ADAPTIVE STATE FEEDBACK OF DC MOTOR SPEED CONTROL

NGANTAWEE RUJIRAKAMTHORNCHAI

A THESIS SUBMITTED IN PARTIAL FULFILMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN CONTROL ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2008

KMITL – 2008 – EN – M – 080 – 087

หัวข้อวิทยานิพนธ์	ระบบควบคุมความเร็วมอเตอร์กระแสตรงโดยการป้อนกลับตัวแปรสถานะแบบอแดปทีฟ
นักศึกษา	นาย งานทวี รุจิระกำธรชัย
รหัสนักศึกษา	47060506
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมระบบควบคุม
พ.ศ.	2551
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ. พรสุข รติโรจน์อนันต์
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	รศ. ดร. โยธิน เปรมปราณีรัชต์

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้เสนอ การควบคุมความเร็วรอบของมอเตอร์กระแสตรงด้วยระบบควบคุมแบบอแดปทีฟ ที่ใช้วิธีป้อนกลับตัวแปรสถานะ ในการกำหนดโพลของระบบวงปิดเพื่อปรับปรุงประสิทธิภาพของระบบให้ดีขึ้น ซึ่งการป้อนกลับนี้จะเข้ามาทำหน้าที่แทนการใช้ตัวชดเชยในวิธีการแบบดั้งเดิมที่วิเคราะห์ระบบจากฟังก์ชันถ่ายโอน โดยที่ตัวแปรสถานะที่ใช้ในการป้อนกลับจะถูกสร้างขึ้นจากตัวสังเกตสถานะโดยที่แบบจำลองทางคณิตศาสตร์ของตัวสังเกตสถานะจะมีค่าตามฟังก์ชันถ่ายโอนของระบบที่ถูกประมาณหามาจาก อาร์แอลเอสอัลกอริธึม และ อัตราขยายป้อนกลับจะถูกกำหนดค่าตามแบบจำลองทางคณิตศาสตร์ของตัวสังเกตสถานะนี้ร่วมกับประสิทธิภาพของระบบที่ออกแบบไว้ อีกทั้งอัตราการติดตามสัญญาณอ้างอิงของระบบ จะถูกกำหนดจากรูปแบบของสัญญาณอินพุทที่ป้อนเข้ามา ในวิทยานิพนธ์จะทดสอบประสิทธิภาพการควบคุมของระบบเปรียบเทียบทั้งระบบควบคุมที่นำเสนอกับวิธีการแบบดั้งเดิม รวมถึงความแม่นยำของตัวสังเกตสถานะในการสร้างตัวแปรสถานะที่ใช้ในการป้อนกลับ การทำคอมพิวเตอร์ซิมูเลชันร่วมกับการทดลองในระบบมอเตอร์กระแสตรงจริงให้ผลการทดลองที่ว่า ระบบควบคุมแบบอแดปทีฟที่ใช้การป้อนกลับตัวแปรสถานะสามารถควบคุมความเร็วของมอเตอร์กระแสตรงได้ดีทั้งในเรื่องผลตอบสนองและสัญญาณควบคุม

Thesis Title	Adaptive State Feedback of DC Motor Speed Control
Student	Mr. NGANTAWEE RUJIRAKAMTHORNCHAI
Student ID.	47060506
Degree	Master of Engineering
Program	Control Engineering
Year	2007
Thesis Advisor	Assoc. Prof. Phornsuk Ratiroch-anant
Thesis Co-adviser	Assoc. Prof. Dr. Yothin Prempraneerach

ABSTRACT

This thesis proposes the adaptive dc motor speed control which exerts state feedback method on the system regulation performance. State feedback takes part in the pole placement instead of compensator usually used in classical control that analyses system from the transfer function. The state variables are generated by a state observer whose mathematical model (A, B, C and D) can be adapted to system's parameters estimated from Recursive Least Square algorithm. Consequently, the feedback gain K is defined according to estimated system model and desired close-loop pole. Moreover, for tracking problem, the desired tracking trajectory signal is introduced as the independent reference signal. The experiments are performed in the computer simulation as well as in the real DC motor system. And, as the result of many experiments, the adaptive state feedback control of the dc motor's speed gives adequate performance of response and control signal.

ACKNOWLEDGMENTS

I wish to express my great thanks to Assoc. Prof. Phornsuk Ratiroch-anant and Assoc. Prof. Dr. Yothin Prempraneerach for their valuable helps, suggestion and encouragement throughout the course of this thesis. Without their strong support, this thesis would not come true.

I wish to thank Associate Professor Hiroshi Hirata for his support on research tool such as personal computer and research equipments.

I am extremely grateful to all examiners: Assoc. Prof. Suthian Kiatsunthorn, Assoc. Prof. Dr. Yothin Prempraneerach, Assist. Prof. Dr. Poramate Pranayanuntana, Assoc. Prof. Vipan Preechapanij and Assoc. Prof. Phornsuk Ratiroch-anant for their helpful recommendation.

Finally, I would like to thank my father and mother for their understanding, patience, love and encouragement.

CONTENTS

CONTENTS (cont.)

CONTENTS (cont.)

CONTENTS (cont.)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

Chapter 1

Introduction

It is well known that the adaptive control is effectively used to cope with the systems whose parameters are slowly time-varying or uncertain. This is because of the feature of adaptive control in the sense that its parameters can be adapted to the changing parameters without a priori information about these uncertain or time-varying parameters. With computer based, the controller has the ability to collect dynamic information from the process and compute the controller tuning parameters from these data.

This is quite different from the general control analysis. Normally, in order to design the control system to obtain the required specification, the open loop polynomial of the plant or its behavier must be derived. The controller is set as a constant according to the system parameter and the desired tracking and regulation performance properly chosen by an experienced control engineer. For instance, controllers designed using robust control methods tend to be able to cope with small differences between the true system and the nominal model used for designing. Robust control guarantees that if the changes are within given bounds, the control law need not be changed. Therefore the efficiency of the controller can vary more or less depending on this factor. However, this method is unworkable if the system is so complex that it is difficult to achieve its dynamic response.

1.1 Motivation

Several algorithms for adaptive control have been introduced in conferrences, journals and practical applications. Among these one, Model Identification Adaptive Control (MIAC or self - tuning controller) is one effectively useful in controlling the unknown parameter system. It identify the system by means of measured input and output data and then form an appropriate controller by the identified system. In conventional adaptive control system using pole placement method along with z - transfer function analysis, the problem of unstable poles may appear in the used compensators, even if the closed - loop system is selected stable.

Since, the controller arbitarily re-tunes its compensators to get the plant's parameters having the desired regulation performance (closed loop polynomial), it is possible that the poles of compensators are placed in the unstable area. For example, in the motor speed control system, the moment of inertia usually changes in a wide range. In such case it is well known that the unstable pole appear in the series compensator of controller. And the control performance is deteriorated.

In addition, Another problem in this method, the oscillation in the control signal, always occur in transient due to compensators. This defection can make the actuating device of the physical system damaged.

1.2 Goal and Objective

This thesis is tended to study the use of state space analysis for the adaptive control. The state feedback is used instead of compensators in order to solve the problem that comes with them. Therefore the design using state feedback in part of the regulation performance of the control system is proposed in here.

1.3 Hypothesis to be Tested

Because of no compensators used in this method, it assures that the designer no longer need to concern the problem of unstable compensators. And the fluctuation in the control signal could not present since the control signal is only the combination of the refference signal and the feedback signal.

1.4 Methods and theory

As mentioned above, State feedback is employed in this thesis as the main topic as the regulation part. The state variables is fed back through one matrix feedback gain K. Furthermore, these state variables is estimated by the observer in case the system is unaccessable by any reason. Its model is adapted to the estimated system parameter. The recursive least square algorithm is used in system identification. The recursive parameter estimation is used for the real time estimation. So this adaptive control is the real time control system that use the recursive least square algorithm for estimation.

For experiment of adaptive function, the sample case of the research is selected as the dc motor speed system which has a vary of the moment of inertia. The extra load is added to normal load by the clutch while the motor is running. This adding will affect on the the present speed and the parameter of the system. The experiments are performed in both the computer simulation and the real dc motor.

1.5 Comparison of the Proposed Method with Conventional

In classical method, there is inversion of matrix to compute the parameter for the compensators all the time. This calculation takes long time to find out the exact value of compensator. On the contrary in this method, since the output signal of the controller is the sum of two value and the rest are only basic calculation, the calculation time for control signal is quite shorter than is that of the classical method.

1.6 Limitation of Study

At first, the proposed adaptive control method is tested in the computer simulation in order to find out the possibility and sum up the problems. The experiments are also performed to compare the efficiency of the proposed method with the classical one. Afterward this control method is applied to the real dc motor speed system.

1.7 Process of Study

The thesis is organized into 8 chapters and appendix. Its outline is as follows: Chapter 1 presents introduction of the adaptive control and the purpose of the thesis. Chapter 2 and 3 presents the recursive least squares method for system identification and problem in using this method. Some simulations on system identification are performed.

Chapter 4 presents a basic analysis of control systems in state space. Concept of controllability and observabilty are given here. The transformation off the system model (from transfer function model to state space model) is included in this chapter. Chapter 5 treats the design of control systems in state space. This chapter begin with pole placement design problems followed by the design of state observer. The ideas of tracking and regulation with independent objective or MRAC is also described. The accuracy of the state feedback and the observer tested by program simulation are shown in this chapter.

In chapter 6, the result of the experiments in the computer simulation is given comparing to other conventional method such as R-ST Tribrance , PID and so on. While in chapter 7 only the result of the experiments in dc motor speed system is given. The distiction of simulation and real system are dicussed in here. Chapter 8 gives the overall discussions and suggestions of the proposed design method.

Chapter 2

Recursive System Identification

When the plant is an unknown parameter plant or time varying parameter, the stable controller design through pole placement method can not be achieve. Alternatively, system identification can be performed through an experiment approach to estimate the unknown parameters from input output measured data of the plant. Many identification methods exist, but Recursive Least Squares identification method which is the real-time identification based on the least squares algorithm will be discussed in this chapter. This is due to that the real time identification is the method which can easily be modified to track time varying parameters. Therefore it is appropriate for the time-varying parameter plant.

The derivation, properties and use of the recursive least square for system identification are the objective of the chapter.

2.1 Least Squares Algorithm

Consider Equation (2.1), discrete time transfer function model of a system as shown in Figure 2.1 with control input sequence and output subject to measurable disturbance drift and random noise (their information is in Appendix A).

$$Ay(t) = Bu(t-1) + Dv(t) + D(t) + Ce(t) \tag{2.1}$$

where

$$A = 1 + a_1 z^{-1} + \cdots + a_{n_a} z^{-n_a}$$

$$B = b_0 + b_1 z^{-1} + \cdots + b_{n_b} z^{-n_b}$$

$$D = d_0 + d_1 z^{-1} + \cdots + d_{n_d} z^{-n_d}$$

and

$$C = 1 + c_1 z^{-1} + \cdots + c_{n_c} z^{-n_c}$$

The coefficient of these polynomials (*A, B, C* and *D*) is determined by estimation in case of unknown parametric system.

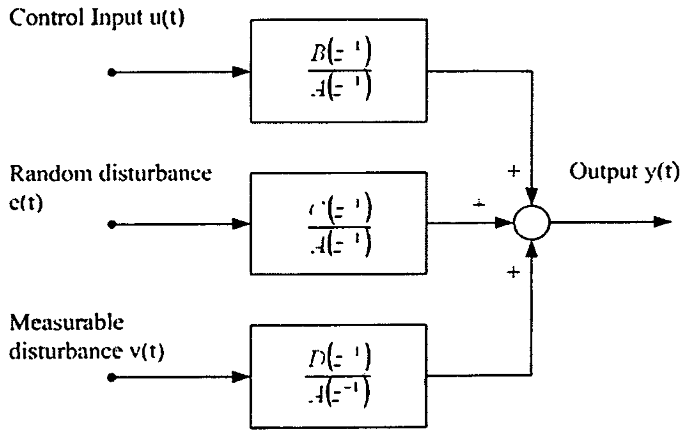

Figure 2.1 Discrete Time system transfer function of a system and disturbance components.

For the estimation purpose, Equation (2.1) is rewritten in form that focuses on the object to be estimated and the data available. The backward shift interpretation of z-1 is used to cast equation

$$y(t) = \varphi^T(t)\theta + e(t) \tag{2.2a}$$

where θ is the vector of unknown parameters defined by

$$\theta^T = \left[-a_1, \ldots, -a_{n_a}, b_0, \ldots, b_{n_b}, d_0, \ldots d_{n_d}, c_1, \ldots, c_{n_c}\right] \tag{2.2b}$$

and $\varphi(t)$ is a regression (or measured data) vector containing the input/output variables information from the system. It is defined by

$$\varphi^T(t) = [y(t-1), \ldots, y(t-n_a), \ldots, u(t-1), \ldots, u(t-n_b-1), \ldots, v(t), \ldots, v(t-n_d), \ldots, e(t-1), \ldots, e(t-n_c)] \tag{2.2c}$$

However, the assumed model representing the system mostly is ARX model, Figure 2.2, which is the discrete time model for general plant. Therefore it is convenient to rewrite the measured data vector in Equation (2.2d) according to the ARX model.

$$\varphi^T(t) = [y(t-1), \ldots, y(t-n_a), \ldots, u(t-1), \ldots, u(t-n_b-1)] \tag{2.2d}$$

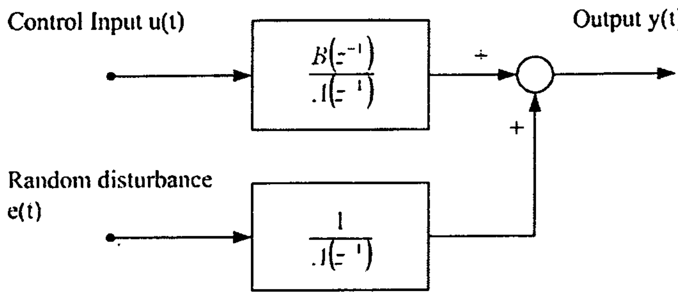

Figure 2.2 Auto regressive with control (ARX Model).

