as

$$u = u_m - u_0$$

where the subscripts m and 0 denote measured and initial values. The velocity algorithm of the discrete PID controller is obtained by subtraction of the control law at k-1 from the law at k. This gives

$$\Delta_1 u(k) = Kc\left(e(k) - e(k-1) + \frac{\Delta Tc}{\tau_i}e(k) + \tau_D\frac{e(k) - 2e(k-1) + e(k-2)}{\Delta Tc} \right)$$

where $\Delta_1 u(k) = u(k) - u(k-1)$

A2.3 Matrix manipulation modules in Basic09

Module	Function
invec(r,**x**)	initialise **x** entries with zeroes
scavec(r,sca,**x**,**y**)	multiplies **x** by scalar sca, **y** = sca.**x**
veceq(r,**x**,**y**)	equates vectors, **y** = **x**
veclen(r,**x**,val)	returns val which is 2-norm of **x**
vadsub(r,iflag,**x**,**y**,**xy**)	adds and subtracts vectors; **xy** = **x** + **y** if iflag =1; **xy** = **x** - **y** if iflag =-1.
max_elm(r,**x**,val)	returns maximum element in **x**, val = max(x(i)) $_{i = 1 \text{ to } r}$
max_elms(r,**x**,**y**,**val**)	returns **val** which is a vector with the maximum elements of corresponding **x** and **y** entries; **val**(i) = max(x(i),y(i)) $_{i = 1 \text{ to } r}$.
vectvec(r,**x**,**y**,val)	returns val = $\mathbf{x}^T\mathbf{y}$
vecvect(r,c,**x**,**y**,**a**)	returns r by c matrix **a**, **a** = $\mathbf{x}.\mathbf{y}^T$
inmat(r,c,**a**)	initialises matrix **a** with zero entries
iden(**id**)	returns identity matrix, **id**.
scamat(r,c,sca,**a**,**b**)	multiplies matrix **a** by a scalar sca, **b** = sca.**a**
mateq(r,c,**a**,**b**)	equates matrices, **b** = **a**
matps(r,c,**a**,**at**)	transposes **a** to give **at**, **at** is c by r matrix
madsub(r,c,iflag,**a**,**b**,**c**)	adds and subtracts matrices; **c** = **a** + **b** if iflag =1; **c** = **a** - **b** if iflag=-1
matmul(r,c,ip,**a**,**b**,**c**)	product of two matrices, **c** = **a**.**b** where **a** is r by c, **b** is c by r, and **c** is r by ip.
matvtrs(r,c,**x**,**a**,**c**)	premultiplies matrix **a** by vector $\mathbf{x}^T$, c = $\mathbf{x}^T.\mathbf{a}$
matmod_inf(r,c,**a**,val)	returns val = infinite norm of matrix **a**.
matmod_2(r,c,**a**,val)	returns val = 2-norm of matrix **a**.
diaadd(r,**a**,**b**,**c**)	adds diagonal element of **a** to diagonal elements of **b** to give new **b** returned as **c**.
mat_multiple(r,c,n,**a**,**c**)	computes $\mathbf{a}^n$; $\mathbf{c} = \mathbf{a}^n$.
matinv_par(**a**,r,path)	matrix inversion of a square matrix **a**, an r by r matrix; returns inverse as **a**. Method used is the maximal pivotal strategy given in Carnahan et al. (72)

Key : r, c, ip denote row, column and column of a matrix. Bold letters represent vectors and matrices.

APPENDIX A3

Appendix A3.1 Software for the synthesis and implementation of the Decoupling and Disturbance Rejection Control scheme

Program Module	Function
dec-index	Computes the decoupling indices d_i
ab-star	Computes A* and B*
DEC-CSVFBPMS	Accepts the pole assignments. Computes the constant gain feedback matrix, **F**, and the precompensator, **G**. Computes **H** which indicates the minimum number of state variables and the state variables themselves using the procedure of Takamatsu and Kawachi (129).
Collect-matr	Computes the feed forward compensator $\mathbf{T_f}$ from the relationship $\mathbf{T_f} = -(B^T B)^{-1} B^T D$
Calc-Kstar	Computes the matrix **K*** for use in setpoint tracking from using the relationship $C(-(A + BF))^{-1} B(B^*)^{-1} \mathbf{K}^* = I$.
DCOL-ABDEC	Sumulates the dynamic behaviour of the distillation column described by dx/dt = A**x** + B**u** + D**zd** and implements control action **u** = **Fx** + **Gw** at specified frequency.
SIMUL-ABDDEC	Main calling program. It calls **DCOL-ABDDEC** and introduces the sequence of disturbances specified by the user.