Equation (2.2) is assumed to be an exact description of the system, producing the true data for the regression vector X. In order to determine the vector X from X, the estimate model is

$$y(t)=\varphi^T(t)\hat{\theta}+\hat{e}(t) \tag{2.3}$$

$\hat{\theta}$ is a vector of adjustable model parameters and $\hat{e}(t)$ is the corresponding modeling error. The main aim is to find $\hat{\theta}$ that overall modeling error becomes minimal. Equation (2.2) and (2.3) imply that $\hat{e}(t)$ depends on $\hat{\theta}$.

$$\hat{e}(t)=e(t)+\varphi^T(t)(\theta-\hat{\theta}) \tag{2.4}$$

In order to estimate the system parameters, a number N of the consecutive data vector in $\varphi(t)$ are required. It is indicated in the vector/matrix form of Equation (2.3) as

$$y(t)=W(t)\hat{\theta}+\hat{e}(t) \tag{2.5}$$

in which

$$y(t)=[y(1)\quad y(2)\quad \ldots\quad y(N)]$$
$$\hat{e}(t)=[\hat{e}(1)\quad \hat{e}(2)\quad \ldots\quad \hat{e}(N)]$$

and

$$W(t)=[\varphi^T(1)\quad \varphi^T(2)\quad \ldots\quad \varphi^T(N)]$$

If the number N must not be less than the number m of unknown parameters in the vector X, then Equation (2.5) can be solved as the set of linear functions. However, when noise is present in practical systems, the error induced by noise defined by Equation (2.6) can become much great. Furthermore, the number N must be far larger than m.

The Modeling error is defined by rearranging Equation (2.5).

$$\hat{e}(t) = y(t) - W(t)\hat{\theta} \tag{2.6}$$

The technique most widely used to minimize such error is Linear Least Squares. The main idea of the method is to minimize J, the sum of squares of errors.

$$J = \sum_{i=1}^{t} \hat{e}^2(i) = \hat{e}^T \hat{e} \tag{2.7}$$

To find the least squares estimate, rewrite Equation (2.7) by replacing the error with the data vector and parameter vector in Equation (2.6).

$$J = (y - W\hat{\theta})^T (y - W\hat{\theta}) \tag{2.8}$$

$$= y^T y - \hat{\theta}^T W^T y - y^T W\hat{\theta} + \hat{\theta}^T W^T W\hat{\theta} \tag{2.9}$$

Set the derivative of J with respect to $\hat{\theta}$ to zero for stationary point.

$$\frac{\partial J}{\partial \hat{\theta}} = -2W^T y + 2W^T W\hat{\theta} = 0 \tag{2.10}$$

$$W^T W\hat{\theta} = W^T y \tag{2.11}$$

The solution of Equation (2.11) is unique minimal value if the second derivative matrix in Equation (2.12) is positive definite (See Appendix X for further information of positive definite).

$$\frac{\partial^2 J}{\partial \hat{\theta}^2} = 2(W^T W) \tag{2.12}$$

Hence the least squares estimator for the parameter vector is

$$\hat{\theta} = \left(W^T W\right)^{-1}\left(W^T y\right) \tag{2.13}$$

The resulting modeling error X is denoted by

$$\eta^T = \left[\eta(1) \quad \eta(2) \quad \eta(3) \quad \dots \quad \eta(N)\right] \tag{2.14}$$

whose components are called residuals.

2.2 Properties of the Least Squares Estimator

The least squares estimator X is a random variable whose properties can be analyzed using Equation (2.3) for the actual system and disturbance. Two properties, bias and covariance are important in this respect.

Bias: The term of bias refers to the systematic error can occur in the parameter estimate. Substituting Equation (2.5) to Equation (2.13) yields

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

$$\hat{\theta} = \theta + \left(W^T W\right)^{-1}\left(W^T e\right) \tag{2.15}$$

and the term bias is given by a rearrangement of Equation (2.15).

$$\hat{\theta} - \theta = \left(W^T W\right)^{-1}\left(W^T e\right) \tag{2.16}$$

$$\hat{\theta} - \theta = \left[\sum_{t=1}^{N} \varphi(t)\varphi^T(t)\right]^{-1}\left[\sum_{t=1}^{N} \varphi(t)e(t)\right]$$

Multiply the right side of equation by N/N to obtain.

$$\hat{\theta} - \theta = \left[\frac{1}{N}\sum_{t=1}^{N} \varphi(t)\varphi^T(t)\right]^{-1}\left[\frac{1}{N}\sum_{t=1}^{N} \varphi(t)e(t)\right] = \left(N^{-1}W^T W\right)^{-1} N^{-1} W^T e \tag{2.17}$$

Equation (2.18) is used to show that the estimate model is unbiased in case the data $\varphi(t)$ and the noise $e(t)$ are correlated.

$$\underset{N\to\infty}{p\lim}\left(N^{-1}W^{T}e\right)=0 \tag{2.18}$$

Covariance: The term covariance is related to the spread of estimates arising from the random errors (the accuracy of estimates).

When the data $\varphi(t)$ and $e(t)$ are uncorrelated and $e(t)$ is zero mean, the covariance of least squares is defined by

$$\operatorname{cov}\left(\hat{\theta}\right)=E\left[\left(\hat{\theta}-\theta\right)\left(\hat{\theta}-\theta\right)^{T}\right]$$

$$\operatorname{cov}\left(\hat{\theta}\right)=E\left[\left(\left(W^{T}W\right)^{-1}W^{T}e\right)\left(\left(W^{T}W\right)^{-1}W^{T}e\right)^{T}\right]$$

$$\operatorname{cov}\left(\hat{\theta}\right)=E\left[\left(W^{T}W\right)^{-1}W^{T}ee^{T}W\left(W^{T}W\right)^{-1}\right]$$

$$\operatorname{cov}\left(\hat{\theta}\right)=E_{x}\left[\left(W^{T}W\right)^{-1}W^{T}E_{e}\left[ee^{T}\right]W\left(W^{T}W\right)^{-1}\right] \tag{2.19}$$

In this case the errors are zero mean white noise and uncorrelated, so

$$E_{e}\left[ee^{T}\right]=\sigma_{e}^{2}I_{m} \tag{2.20}$$

As consider from Equation (2.19) and Equation (2.20), the covariance of error in LS estimates is given by

$$\operatorname{cov}\left(\hat{\theta}\right)=\sigma_{e}^{2}IE_{x}\left[\left(W^{T}W\right)^{-1}\right] \tag{2.21}$$

If the data is deterministic, the expectation operation can be cut.

Equation (2.21) indicates that the matrix $\left(W^T W\right)^{-1}$ the most importance in self tuning system because it directly dominates on the variability and covariability of the parameter estimates.

2.3 Relationship between the Residual and Data

The relationship between the residual (modeling error sequence) and data is briefly discussed. Noting that $\hat{e} = \eta$ and pre multiplying the Equation (2.5) by W^T , gives

$$W^T y(t) = W^T W \hat{\theta} + W^T \eta \tag{2.22}$$

Equation (2.13), $\hat{\theta} = \left(W^T W\right)^{-1}\left(W^T y\right)$, and Equation (2.22) imply that

$$W^T \eta = 0 \tag{2.23}$$

$$[\varphi(1) \quad \varphi(2) \quad \varphi(3) \quad \ldots \quad \varphi(N)]\eta = 0 \tag{2.24}$$

$$\varphi(1)\eta(1) + \varphi(2)\eta(2) + \varphi(3)\eta(3) + \ldots + \varphi(N)\eta(N) = 0 \tag{2.25}$$

By recalling the definition of $\varphi(t), \varphi^T(t) = [y(t-1), \ldots, y(t-n_a), \ldots, u(t-1), \ldots, u(t-n_b-1)]$, Equation (2.25) can be rewritten as

$$\left.\begin{aligned} &\sum_{t=1}^{N} y(t-i)\eta(t) = 0 \qquad \text{for} \quad i = 1, \ldots, n_a \\ &\sum_{t=1}^{N} u(t-i)\eta(t) = 0 \qquad \text{for} \quad i = 1, \ldots, n_b + 1 \end{aligned}\right\} \tag{2.26}$$

And so on for other entries in $\varphi(t)$. Equation (2.26) imply that

$$\left.\begin{aligned} &E[y(t-i)\eta(t)] = 0 \qquad \text{for} \quad i = 1, \ldots, n_a \\ &E[u(t-i)\eta(t)] = 0 \qquad \text{for} \quad i = 1, \ldots, n_b + 1 \end{aligned}\right\} \tag{2.27}$$

These two equations, Equation (2.26) and Equation (2.27), sometimes are the expression for orthogonality property of least squares and are the basis of other results.

2.4 Recursive Least Squares

In self tuning control, the estimation should allow the update estimated model at each sampling interval as the new available data X as describe in Figure 2.3, the estimation process. The model based on past information X is used to obtain an estimated Y. An error E generated by comparing the estimated Y with the real output Y will correct the old model X(t) to the new one X(t+1). This recursive prediction corrector results in saving significantly computation time. The estimates in equation (2.8) at step t and t+1 are given by

$$\hat{\theta}(t) = \left(W^T(t)W(t)\right)^{-1}\left(W^T(t)y(t)\right) \tag{2.28}$$

$$\hat{\theta}(t+1) = \left(W^T(t+1)W(t+1)\right)^{-1}\left(W^T(t+1)y(t+1)\right) \tag{2.29}$$

respectively. Where

$$W^T(t+1)W(t+1) = \left[W^T(t) \quad \varphi(t+1)\right]\begin{bmatrix} W(t) \\ \varphi^T(t+1) \end{bmatrix}$$

$$= W^T(t)W(t) + \varphi(t+1)\varphi^T(t+1) \tag{2.30}$$

And

$$W^T(t+1)y(t+1) = \left[W^T(t) \quad \varphi(t+1)\right]\begin{bmatrix} y(t) \\ y(t+1) \end{bmatrix}$$

$$= W^T(t)y(t) + \varphi(t+1)y(t+1) \tag{2.31}$$

So $\varphi(t+1)$ is easily added to the old matrix $W^T(t)W(t)$ to get the new $W^T(t+1)W(t+1)$. But actually a way to update the inverse of $W^T(t+1)W(t+1)$ without continuous inversion of matrix is required. Therefore to obtain this requirement, notation $P(t)$ and $B(t)$ are introduced as

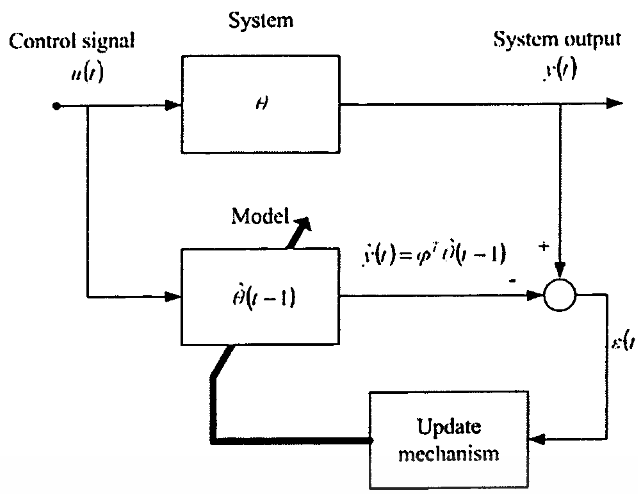

Figure 2.3 Visualization of recursive system identification as an interactive process.

$$\left.\begin{aligned} P(t) &= \left[W^T(t)W(t)\right]^{-1} \\ B(t) &= W^T(t)y(t) \end{aligned}\right\} \tag{2.32}$$

And Equation (2.29), (2.30) and (2.31) yield,

$$\left.\begin{aligned} \hat{\theta}(t+1) &= P(t+1)B(t+1) \\ \hat{\theta}(t) &= P(t)B(t) \end{aligned}\right\} \tag{2.33}$$

$$P^{-1}(t+1) = P^{-1}(t) + \varphi(t+1)\varphi^T(t+1) \tag{2.34}$$

and

$$B(t+1) = B(t) + \varphi(t+1)y(t+1) \tag{2.35}$$

The crucial step is to establish the direct update from $P(t)$ to $P(t+1)$ by applying the Matrix Inversion Lemma.

$$(A+BCD)^{-1} = A^{-1} - A^{-1}B\left(C^{-1} + DA^{-1}B\right)^{-1} DA^{-1} \tag{2.36}$$

to (2.34). Assigning $A = P^{-1}(t)$, $B = \varphi(t+1)$, $C = 1$, $D = \varphi^T(t+1)$ gives

$$P(t+1) = P(t)\left[I - \frac{\varphi(t+1)\varphi^T(t+1)P(t)}{1+\varphi^T(t+1)P(t)\varphi(t+1)}\right] \tag{2.37}$$

Equation (2.37) gives the means to update $P(t)$ to $P(t+1)$ without matrix inversion and it can be combined with Equation (2.35) to obtain a direct recursion for X in the following way.

First, define the error variable as

$$\varepsilon(t+1) = y(t+1) - \varphi^T(t+1)\hat{\theta}(t) \tag{2.38}$$

and used variable $y(t+1)$ in this equation for $y(t+1)$ in Equation (2.35). This gives

$$B(t+1) = B(t) + \varphi(t+1)\varphi^T(t+1)\hat{\theta}(t) + \varphi(t+1)\varepsilon(t+1) \tag{2.39}$$

Then substitute for $B(t)$, $B(t+1)$ using Equation (2.32) to get

$$P(t+1)^{-1}\hat{\theta}(t+1) = P(t)^{-1}\hat{\theta}(t) + \varphi(t+1)\varphi^T(t+1)\hat{\theta}(t) + P(t+1)\varphi(t+1)\varepsilon(t+1)$$

$$P(t+1)^{-1}\hat{\theta}(t+1) = \left[P(t)^{-1} + \varphi(t+1)\varphi^T(t+1)\right]\hat{\theta}(t) + P(t+1)\varphi(t+1)\varepsilon(t+1)$$

$$P(t+1)^{-1}\hat{\theta}(t+1) = P(t+1)^{-1}\hat{\theta}(t) + P(t+1)\varphi(t+1)\varepsilon(t+1)$$

$$\hat{\theta}(t+1) = \hat{\theta}(t) + P(t+1)\varphi(t+1)\varepsilon(t+1) \tag{2.40}$$

Multiplying Equation (2.37) by $\varphi(t+1)$ gives

$$P(t+1)\varphi(t+1) = P(t)\left[I - \frac{\varphi(t+1)\varphi^T(t+1)P(t)}{1+\varphi^T(t+1)P(t)\varphi(t+1)}\right]\varphi(t+1) \tag{2.41}$$

and assign new variable $L(t+1) = P(t+1)\varphi(t+1)$, so that

$$L(t+1)=\left[\frac{P(t)\varphi(t+1)}{1+\varphi^T(t+1)P(t)\varphi(t+1)}\right] \qquad (2.42)$$

From Equation (2.41) and (2.42), Equation (2.37) and (2.40) are transformed to the equation that have variable $L(t+1)$ as their content and named as shown below.

Estimated System Parameter

$$\hat{\theta}(t+1)=\hat{\theta}(t)+L(t+1)\varepsilon(t+1) \qquad (2.43)$$

The adaptation gain adjusting

$$P(t+1)=P(t)-L(t+1)[P(t)\varphi(t+1)]^T \qquad (2.44).$$

The recursive least squares algorithm for estimating X is as follows:

At time step t+1:

1) Form $\varphi(t+1)$ using new data.
2) Form $\varepsilon(t+1)$ as equation:

$$\varepsilon(t+1)=y(t+1)-\varphi^T(t+1)\hat{\theta}(t)$$

3) Form $L(t+1)$ using Equation (2.42)
4) Update $\hat{\theta}(t)$ to obtain $\hat{\theta}(t+1)$:
5) Form $P(t+1)$using equation (2.44)
6) Wait for the next time step to elapse and loop back to step 1).

2.5 Residual and Prediction Errors

An influence variable in RLS algorithm is the modeling error $\varepsilon(t)$which can be related to the residual associated with the least square procedure. From Equation (2.38), that error is defined by $\varepsilon(t)=y(t)-\varphi^T(t)\hat{\theta}(t-1)$. In other words it is the error between the current system output $y(t)$and the predicted output using parameter estimates $\hat{\theta}(t-1)$. For this reason it is normally called the priori output prediction error.

While the true modeling error (called residual or a posterior prediction error) at the current time T is

$$\eta(t) = y(t) - \varphi^T(t)\hat{\theta}(t) \tag{2.45}$$

There is an only difference between e and n that n is based upon the parameter estimates at step t. This difference should become insignificant after the estimation procedure has run for a long time. The difference is significant during the first few recursions though. The algorithm in section 2.6 will exploit it. And the residual can be indicated associated with $\varepsilon(t)$ in

$$\eta(t) = \varepsilon(t) - \varphi^T(t)\left[\hat{\theta}(t) - \hat{\theta}(t-1)\right] \tag{2.46}$$

Using Equation (2.43), the difference $\hat{\theta}(t) - \hat{\theta}(t-1)$ can be substituted in Equation (2.46) to yield

$$\eta(t) = \left[1 - \varphi^T(t)L(t)\right]\varepsilon(t) \tag{2.47}$$

From Equation (2.42), the above expression can be re-expressed as

$$\eta(t) = \frac{\varepsilon(t)}{1 + \varphi^T(t)P(t-1)\varphi(t)} \tag{2.48}$$

2.6 Recursive Extended Least Squares or RELS

This method only adds the priori or posteriori prediction error in data vector. Therefore the parameter vector $\hat{\theta}(t)$ is extended corresponding to the data vector $\varphi(t)$ as Equation (2.49) and (2.50).

$$\hat{\theta}^T = \left[-a_1, \ldots, -a_{n_a}, b_0, \ldots, b_{n_b}, c_1, \ldots, c_{n_c}\right] \tag{2.49}$$

$$\varphi^T(t) = \left[y(t-1), \ldots, y(t-n_a), \ldots, u(t-1), \ldots, u(t-n_b-1), \varepsilon(t-1), \ldots, \varepsilon(t-n_c)\right] \tag{2.50}$$

The error $\varepsilon(t)$ in data vector $\varphi(t)$ can also be replaced by the residual $\eta(t)$ (RML algorithm):

$$\varphi^T(t) = \left[y(t-1), \ldots, y(t-n_a), \ldots, u(t-1), \ldots, u(t-n_b-1), \eta(t-1), \ldots, \eta(t-n_c)\right] \tag{2.51}$$

The main concept of RELS is still as same as that of normal RLS, minimizing the covariance of errors. The RELS algorithm is then:

RELS or RML algorithm:

At time step t+1

1. From $\varphi(t+1)$ using new data $u(t+1)$, $y(t+1)$ and

$$\varepsilon(t) = y(t) - \varphi^T(t)\hat{\theta}(t-1) \text{ or } \eta(t) = y(t) - \varphi^T(t)\hat{\theta}(t)$$

2. $$P(t+1) = P(t)\left[I - \frac{\varphi(t+1)\varphi^T(t+1)P(t)}{1+\varphi^T(t+1)P(t)\varphi(t+1)}\right]$$

3. $$\hat{\theta}(t+1) = \hat{\theta}(t) + P(t+1)\varphi(t+1)\varepsilon(t+1)$$

2.7 Conclusion

The chapter has developed the basic ideas and algorithm associated with recursive least square and recursive extened least square estimation. Using these algorithms in practical situation will be provided in Chapter 3 which discusses the effect of some initial parameters and possible problem.

Chapter 3

Using Recursive Estimation

In chapter 2, the recursive form of least squares estimation and the algorithm basic are described. Then, here how to apply these algorithms in practical situation is given. There are roughly four parts (section 3.1 to 3.4) in setting and operating this self tuning estimator. The different versions of least square are briefly discussed in section 3.5. In the last section some experiments in simulation and practical are performed under given conditions.

3.1 Initializing the Estimator

The choice of values for the data vector, parameter vector and covariance matrix at the first time step are considered.

3.1.1 Initial Values for the Data Vector

Usually the way to fill data to the data vector is up to the number of maximum order of its entries as Equation (3.1). The recursion will start after the first n step of this number. This ensures that the RLS estimator always has the vector data filled at the beginning.

$$n = \max\{n_a \quad n_b + 1 \quad n_d \quad n_d \quad n_c\} \tag{3.1}$$

3.1.2 Initial Parameter Values

The typical methods are used for setting the initial estimate of the parameter vector $\hat{\theta}(0)$ as follows:

i. Sometimes if the system can be approximately known, these parameters are set in the parameter vector.

ii. The system is expected being an integrator with unit gain. Thus the initial parameter would be selected as: $a_1 = -1$; $b_0 = t_s$; while $a_i = 0$; $b_i = 0$;

The initial value of other parameter estimates (noise and disturbance) are set at zero.

3.1.3 Initial Values of the Covariance Matrix

To understanding that the size of the covariance matrix has the effect on itself, its function is written here in Equation (3.2).

$$P(t)=\left[P^{-1}(0)+\sum_{i=1}^{t}\varphi(i)\varphi^{T}(i)\right]^{-1} \tag{3.2}$$

It is simple to see that the influent of $P(0)$ depends upon whether the inversion of its initial value is large or small. A standard choice $P(0)$ for is the unit matrix scaled by a positive scalar, ie:

$$P(0)=rI_{m} \tag{3.3}$$

For large $P(0)$ the value of r is adjusted within the range 100 to 1000 and for small $P(0)$ the value of r is adjusted within the range 1 to 10. The criterion to select the value of $P(0)$ is simply expressed below:

i. If priori knowledge of system parameters is known in the model, $\hat{\theta}(0)$ is close to the true values, $P(0)$ should be large. It causes rapid and large recursions in the parameter estimates.
ii. If priori no knowledge of system parameters is unknown in the model, $\hat{\theta}(0)$ is far from the true values, $P(0)$ should be small. By this small $P(0)$, the parameters change slowly.

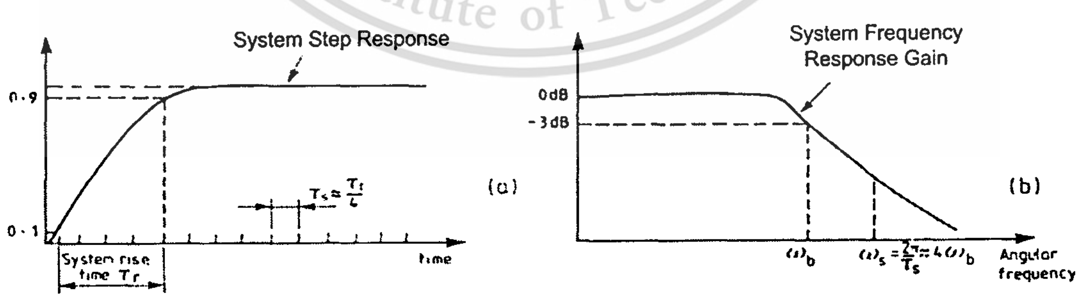

Figure 3.1 Sample interval selection: illustrating reasonable sample intervals in (a) the time domain and (b) the frequency domain.

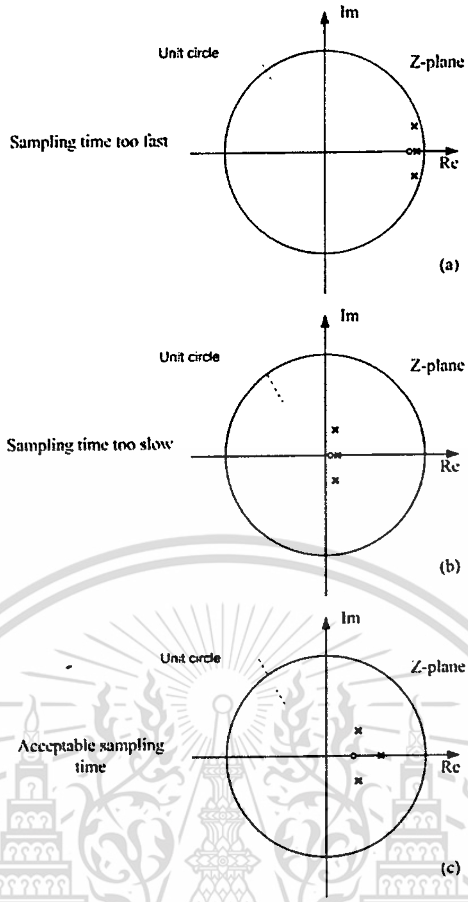

Figure 3.2 Showing typical pole/zero patterns for (a) fast, (b) slow and (c) acceptable sampling rate.

3.2 Specification of the Estimator

This section concerns the estimator model selection and other involved parts when the estimator is coupled to the real world's system.

3.2.1 Sample Interval Selection

There are some useful guidelines for selecting the sample interval T_s in recursion. Figure 3.1 illustrates the idea of these guiding rules as follows:

i. There should be at least four points on the system response in transient.

ii. The sampling frequency ω_s (1 / sampling interval T_s) should be four times the system bandwidth in order to capture all the information in this frequency band.

Certain symptoms indicating the improper T_s appear in the estimated model. The poles of model clustered around the point of either z = 1 or z = 0. The relationship between T_s and model's poles in z plane is shown in Equation (3.4). In case of stable pole $s \le 0$:

$$z = e^{sT_s} \to 1 \text{ as } T_s \to \infty \text{, sampling too slowly.}$$
$$z = e^{sT_s} \to 0 \text{ as } T_s \to 0 \text{, sampling too rapidly.} \qquad (3.4)$$

Therefore the estimated model's poles should appropriately spread in the unit circle on z plane as shown in Figure 3.2. These remarks are generally correct for recursive estimation. Some additional considerations are fund, however, in certain self tuning application such as:

Bandwidth Limitations:

In control system, the objective is often to change the response of entire system by feedback. The sample interval of digital control will limit the attainable closed loop speed by ZOH working as a low pass filter as figure 3.3.

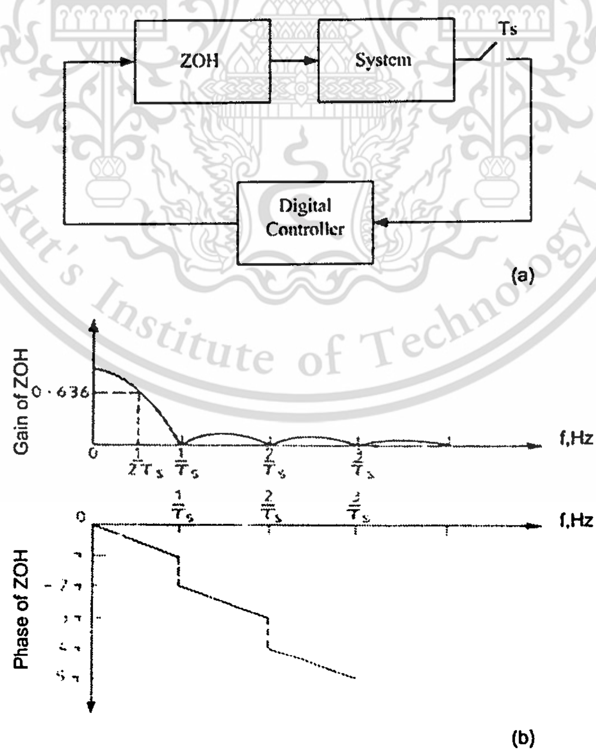

Figure 3.3 Illustrating the bandwidth limiting properties of the ZOH: (a) as a feedback loop element, (b) its frequency response.