** Note: In the synthesis package for the Decoupling and Disturbance Rejection controller, the matrix inversions were done using the maximal pivotal strategy given in Carnahan et al.(72).

A3.2 On the formulation of Equation 2.18 in Chapter 2.

The relationship in Equation 2.18 in Chapter 2 is not presented as was presented in Shimizu et al. (189). From Equatiuon 2.16 the control input for an IMC controller is

$$\mathbf{u} = Gc(\mathbf{y}_S - \mathbf{d})$$

with $Gc = Gp^{-1}$. The $Gp = VQW^T$ from SVD analysis in Equation 2.10; the V and W are unitary matrices so that $VV^T = I$ and $WW^T = I$, where I is the identity matrix.

According to Shimizu et al., substituting Gp into the control law gives

$$\mathbf{u} = VQ^{-1}W^T(\mathbf{y}_S - \mathbf{d}).$$

This representation is wrong since must $VQ^{-1}W^TVQW^T \neq I$. The correct representation is

$$\mathbf{u} = WQ^{-1}V^T(\mathbf{y}_S - \mathbf{d})$$

as was presented in this thesis, Equation 2.18 in Chapter 2. This is because

$$WQ^{-1}V^TVQW^T = I.$$

APPENDIX A4

A4.1 Modules that perform the Kalman filtering

Module	Function
trans_matrix	forms the transition matrix
mes_matrix	forms the linearised measurement matrix
temp_deriv	form the derivative of temperature with respect to composition, for a tray; it computes $\frac{dT_j}{dx_j}$
pkk_1	Computes error covariance matrix $\underline{P}(k+1,k+1)$
pkk_2	Computes error covariance matrix $\underline{P}(k+1,k)$
kalman	implements the Kalman filtering algorithm; each call to **kalman** performs a filter cycle.
main_kalman	main calling program; initialises arrays and vectors, gets initial state estimates and covariance matrices and calls **kalman** for each filter cycle

APPENDIX A5

A5.1 The Square Root algorithm for updating the covariance matrix (Kiovo (70))

$$P(t+1) = S(t+1)S^T(t+1)$$

where

$$S(t+1) = \sigma_{j-1}\left(S(t)_{ij} - f_i s_i^{(j-1)}/\sigma^2_{j-1}\right)/\left(\sigma_j \upsilon^{(1/2)}\right), \qquad i,j = 1, \ldots\ldots p$$

$$\sigma_0 = \upsilon^{(1/2)}$$

$$\sigma_j = (\sigma^2_{j-1} + f^2_j)^{(1/2)}, \qquad j = 1, \ldots\ldots. p$$

$$f_j = \sum_{i=1}^{j} S(t)_{ij} \text{Ø}(t-k)_i, \qquad j = 1, \ldots\ldots. p$$

$$s_i^{(q)} = \begin{cases} \sum_{z=i}^{q} S(t)_{iq} f_q, & i = 1, \ldots\ldots. q \\ 0, & \text{if } i > 1 \end{cases}$$

$( )_{ij}$ denotes the ij-th element of the matrix, p is the dimension of the data vector, υ is the forgetting factor and k is the time delay in terms of the number of sampling intervals. Note that variable forgetting factor can easily be incorporated in the algorithm. This approach has been used in this work.

A5.2 Software used for implementing PI control and self tuning control on the column simulator

Programs in Quickbasic on IBM-PC AT

Key modules	Functions
DYNMODEL	Simulates the dynamic behaviour of the column
STCONTROL	Estimates the self tuning controller parameters
STC	Calculates self tuning control actions

Capabilities of the software package

Implements positional and 1-incremental (ordinary incremental) forms of SV-STC, MD1-STC, MD2-STC and MD3-STC on the basis of first order system. Incorporates moving average filter when incremental versions are implemented.

Implements single loop PI top tray temperature control and simultaneous control of the top and bottom tray temperatures.

Allows the choice of constant forgetting factor and the 3 variable forgetting factors methods given in Chapter 8.

Specification of the reference model and setpoint filter time constant in minutes; only first order models allowwed.

Allows the use of the parameter correction methods to modify a subset of the self tuning controller parameters.

Specification of the control interval, time duration of experiment and the load and setpoint disturbance changes in form of square waves.

Implements the parameter correction and the simplified parameter correction methods.

APPENDIX A6

Description of the programs used for on-line control of the distillation column

Program	Functions
POSTC	Self tuning control computer program. Implements positional SV-STC, MD1-STC, MD2-STC and MD3-STC
KISTC	Self tuning control computer program. Implements incremental SV-STC, MD1-STC, MD2-STC and MD3-STC
PI-decouple	Implements sigle loop PI control, Multiple loop PI control (2 control loops) and multiple loop PI control with steady state simplified decoupling