External Constraints:

Some applications require the sampling interval being the values according to external factors: traditional, specification of used devices and so on. In spite of previous statement, the RLS can operate over a wide range of recommended sampling interval.

3.2.2 Model Type

The idea is decide which of the components, such as disturbance, in the general model are present or which of the assumed model is the most suitable.

$$Ay(t) = Bu(t) + Ce(t) \tag{3.5}$$

For example, Equation (3.5) is the model of the system which is subjected to only the control signal and noise.

3.2.3 Model Order

The model order decision concerns the selection of integer n_a, n_b, n_c and etc. These decisions should be the physical consideration of the system with $n_b < n_a$. The following remarks are proper for selecting n_c of the noise model:

i. $n_c = 0$, if the system is deterministic or there are very low.

ii. $n_c = 1$ or $n_c = 2$ (maximum in practical), if the random noise is too significant.

In many self tuning controls it is possible to set the model orders less than the technical correct values. This is because it is important to capture the dominant dynamic features

3.2.4 System Delay

The delay through a system in discrete time can be calculated from knowledge of the continuous time transport delay and the sample interval. If the transport delay is τ_d then the corresponding discrete time d is given by

$$d = \text{int}\left\{\frac{\tau_d}{T_s}\right\} + 1 \tag{3.6}$$

The system delay is always at least one sampling time $d = 1$ because of ZOH as shown in Equation (3.6).

3.3 Operation Conditions

Mainly this section involves nonlinear behavior and inadequate excitation signal. The way to dealt with these subjects is divided into following subsections.

3.3.1 System Linearity and Signal Limiting

The assumption that the signal sources and system are linear is only conditionally true in reality. At most they can be expected to be linear in the operating range. Two major implication of this subject are described as follows:

Signal Levels:

Many systems and signal sources have nonlinear characteristics associated with actuation and sensing, Figure 3.4. Figure 3.5 shows a possible input nonlinearity known as saturation and dead zone. Signal limiter and dead zone compensator are applied to solve these problems.

Another input nonlinearity called slew rate limit is the maximum rate of signal change of the actuator. For large and rapid changing, it becomes the major problem of this thesis which adversely affects on the actuating devices. This can also be encountered in classical pole placement method (using compensators).

Operating Levels:

The system normally functions at a specific operating level. The system is approximately linear when it is considered in this area, for small excursions.

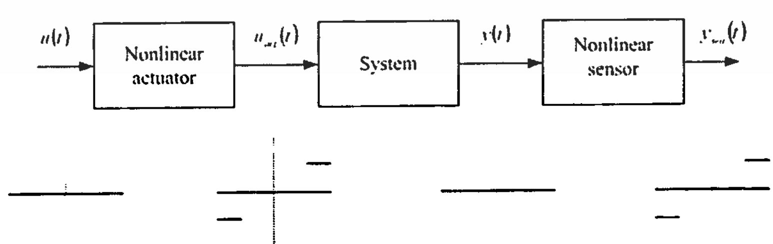

Figure 3.4 Illustrating the possible presence of input (actuation) nonlinearity and output (sensor) nonlinearity.

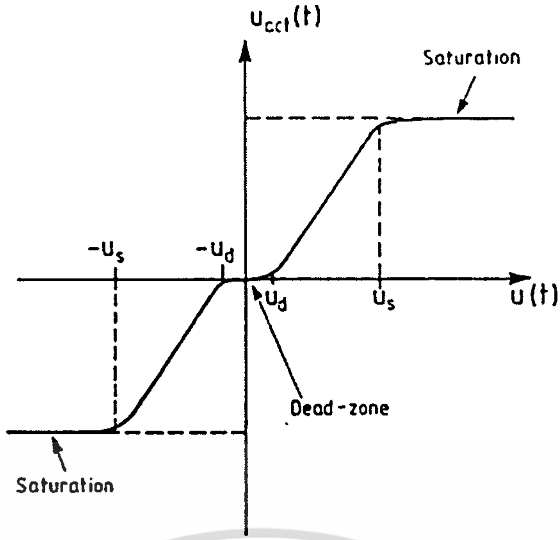

Figure 3.5 Showing a typical actuator or input nonlinearity with saturation limits $\pm U_s$ and dead zone limit of $\pm U_d$.

3.3.2 Signal Filtering

In some system, the output variable $y(t)$ is subject to significant wide band noises which come from the output measurement. Such noised is called sensor noise. It should be removed or minimized by means of filter since it has no relation to process. A normal low pass filter could be adequate.

3.3.3 Quantization

A further problem in recursive estimation concerns the quantization process (analog to digital). With the modern A/D over 12 bits converter, the quantization is quite small. However it should be ensure that the converters are operating in their full range. The amplitude of the signal should not be so small that only a few bits are used.

3.4 Covariance Management

As mentioned before about the function of covariance matrix $P(t)$ (sometime called adjustment gain) on which the estimation depends, in this section the management to keep $P(t)$ in good condition is discussed. The RLS algorithm is written here again.

$$\hat{\theta}(t+1) = \hat{\theta}(t) + P(t+1)\varphi(t+1)\varepsilon(t+1) \quad (3.7)$$

$$P(t+1)=P(t)\left[I_m-\frac{\varphi(t+1)\varphi^T(t+1)p(t)}{1+\varphi^T(t+1)p(t)\varphi(t+1)}\right] \tag{3.8}$$

3.4.1 Presence of Feedback

Since the feedback signal can cause ambiguity in the relationship between input and output, so the independent signal is added into the loop as usual control as indicated in Figure 3.6.

3.4.2 Excitation Level of Signal

In recursive estimation, the input signal is sufficient to excite the estimator to identify the unique system parameter. In practical terms the following forms of signal are suitable for recursive estimation.

Square wave

Pseudo random binary noise

Uniformly distributed noise

Gaussian distributed noise

3.4.3 Tracking Parameter Change

The adaptive control is designed to adapt its control figure to the vary system in real time unlike the self tuning control. This variation can result from many reasons such as operating condition change. To solve the parameter variation problem the forgetting factor is provided to the estimator. The concept and importance of forgetting factor are discussed in next section.

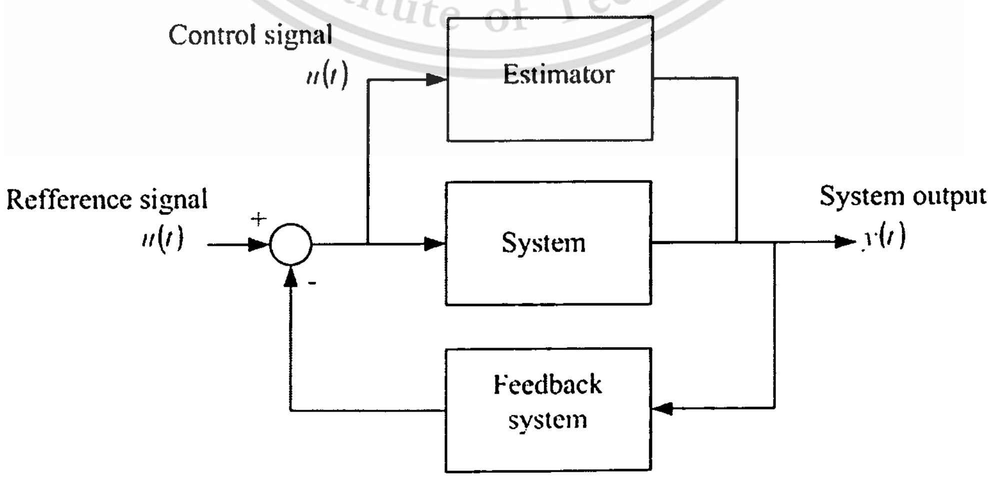

Figure 3.6 Operational condition involving feedback.

3.4.4 Forgetting Factor

The forgetting factor technique is the most popular way of getting adaptable recursive estimator. Its idea is to weight the information of the least square function.

$$\bar{J}_t = \sum_{i=1}^{t} \lambda^{t-i} \hat{e}^2(i) \tag{3.9a}$$

The discounting effect of λ is upon the past errors as indicated in Equation (3.9b).

$$\bar{J}_t = \lambda \bar{J}_{t-1} + \hat{e}^2(i) \tag{3.9b}$$

Then by applying Equation (4.9) in Equation () to construct the minimum of $\bar{J}$ (see prove in Appendix), the covariance matrix $P(t)$ becomes

$$P^{-1}(t) = \lambda P^{-1}(t-1) + \varphi(t)\varphi^T(t) \tag{3.10}$$

That is

$$P(t+1) = \lambda^{-1} P(t)\left[I_m - \frac{\varphi(t+1)\varphi^T(t+1)P(t)}{\lambda + \varphi^T(t+1)P(t)\varphi(t+1)} \right] \tag{3.11}$$

3.45 Estimator Wind-up

From Equation (3.10) if no data information (very small or zero value) enters the data vector, the covariance matrix $P(t)$ will increase every step. That means the previous data information is reset to original zero value. This causes the parameter estimates become erratic until the estimation completely fails because of numerical overflow problem. The Estimator wind-up is prevented by use of such techniques as Start-Up Forgetting Factor.

Start-Up Forgetting Factor:

As stated above for the ideal forgetting factor, it should be unity when the system parameters are constant. Conversely it should be low when system parameters change. The concept of this method is that, at the beginning, the forgetting factor is small and increases at specific rate toward unity. To do so, the forgetting factor λ becomes the function associated with time (the solution of first order differential equation).

$$\lambda(t) = \lambda(0) + (1 - \lambda(0))\left(1 - e^{-t/t_f}\right) \tag{3.12}$$

Where t_f is time constant of the system. And Equation (3.12) can be cast in the simple recursion form.

$$\lambda(t) = \alpha\lambda(t-1) + (1-\alpha);\ \alpha = e^{-1/t_f} \tag{3.13}$$

3.5 Computational Alternatives for Recursive Estimation

Sometimes the problem of inadequate computational precision or restricted computation time may be encountered in use of the basic form of the recursion based on Matrix Inversion Lemma.

The matrix inversion lemma version of RLS can be ill conditioned. In particular, errors can accumulate and cause the error in both estimates and covariance matrix. A class of matrix factorization algorithm, Numerically Robust Recursion, gives increased computational precision. It has approximately the same program operating requirement as the basic RLS.

Some situations exist, particularly in digital signal processing, in which the matrix inversion and is too slow. A class of fast algorithm, Fast Least Square Recursion can overcome such problem. However, in most circumstances, the standard RLS is completely adequate.

3.6 Simulation Example

In this section the RLS algorithm is tested through its operation to estimate system parameter in case of parameter invariant and variant system.

3.6.1 Parameter Invariant System

In this simulation example, the square wave of 5 v amplitude and 200 samplings period as shown in Figure 3.7 is used as the input excitation signal to the plant without the presence of noise and constant disturbance. The RLS algorithm estimates the system's parameters under the following examples:

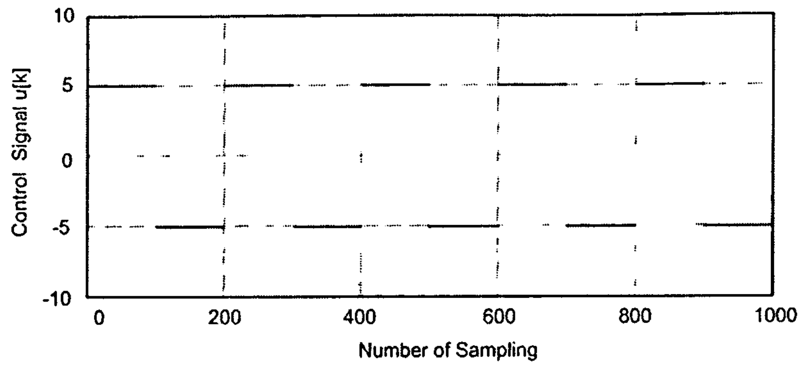

Figure 3.7 Reference signal (5 cycles pulse signal).

Example1. The plant transfer function $G_1(z^{-1})$ is defined as

$$G_1(z^{-1}) = z^{-1}\frac{0.0562 + 0.0438z^{-1}}{1 - 1.4574z^{-1} + 0.4724z^{-2}}$$

Given RLS Estimator parameters $N_a = 2$, $N_b = 1$, $d = 1$, $P(0) = 10^4 I$, $\hat{\theta}^T = [0.1 \quad 0.2 \quad 0.3 \quad 0.4]$ and the sampling number of 1000 with sampling time $T_s = 0.001$.

Simulation results:

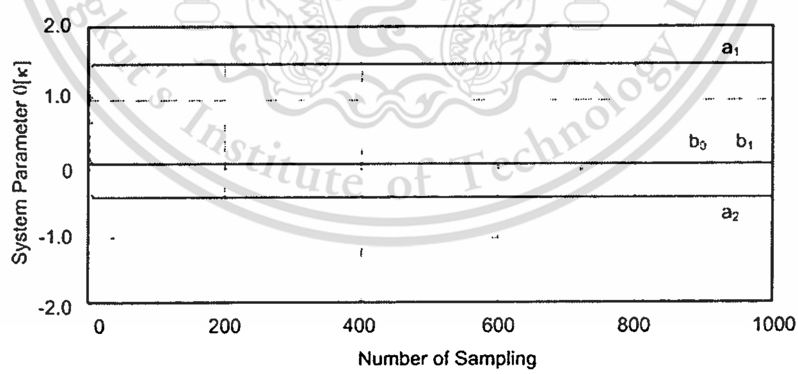

Figure 3.8 Estimated parameters of $G_1(z^{-1})$.

Example 2. The plant transfer function $G_2(z^{-1})$ is defined as

$$G_2(z^{-1}) = z^{-1}\frac{0.0281 + 0.0220z^{-1}}{1 - 1.4651z^{-1} + 0.4726z^{-2}}$$

Given RLS Estimator parameter $N_a = 2$, $N_b = 1$, $d = 1$, $P(0) = 10^4 I$, $\hat{\theta}^T = [0.1 \quad 0.2 \quad 0.3 \quad 0.4]$ and the sampling number 1000 samplings with sampling time $T_s = 0.001$.

Simulation results:

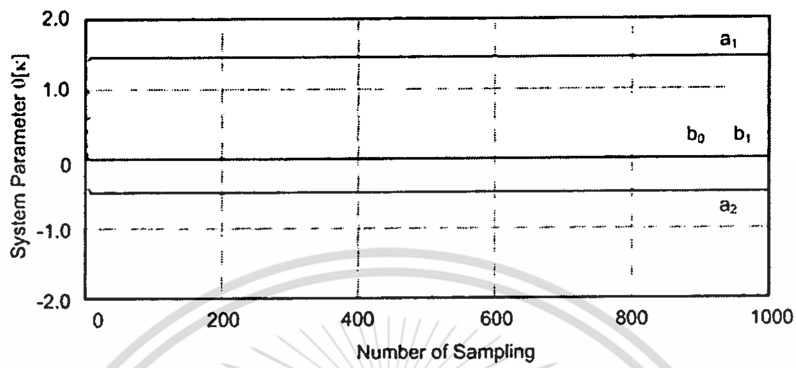

Figure 3.9 Estimated parameters of $G_1(z^{-1})$.

3.6.2 Parameter Variant System

In this example, the plant parameters are changed from $G_1(z^{-1})$ to $G_2(z^{-1})$ at the sampling number 200 and then change back at the sampling number of 600.

$$G_1(z^{-1}) = z^{-1}\frac{0.0562 + 0.0438z^{-1}}{1 - 1.4574z^{-1} + 0.4724z^{-2}} \quad G_2(z^{-1}) = z^{-1}\frac{0.0281 + 0.0220z^{-1}}{1 - 1.4651z^{-1} + 0.4726z^{-2}}$$

The RLS Estimator parameter and reference signal are as same as previous example.

Simulation results:

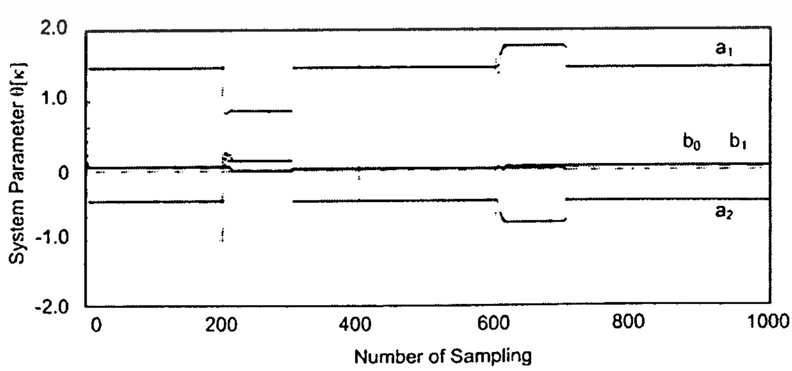

Figure 3.10 Estimated parameters of $G_1(z^{-1})$ and $G_2(z^{-1})$.

When consider the results indicated in Figure 3.8, 3.9 and 3.10, the RLS estimator can estimate the system parameter very well. The change of load does not have effect on estimation.

3.7 Practical DC Motor Example

The experiment for dc motor estimation is executed here. The experiment is divided into two parts of simple case and load changing case as follows.

3.7.1 Simple case

In this example, the RLS estimator identifies the dc motor parameter in case of constant load inertia of motor at J = 0.6 x 10^{-4} kgm^2 and J = 3.0 x 10^{-4} kgm^2. The sampling time is selected as 0.005 sec.

Experimental Result:

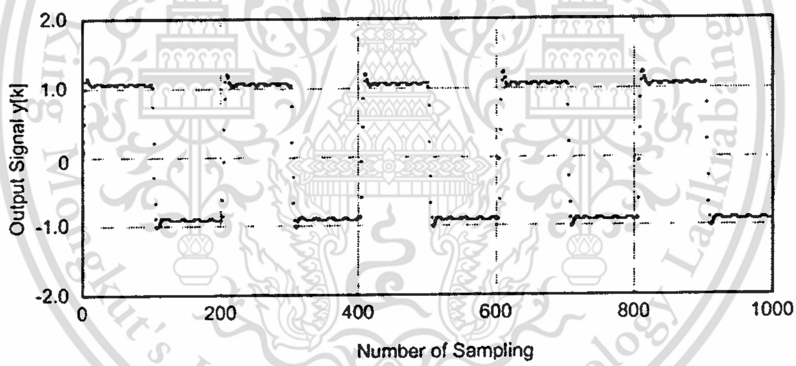

Figure 3.11 Plant output of the motor with J = 0.6 x 10^{-4} kgm^2.

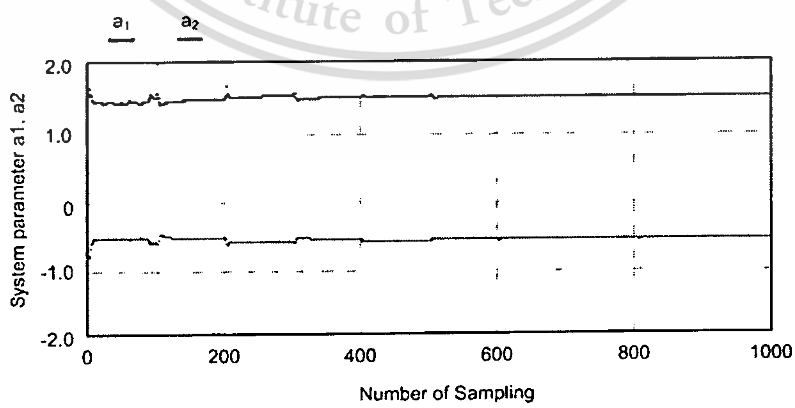

Figure 3.12 Estimated plant denominator $a_1 = 1.339, a_2 = -0.5291$ (J = 0.6 x 10^{-4} kgm^2).

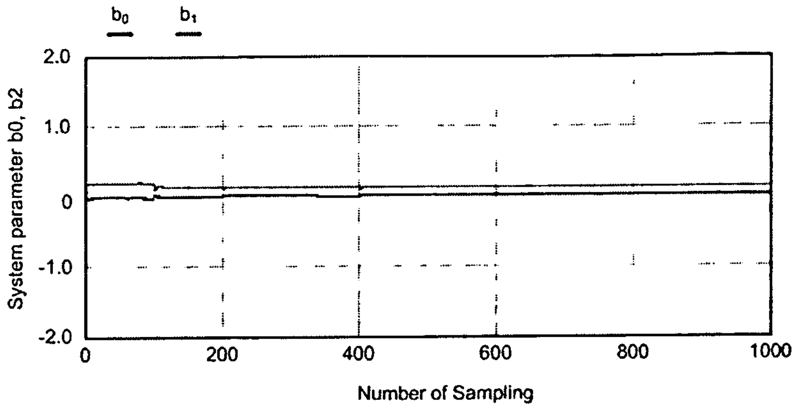

Figure 3.13 Estimated plant numerator $b_0 = 0.1154, b_1 = 0.1769$ (J = 0.6 x 10^{-4} kgm^2).

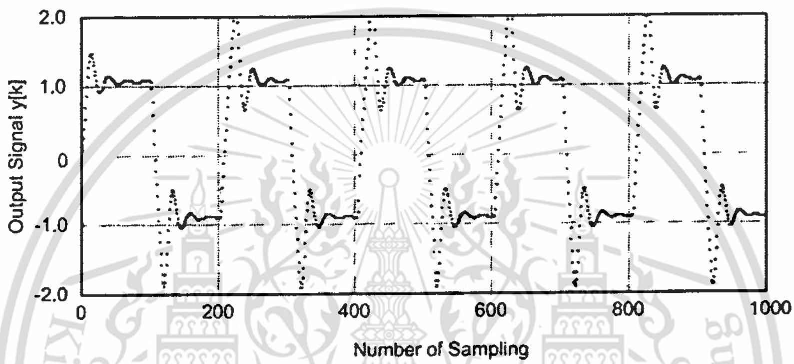

Figure 3.14 Plant output of the motor with J = 3.0 x 10^{-4} kgm^2.

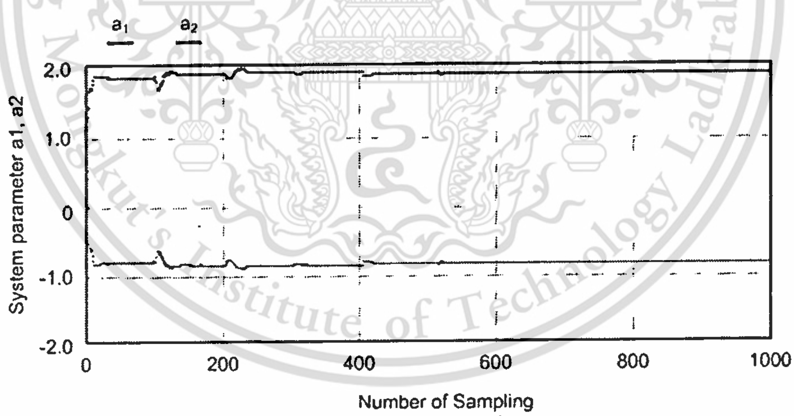

Figure 3.15 Estimated plant denominator $a_1 = 1.7757, a_2 = -0.8370$ (J = 3.0 x 10^{-4} kgm^2).

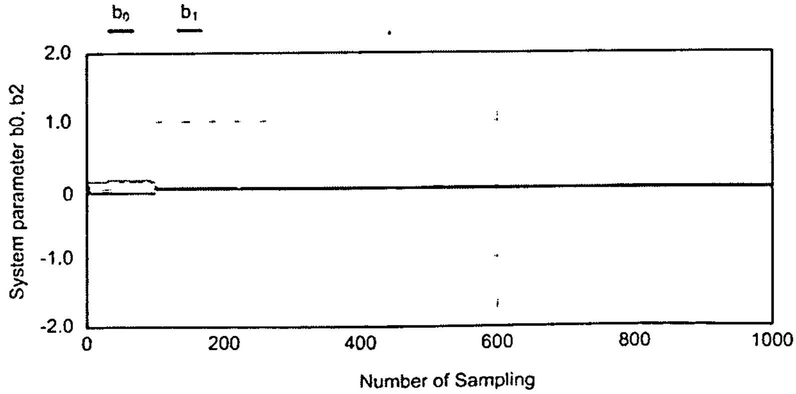

Figure 3.16 Estimated plant numerator $b_0 = 0.0045, b_1 = 0.0558$ (J = 3.0 x 10^{-4} kgm^2).

3.7.2 Plant Parameter Change

In this example, the RLS estimator identifies the dc motor parameter in case of the load inertia of motor changing from $J = 0.6 \times 10^{-4}$ kgm^2 to $J = 3.0 \times 10^{-4}$ kgm^2 at the sampling number of 500. The sampling time is selected as 0.005 sec.

Experimental Result:

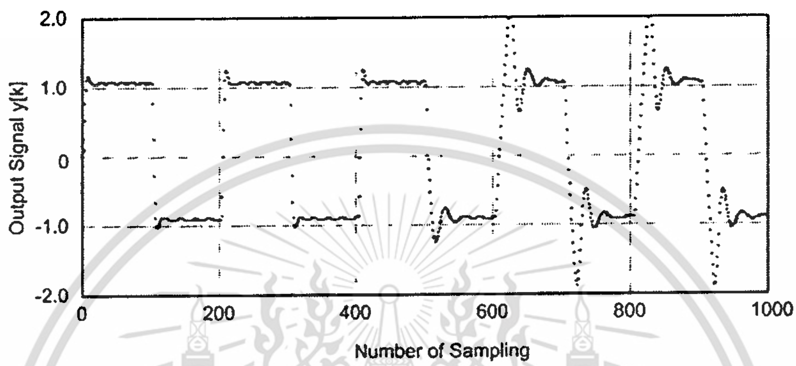

Figure 3.17 Plant output.

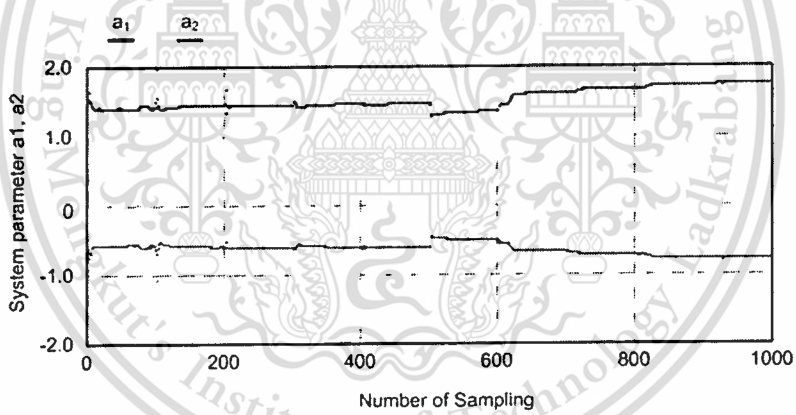

Figure 3.18 Estimated plant denominator a_1, a_2 .

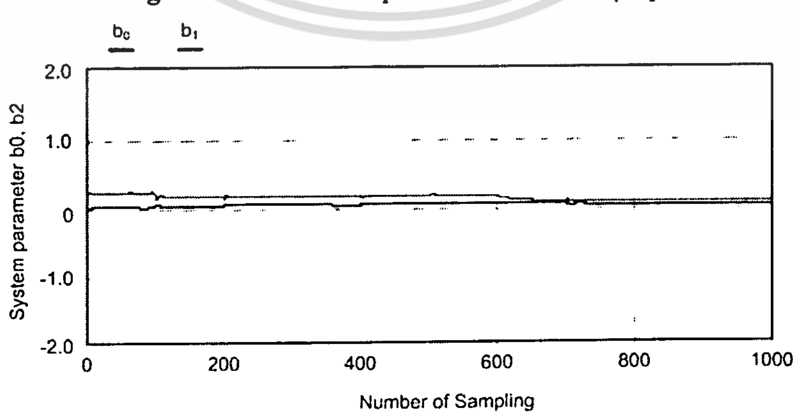

Figure 3.19 Estimated plant numerator b_0, b_1.

As comparing the value of estimated parameters in Figure 3.17 and 3.18, they can go to the correct value after the load changed (refer to the system parameters in section 3.7.1).

3.8 Conclusion

In this chapter a recursive system identification algorithms has been derived. The way to start and run a recursive estimator includes:

Starting up the estimator.

Initial design choices required to set up the estimation model and data collection procedure.

The experimental conditions which may be encountered and the various signal conditioning procedures which can be used.

The modifications required for a recursive estimator to track parameter changes.

The guideline provided here is general useful in estimation. Section 3.6 shows the application of RLS algorithm to the parameter identification in both simple case and changing parameter case. In addition, the RLS estimation is applied to the real dc motor in Section 3.7.

Now the concept and using of RLS estimation have been derived. The next chapter will discuss the analysis in the state space which plays another important role in the research.

Chapter 4

Analysis of Control System in State Space

As the statement in chapter 1, introduction, the modern control theory is used as the main concept of adaptive control system in this thesis. So the analysis of modern control will be discussed briefly in this chapter. Since our study involves the digital signal processing, it is convenient to place the discrete time model instead of continuous time one in every aspects of its property (such as controllability).

A modern system may have many inputs and many outputs and they may be interrelated in a complicated manner. To analyze such a system, the state space is the best suite approach.

4.1 State Space Representations of Transfer Function

This section will present the state space representation in the controllable and observable canonical form. The way to obtain this is shown as follows:

4.1.1 Controllable Canonical Form

To begin with, consider a system defined by

$$y(k)+a_1 y(k-1)+\cdots+a_{n_a} y(k-n_a)=b_0 u(k)+b_1 u(k-1)+\cdots+b_{n_b} u(k-n_b) \tag{4.1}$$

where u is the input and y is the output. Equation (4.1) can also be written by z transform as

$$\frac{Y(z)}{U(z)}=\frac{b_0+b_1 z^{-1}+\cdots+b_{n_b} z^{-n_b}}{1+a_1 z^{-1}+\cdots+a_{n_a} z^{-n_a}}=\frac{A\left(z^{-1}\right)}{B\left(z^{-1}\right)} \tag{4.2a}$$

or

$$\frac{Y(z)}{U(z)}=\frac{b_0 z^{n_b}+b_1 z^{n_b-1}+\cdots+b_{n_b}}{z^{n_a}+a_1 z^{n_a-1}+\cdots+a_{n_a}}=\frac{A\left(z^{-1}\right)}{B\left(z^{-1}\right)} \tag{4.2b}$$

Rearrange Equation (4.2) by separating it into two parts and assume $n_a=n_b=n$, then

$$\frac{Y(z)}{U(z)} = b_0 + \frac{(b_1 - a_1 b_0)z^n + (b_2 - a_2 b_0)z^{n-1} + \cdots + (b_2 - a_2 b_0)}{z^n + a_1 z^{n-1} + \cdots + a_n} \tag{4.3}$$

which can be modified to

$$Y(z) = b_0 U(z) + \hat{Y}(z) \tag{4.4}$$

That is

$$Y(z) = b_0 U(z) + \frac{(b_1 - a_1 b_0)z^n + (b_2 - a_2 b_0)z^{n-1} + \cdots + (b_2 - a_2 b_0)}{z^n + a_1 z^{n-1} + \cdots + a_n} U(z) \tag{4.5}$$

Rewrite two last equations, Equation (4.4) and (4.5) in the following form:

$$\frac{\hat{Y}(z)}{(b_1 - a_1 b_0)z^n + (b_2 - a_2 b_0)z^{n-1} + \cdots + (b_2 - a_2 b_0)} = \frac{U(z)}{z^n + a_1 z^{n-1} + \cdots + a_n} = Q(z) \tag{4.6}$$

Then, from this last equation, the following equations are obtained:

$$\hat{Y}(z) = (b_1 - a_1 b_0)z^n Q(z) + (b_2 - a_2 b_0)z^{n-1} Q(z) + \cdots + (b_2 - a_2 b_0)Q(z) \tag{4.7}$$

$$z^n Q(z) = -a_1 z^{n-1} Q(z) - \cdots - a_n Q(z) + U(z) \tag{4.8}$$

Now define state variables as follows:

$$\begin{aligned} X_1(z) &= Q(z) \\ X_2(z) &= zQ(z) = zX_1(z) \\ &\vdots \\ X_{n-1}(z) &= z^{n-2}Q(z) = zX_{n-2}(z) \\ X_n(z) &= z^{n-1}Q(z) = zX_{n-1}(z) \end{aligned}$$

They are rewritten as

$$
\begin{gathered}
x_1(k+1) = x_2(k) \\
x_2(k+1) = x_3(k) \\
\vdots \\
x_{n-2}(k+1) = x_{n-1}(k) \\
x_{n-1}(k+1) = x_n(k)
\end{gathered}
\tag{4.9}
$$

Noting that $z^n Q(z) = zX_n(z)$, Equation (4.8) can be rewritten as

$$
zX_n(z) = -a_1 X_{n-1}(z) - \cdots - a_n X_1(z) + U(z) \tag{4.10a}
$$

or

$$
x_n(k+1) = -a_1 x_{n-1}(k) - \cdots - a_n x_1(k) + u(k) \tag{4.10b}
$$

And in the way, from Equation (4.4) and (4.7) we obtain

$$
Y(z) = b_0 U(z) + (b_1 - a_1 b_0) X_n(z) + (b_2 - a_2 b_0) X_{n-1}(z) + \cdots + (b_n - a_n b_0) X_1(z) \tag{4.11a}
$$

$$
y(k) = b_0 u(k) + (b_1 - a_1 b_0) x_n(k) + (b_2 - a_2 b_0) x_{n-1}(k) + \cdots + (b_n - a_n b_0) x_1(k) \tag{4.11b}
$$

Combine Equation (4.9), (4.10b) and (4.11b) to form the state space representation in the controllable canonical form as indicated as:

$$
\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \vdots \\ x_{n-2}(k+1) \\ x_{n-1}(k+1) \end{bmatrix}
=
\begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
0 & 0 & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & & \vdots \\
0 & 0 & 0 & \cdots & 1 \\
-a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1
\end{bmatrix}
\begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix}
+
\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(k)
\tag{4.12}
$$

$$
y(k) = \begin{bmatrix} b_n - a_n b_0 & b_{n-1} - a_{n-2} b_0 & \ldots & b_2 - a_2 b_0 & b_1 - a_1 b_0 \end{bmatrix}
\begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix}
+ b_0 u(k)
\tag{4.13}
$$

4.1.2 Observable Canonical Form

Equation (4.2a) can be modified into the following form with $n_a = n_b = n$:

$$Y(z) = b_0 U(z) + z^{-1}\left(b_1 U(z) - a_1 Y(z)\right) + \ldots + b_n z^{-n} U(z) - a_n z^{-n} Y(z) \tag{4.14}$$

Define state variables as follow:

$$\begin{aligned}
X_n(z) &= \frac{1}{z}\left[b_1 U(z) - a_1 Y(z) + X_{n-1}(z)\right] \\
X_{n-1}(z) &= \frac{1}{z}\left[b_2 U(z) - a_2 Y(z) + X_{n-2}(z)\right] \\
&\vdots \\
X_2(z) &= \frac{1}{z}\left[b_{n-1} U(z) - a_{n-1} Y(z) + X_1(z)\right] \\
X_1(z) &= \frac{1}{z}\left[b_n U(z) - a_n Y(z)\right]
\end{aligned} \tag{4.15}$$

Then Equation (4.14) can be written as

$$Y(z) = b_0 U(z) + X_n(z) \tag{4.16}$$

By substituting Equation (4.16) into Equation (4.15) and multiplying both sides by s, we obtain

$$\begin{aligned}
zX_n(z) &= X_{n-1}(z) - a_1 X_n(z) + \left[b_1 - a_1 b_0\right] U(z) \\
zX_{n-1}(z) &= X_{n-2}(z) - a_2 X_n(z) + \left[b_2 - a_2 b_0\right] U(z) \\
&\vdots \\
zX_2(z) &= X_1(z) - a_{n-2} X_n(z) + \left[b_{n-1} - a_{n-1} b_0\right] U(z) \\
zX_1(z) &= -a_n X_n(z) + \left[b_n - a_n b_0\right] U(z)
\end{aligned} \tag{4.17}$$

Taking the inverse z transforms of the preceding n equations and writing them in the reverse order, we get

$$
\begin{aligned}
x_n(k+1) &= x_{n-1}(k) - a_1 x_n(k) + [b_1 - a_1 b_0]u(k) \\
x_{n-1}(k+1) &= x_{n-2}(k) - a_2 x_n(k) + [b_2 - a_2 b_0]u(k) \\
&\vdots \\
x_2(k+1) &= x_1(k) - a_{n-2} x_n(k) + [b_{n-1} - a_{n-1} b_0]u(k) \\
x_1(k+1) &= -a_n x_n(k) + [b_n - a_n b_0]u(k)
\end{aligned} \tag{4.18}
$$

Also, the inverse z transform of Equation (4.16)

$$
y(k) = b_0 u(k) + x_n(k) \tag{4.19}
$$

Rewriting the state and output equations in the standard vector matrix forms gives Equations (4.20) and (4.21). This is the observable canonical form:

$$
\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \vdots \\ x_{n-2}(k+1) \\ x_{n-1}(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 & -a_n \\ 1 & 0 & \cdots & 0 & -a_{n-1} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & -a_2 \\ 0 & 0 & \cdots & 0 & -a_1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix} + \begin{bmatrix} b_n - a_n b_0 \\ b_{n-1} - a_{n-2} b_0 \\ \vdots \\ b_2 - a_2 b_0 \\ b_1 - a_1 b_0 \end{bmatrix} u(k) \tag{4.20}
$$

$$
y(k) = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix} + b_0 u(k) \tag{4.21}
$$

Both controllable and observable canonical forms of state space representation can be expressed the discrete time state equation as:

$$
\begin{aligned}
\boldsymbol{x}[k+1] &= \boldsymbol{A}_d \boldsymbol{x}[k] + \boldsymbol{B}_d \boldsymbol{u}[k] \\
\boldsymbol{y}[k] &= \boldsymbol{C}_d \boldsymbol{x}[k] + \boldsymbol{D}_d \boldsymbol{u}[k]
\end{aligned} \tag{4.22}
$$

4.2 Solution of Discrete-Time Equations

In order to compute the solution of discrete time state equation in Equation (4.22), the number k is defined as integer and greater than zero.

$$x[1] = A_d x[0] + B_d u[0]$$

$$x[2] = A_d x[1] + B_d u[1] = A_d^2 x[0] + A_d B_d u[0] + B_d u[1]$$

Proceeding forward to obtain

$$x[k] = A_d^k x[0] + \sum_{m=0}^{k-1} A_d^{k-1-m} B_d u[m] \tag{4.23a}$$

$$y[k] = C_d A_d^k x[0] + \sum_{m=0}^{k-1} C_d A_d^{k-1-m} B_d u[m] + D_d u[k] \tag{4.23b}$$

4.3 Controllability

The discrete time state equation, Equation (4.22), or the matrix pair (A,B) is said to be controllable, if for any initial state $x[0] = x_0$ and any final state x1, there exists an input sequence of finite length that transfer x0 to x1. Otherwise the equation or (A,B) is said to be uncontrollable.

The following statements are equivalent for controllability:

1. The n dimensional pair **(A,B)** is controllable.
2. The n x n matrix

$$W_{dc}[k-1] = \sum_{m=0}^{k-1} A_d^m B_d B_d^T \left(A_d^T\right)^m \tag{4.24}$$

is nonsingular.

3. The n x np controllability matrix

$$C = \begin{bmatrix} B_d & A_d B_d & A_d^2 B_d & \cdots & A_d^{n-1} B_d \end{bmatrix} \tag{4.25}$$

has rank n (full row rank).

4. The n x (n + p) matrix $[A_d - \lambda I \quad B_d]$ has full row rank at every eigenvalue, λ , of A.
5. If, in addition, all eigenvalues of A have magnitudes less than 1, then the unique solution of

$$W_{dc} - A_d W_{dc} A_d^T = B_d B_d^T \tag{4.26}$$

is positive definite. The solution is called the discrete controllability Gramian. The discrete Gramian can be expressed as

$$W_{dc} = \sum_{m=0}^{\infty} A_d^m B_d B_d^T \left(A_d^T\right)^m \tag{4.27}$$

There is one important difference between the continuous and discrete time cases. If a continuous state equation is controllable, the input can transfer any state to any other state in any nonzero time interval. If a discrete time state equation is controllable, an input sequence of length $n \geq \mu$ can transfer any state to any other state. If an input sequence is shorter than μ , it is not possible to transfer any state to any other state. μ is called the controllability index and defined as

$$\mu = \max\{\mu_1, \mu_2, \mu_3, \ldots, \mu_p\} \tag{4.28}$$

Whereas p is the number of input and μ_m is the number of linearly independent column associated with b_m in controllability matrix C . It is clear that if C has rank n, then $\mu_1 + \mu_2 + \mu_3 + \ldots + \mu_p = n$

4.4 Observability

The discrete time state equation, Equation (4.22), or the matrix pair (A,C) is said to be observable, if for any unknown initial state x[0] = x_0 and any final state x_1, there exists an finite integer $k_1 > 0$ such that the knowledge of the input sequence $u(k)$ and output $y(k)$ from $k = 0$ to $k = k_1$ suffices to determine uniquely the initial state x[0]. Otherwise the equation or (A,B) is said to be unobservable.

The following statements are equivalent for observability determination.

1. The n dimensional pair (**A,C**) is observable.
2. The n x n matrix

$$W_{do}[k-1]=\sum_{m=0}^{k-1}\left(A_d^T\right)^m C_d^T C_d A_d^m \tag{4.24}$$

   is nonsingular or positive definite.
3. The nq x n observability matrix

$$O=\begin{bmatrix} C_d \\ C_d A_d \\ C_d A_d^2 \\ \vdots \\ C_d A_d^{n-1} \end{bmatrix} \tag{4.25}$$

   has rank n (full row rank).
4. The (n + q) x n matrix

$$\begin{bmatrix} A_d-\lambda I \\ C_d \end{bmatrix}$$

   has full column rank at every eigenvalue, λ, of A.
5. If, in addition, all eigenvalues of A have magnitudes less than 1, then the unique solution of

$$W_{do}-A_d^T W_{do} A_d = C_d^T C_d \tag{4.26}$$

   is positive definite. The solution is called the discrete controllability Gramian. The discrete Gramian can be expressed as

$$W_{do}=\sum_{m=0}^{\infty}\left(A_d^T\right)^m C_d^T C_d A_d^m \tag{4.27}$$

All other properties for the continuous time case can be applied to the discrete time case without any modifications.

The advantageous uses of controllability and observability can be summarized as definitions for transformation of state space representation as follows:

1. The system in Equation (4.22) can be transformed by a nonsingular linear transformation $z = Tx$, to the controllable form (4.12), if and only if system is completely controllable. If this condition is true, T is uniquely defined by

$$T = \begin{bmatrix} \tau \\ \tau A \\ \tau A^2 \\ \vdots \\ \tau A^{n-1} \end{bmatrix} \tag{4.28}$$

where τ is the unique solution of

$$\tau\begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \tag{4.29}$$

2. There exists a nonsingular transformation $z = Tx$ taking Equation (4.22) to (4.12) with $z_1 = y = Cx$, if and only if the system is observable and

$$C\begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \tag{4.30}$$

And then, T is unique and is the matrix in Equation (4.28) with $\tau = C$.

3. The system in Equation (4.22) is observable if and only if the dual system is controllable. The system in Equation (4.22) is controllable if and only if the dual system is observable. The dual system of Equation (4.22) is defined as

$$\begin{aligned} x[k+1] &= A^T x[k] + C^T u[k] \\ y[k] &= B^T x[k] + Du[k] \end{aligned} \tag{4.31}$$

The subscript d has been dropped since only the discrete time case is referred.

4.5 Controllability to the Origin and Reachability

In the literature, there are three different controllability definitions:

1. Transfer any state to any other state.
2. Transfer any state to the zero state, called controllability to origin.
3. Transfer the zero state to any state, called controllability to the origin or reachability.

In the discrete time case, if A is nonsingular, the three definitions are equivalent. But if A is singular, only (1) and (3) are equivalent. This means sometimes when A is singular the system is not reachable but still controllable to the origin.

4.6 Controllability after Sampling

Consider a continuous time state equation

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \qquad (4.32)$$

If the input is piecewise constant or

$$u[k] := u(kT_s) = u(t) \qquad \text{for } kT_s \le t < (k+1)T_s \qquad (4.33)$$

then the equation can be described in Equation (4.22), as developed in Appendix.

$$\begin{aligned} x[k+1] &= A_d x[k] + B_d u[k] \\ y[k] &= C_d x[k] + D_d u[k] \end{aligned} \qquad (4.22 \text{ bis})$$

With

$$A_d = e^{AT_s} \ , \ B_d = \left(\int_0^{T_s} e^{At} dt \right) B \ , \ C_d = C \ , \ D_d = D \qquad (4.34)$$

Whether the discrete time Equation (4.22) is controllable or not depends on sampling interval T_s and the position of the eigenvalues of A. This can be indicated by following theories.

1. Suppose Equation (4.32) is controllable. A sufficient condition for its discretized equation, Equation (4.34), with sampling interval T_s to be controllable is that $\left|\mathrm{Im}[\lambda_i - \lambda_j]\right| \neq 2\pi m / T_s$ for $m = 1,2,3,\ldots$, whenever $\left|\mathrm{Re}[\lambda_i - \lambda_j]\right| = 0$. For the single input case, the condition is necessary as well.
2. If a continuous time linear state equation is not controllable, then its discretized state equation, with any sampling interval T_s, is not controllable.

The theorem (2) is apparent. If a state equation is not controllable using any input, it is, of course, not controliable using only piecewise constant input.

4.7 Conclusion

This Chapter considered the state space representation in different forms. The priciple concepts and some necessary properties of the state space (controllability and observerabilty) are also given. Subsequent Chapter will discuss the design of control involving these properties.

Chapter 5

Design of Control System in State Space

The concepts of controllability and observability were discussed in the previous chapter, in order to study the internal structure of systems and to establish the relationship between the internal and external descriptions. In this chapter their implications in the design of feedback control systems.

5.1 State Feedback

Consider the n dimensional single variable state equation

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k] \end{aligned} \tag{5.1}$$

where D is supposed being zero to simplify discussion. In state feedback, the input u is given by

$$u[k] = r[k] - kx[k] = r - [k_n \quad k_{n-1} \quad \ldots \quad k_1]x[k] \tag{5.2}$$

as indicated in figure 5.1. Each feedback gain is a real constant. This is called the constant gain negative state feedback or, only, state feedback. Substituting Equation (5.2) into Equation (5.1) yields

$$\begin{aligned} x[k+1] &= (A - Bk)x[k] + Br[k] \\ y[k] &= Cx[k] \end{aligned} \tag{5.3}$$

The pair (A-Bk, B) ,for any 1 x n real constant vector k, is controllable if and only if (A,B) is controllable. This can be easily understood from Equation (5.4).

The controllability matrix of open and closed loop system is defined by

$C = [B \quad AB \quad A^2B \quad \cdots \quad A^{n-1}B]$ and $C_f = [B \quad (A-Bk)B \quad (A-Bk)^2 B \quad \cdots \quad (A-Bk)^{n-1} B]$

respectively. Then

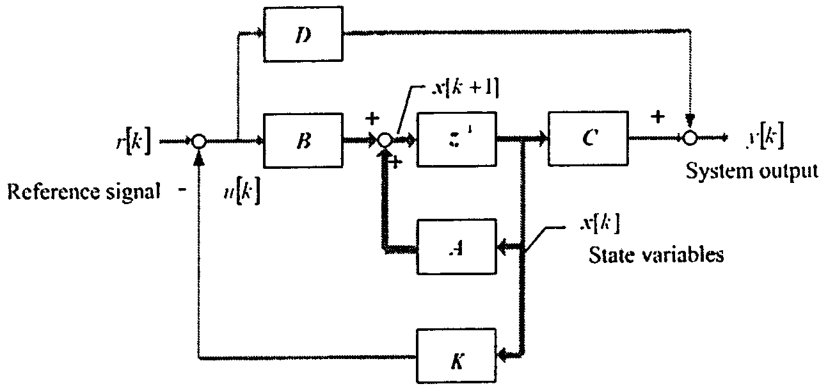

Figure 5.1 State feedback.

$$C_f = C\begin{bmatrix} 1 & -kB & \dots & -k(A-Bk)^2B \\ 0 & 1 & \dots & -k(A-Bk)B \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \tag{5.4}$$

Although the controllability property is invariant under any state feedback, the observability property is not.

The state feedback can be used to arbitrary place eigenvalues to any positions. Further more, the feedback gain can be computed by direct substation, if the state equation is controllable. For example, the system in controllable form in Equation (4.12) is written again as

$$x[k+1] = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \cdots & -a_1 \end{bmatrix} x[k] + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u[k]$$

$$y[k] = [b_n - a_n b_0 \quad b_{n-1} - a_{n-2} b_0 \quad \cdots \quad b_2 - a_2 b_0 \quad b_1 - a_1 b_0] x[k] + b_0 u(k) \tag{5.5}$$

If (A ,B) in Equation (5.3) is the same value as Equation (5.5) then

$$x[k+1]=\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n-k_n & -a_{n-1}-k_{n-1} & -a_{n-2}-k_{n-2} & \cdots & -a_1-k_1 \end{bmatrix} x[k]+\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u[k]$$

$$y[k]=\begin{bmatrix} b_n-a_nb_0 & b_{n-1}-a_{n-2}b_0 & \cdots & b_2-a_2b_0 & b_1-a_1b_0 \end{bmatrix} x[k]+b_0u(k) \tag{5.6}$$

Feedback Transfer Function: The transfer function of open loop system for state equation in Equation (5.5) is defined as

$$G(z)=C(zI-A)^{-1}B+d=\frac{b_0z^n+b_1z^{n-1}+\ldots+b_n}{z^n+a_1z^{n-1}+\ldots+a_n} \tag{5.7}$$

After state feedback, the state equation becomes Equation (5.6) and has the feedback transfer function as

$$G_f(z)=C(zI-A)^{-1}B+d=\frac{b_0z^n+b_1z^{n-1}+\ldots+b_n}{z^n+\bar{a}_1z^{n-1}+\ldots+\bar{a}_n} \tag{5.8}$$

where $\bar{a}_n=a_n+k_n$.

The numerator of Equation (5.7) and (5.8) are the same. In other words, state feedback does not affect on the zeroes of the plant transfer function. This is a general property of state feedback: feedback can shift the poles of a plant but has no effect on the zeros. So if one or more poles are shifted to coincide with any zeros, the state equation will be not observable.

5.2 Desired Eigenvalue Selection

As shown in last section, if the system that is controllable, then it is stabilizable (able to be stable by state feedback) and every eigenvalues of $x[k+1]=(A-Bk)x[k]$ can be assigned arbitrarily. Otherwise the number of poles of closed loop which can be shifted to any place equals to the rank of controllability matrix.

The problem is how to choose the eigenvalue. This section will discuss about the criterion to select the preferable eigenvalues. This depends on the performance criteria, such as response time and overshoot, used in design. One way to do this is find the state feedback that minimized the quadratic performance index called :

$$J = \sum_{k=0}^{\infty} \left(x_k^* Q x_k + x_k^* R x_k \right) T_s \tag{5.9}$$

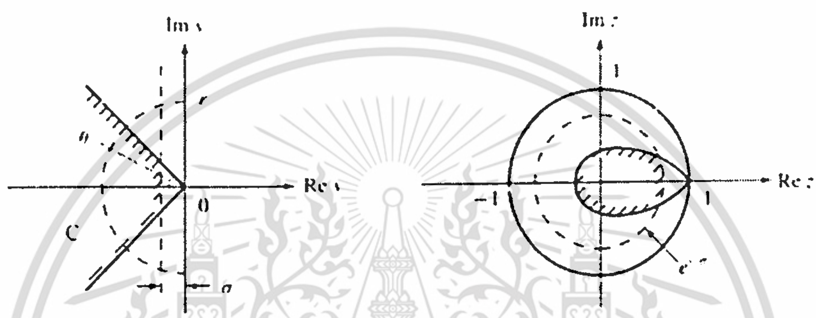

Figure 5.2 Desired eigenvalues location.

However this method, quadratic optimal control, requires much complex of computation. It would not be suitable for real time adaptive control. Another simpler way is to assign the feedback pole being in the region in figure 5.2 by comparing with the pole in continuous time as.

$$z = e^{T_s s} = e^{\mathrm{Re}[s]}\left[\cos(\mathrm{Im}[s]) + \sin(\mathrm{Im}[s]) j\right] \tag{5.10}$$

The speed of the response is reflected by the position of z on the real axis. That is the closer to unity position of z is, the slower the response. On the contrary, the closer to zero, the faster the response. And the frequency of oscillation in response is up to the position on complex axis.

5.3 Regulation and Tracking

5.3.1 Regulation Problem

Consider the system in figure 5.1. Suppose the reference signal is zero, and the system response is exited by some nonzero initial condition x0. The problem, called regulator problem, is to find a state feedback gain so that the response dies out at a desired rate. This problem can occur anytime while the control system is maintaining the plant's output at stable point. By that time if there exists some disturbances that make the response deviate from the needed point, this situation can be said to be the regulator problem.

As refer to state feedback, all discrete eigenvalues of the system can be shift to any position, the response caused by x0 can disappear at any rate. It is shown as

$$y[k] = C\overline{A}^k x[0] = C[A - Bk]^k x[0] \tag{5.11}$$

If all eigenvalues lie inside the unity circle, the effect of initial variables will decay to zero at the design rate. Thus the desired regulation can be easily achieved by introducing state feedback.

5.3.2 Tracking Problem

Another close related problem is the tracking problem. Suppose the reference signal is a constant or $r[k] = a$ for $k > 0$. The problem is to design an overall system so that $y[k]$ approaches $r[k] = a$ as $k \to \infty$. The tracking problem is slightly more complex. In general, in addition to state feedback, we need a feed-forward gain p as

$$u[k] = pr[k] - kx[k] \tag{5.12}$$

Then the transfer function for input $R(z)$ changes from Equation (5.8) to

$$G_f(z) = \frac{Y(z)}{R(z)} = p\frac{b_0 z^n + b_1 z^{n-1} + \ldots + b_n}{z^n + \overline{a}_1 z^{n-1} + \ldots + \overline{a}_n} \tag{5.13}$$

As $k \to \infty$, the output $y[k+n] = \ldots.. = y[k] = G_f(1)a$, for any constant reference signal. Therefore from Equation (5.13) the system gain is defined by

$$\left[z^n + \overline{a}_1 z^{n-1} + \ldots + \overline{a}_n\right]Y(z) = p\left[b_0 z^n + b_1 z^{n-1} + \ldots + b_n\right]R(z) \tag{5.14}$$

$$y[k+n] + \overline{a}_1 y[k+n-1] + \ldots + \overline{a}_n y[k] = p[b_0 + b_1 + \ldots + b_n]a$$

$$[1 + \overline{a}_1 + \ldots + \overline{a}_n]G_f(1)a = p[b_0 + b_1 + \ldots + b_n]a$$

$$G_f(1) = p\frac{b_0 + b_1 + \ldots + b_n}{1 + \bar{a}_1 + \ldots + \bar{a}_n} \tag{5.15}$$

So In order for $y[k]$ to track any step reference input (unity system gain), the feed forward gain is assigned as

$$1 = G_f(1) = p\frac{b_0 + b_1 + \ldots + b_n}{1 + \bar{a}_1 + \ldots + \bar{a}_n} \text{ or } p = \frac{1 + \bar{a}_1 + \ldots + \bar{a}_n}{b_0 + b_1 + \ldots + b_n} \tag{5.16}$$

which require every zeros z not equal to one or, equivalently, $b_0 + b_1 + \ldots + b_n \neq 0$. This will make system gain always being zero, and equivalently, tracking is not possible.

5.3.3 Robust Tracking and Disturbance Rejection

The state equation describing a plant may change due to change of load, development, or aging. Thus plant parameter variations often occur in practice. The equation used in design usually called the nominal equation. The feedforward gain p in Equation (5.12), computed for the nominal plant transfer function, may not give. Then the output will not track asymptotically any step reference input. Such a tracking is said to be nonrobust.

In this subsection a different design that can achieve robust tracking and disturbance rejection is discussed. Consider a plant in Equation (5.1). The constant disturbance is introduced into the system as shown in figure 5.3. With this disturbance, the state equation is modified as

$$\begin{aligned} \boldsymbol{x}[k+1] &= \boldsymbol{A}\boldsymbol{x}[k] + \boldsymbol{B}u[k] + \boldsymbol{B}v[k] \\ y[k] &= \boldsymbol{C}\boldsymbol{x}[k] \end{aligned} \tag{5.17}$$

To design the overall system so that the output y(t) will track asymptotically even with the presence of disturbance and plant parameter variation is called robust tracking and disturbance rejection. In order to achieve this design, in addition to introducing state feedback, we will introduce a discrete transfer function $(z-1)^{-1}$ with amplifying gain k_a and a unity feedback from output as shown in figure (5.3). Let the output from $(z-1)^{-1}$ be denoted by $x_a[k]$. Then the system has the new state vector $[x^T[k] \quad x_a[k]]^T$ and

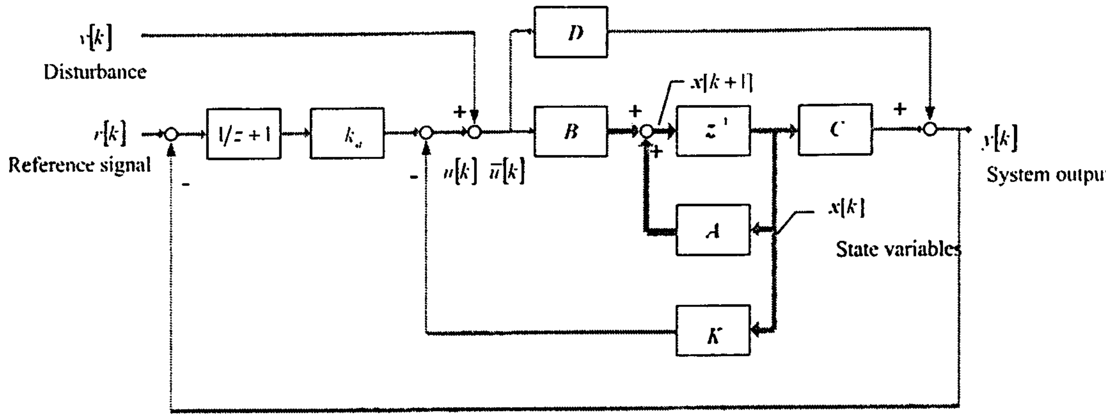

Figure 5.3 State feedback with robust tracking and disturbance rejection.

$$x_a[k+1]=r[k]-y[k]+x_a[k]=r[k]-Cx[k]+x_a[k] \tag{5.18}$$

$$u=\begin{bmatrix}k & k_a\end{bmatrix}\begin{bmatrix}x\\ x_a\end{bmatrix} \tag{5.19}$$

Introducing Equation (5.18) for new state variable $x_a[k]$ and substituting Equation (5.19) into Equation (5.17) gives

$$\begin{bmatrix}x[k+1]\\ x_a[k+1]\end{bmatrix}=\begin{bmatrix}A & 0\\ -C & 1\end{bmatrix}\begin{bmatrix}x[k]\\ x_a[k]\end{bmatrix}+\begin{bmatrix}B\\ 0\end{bmatrix}u[k]+\begin{bmatrix}0\\ 1\end{bmatrix}r[k]+\begin{bmatrix}B\\ 0\end{bmatrix}v[k] \tag{5.20}$$

$$\begin{bmatrix}x[k+1]\\ x_a[k+1]\end{bmatrix}=\begin{bmatrix}A-Bk & -Bk_a\\ -C & 1\end{bmatrix}\begin{bmatrix}x[k]\\ x_a[k]\end{bmatrix}+\begin{bmatrix}0\\ 1\end{bmatrix}r[k]+\begin{bmatrix}B\\ 0\end{bmatrix}v[k] \tag{5.21}$$

The eigenvalues of the A-matrix in Equation (5.20) can be assigned arbitrarily by selecting a feedback gain $\begin{bmatrix}k & k_a\end{bmatrix}$, if (A, B) is controllable and $G(z)=C(zI-A)^{-1}B+d$ has no zero at $z=1$. This can be proved by the close-loop system gain

$$G_f(z)=\frac{G(z)/(z-1)}{1+G(z)/(z-1)}=\frac{G(z)}{(z-1)+G(z)} \tag{5.22}$$

$$G_f(1)=\frac{G(1)}{((1)-1)+G(1)}=\frac{G(1)}{G(1)}=1 \tag{5.23}$$

Equation (5.23) indicates that the close-loop gain always equals one. But if $G(z)$ has one unity zero $z=1$, that is $G(z)=(z-1)\overline{G}(z)$, then the close-loop gain becomes

$$G_f(z)=\frac{(z-1)\overline{G}(z)}{(z-1)+(z-1)\overline{G}(z)}=\frac{\overline{G}(z)}{1+\overline{G}(z)} \tag{5.23}$$

Moreover, it can also be proved by the controllability matrix of the system in Equation (5.20), or the pair

$$\begin{bmatrix} A & 0 \\ -C & 1 \end{bmatrix} \begin{bmatrix} B \\ 0 \end{bmatrix}$$

The matrix can be modified by elementary operations, not affecting the rank of matrix.

$$C=\begin{bmatrix} 1 & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} \\ 0 & 1 & \cdots & a_{2,n-1} & a_{2,n} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & a_{n-1,n} \\ 0 & 0 & \cdots & 0 & [b_1+b_2+\ldots+b_n] \end{bmatrix} \tag{5.24}$$

From Equation (5.24) the rank of controllability matrix is n, if $b_0+b_1+\ldots+b_n\neq 0$. This means the transfer function $G(z)$ does not have its zero at $z=1$.

5.4 State Observer

The state feedback under the implicit assumption that all state variables are available or measurable has been introduced in the preceding section. This assumption may not hold in practice because of any measuring or accessing problem. In this case, in order to apply state feedback, the device generating an estimate of the state is design. It is called state observer. The full dimensional state observer with the estimated variable $\hat{x}[k]$ is indicated as

$$\hat{x}[k+1]=A\hat{x}[k]+Bu[k] \tag{5.25}$$

and as Figure 5.4 this observer can be called an open loop type. There are two disadvantages in using an open loop observer. First, the initial state must be corrected and set each time. Second, the error from disturbance or initial state setting can grow up if the matrix A has unstable poles. Therefore this observer is not satisfactory.

The observer can be modified from figure 5.4 to figure 5.5, the closed loop state observer. The output $y[k]=Cx[k]$ is compared with $\hat{y}[k]=C\hat{x}[k]$. Their difference passes through a constant matrix l to correct the estimated state variable. The open loop observer is modified as

$$\hat{x}[k+1]=A\hat{x}[k]+Bu[k]+l(y[k]-C\hat{x}[k]) \tag{5.26}$$

which can be written as

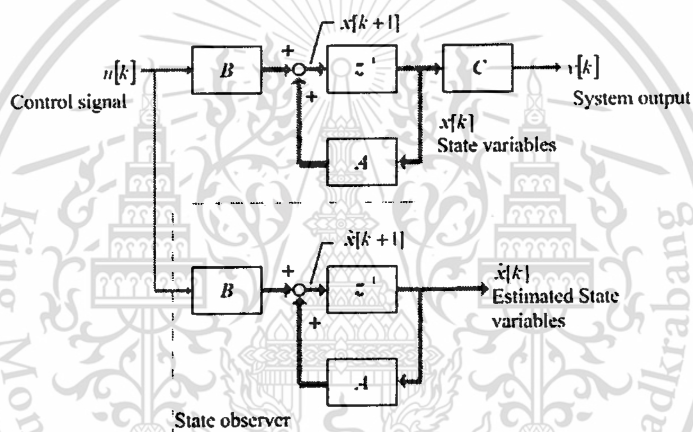

Figure 5.4 Open-loop state observer.

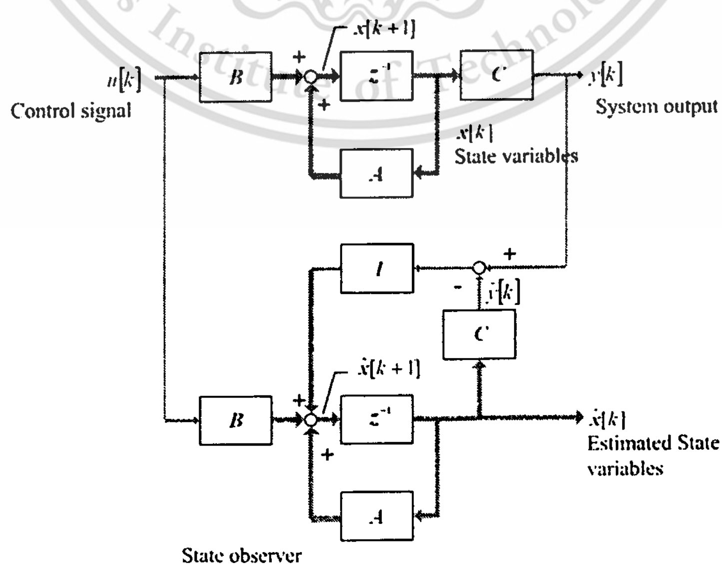

Figure 5.5 Closed-loop state observer 1.

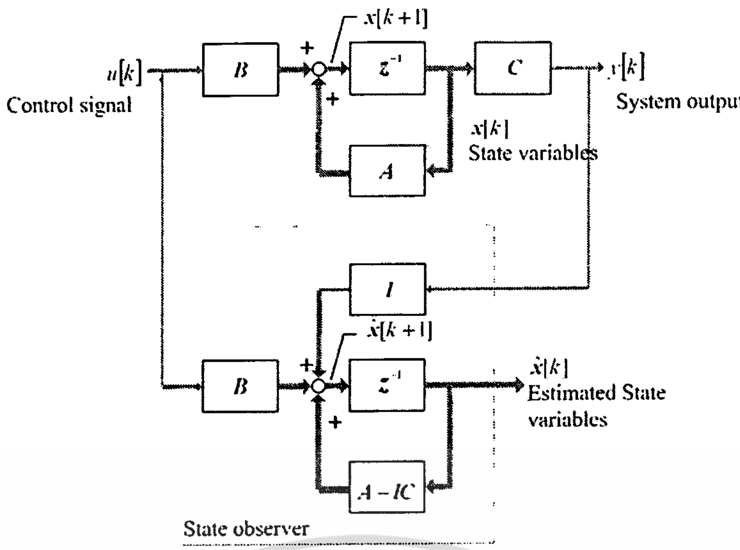

Figure 5.6 Closed-loop state observer 2.

$$\hat{x}[k+1]=(A-lC)\hat{x}[k]+Bu[k]+ly[k] \tag{5.27}$$

and is shown in figure 5.6. It has two inputs, $u[k]$ and $y[k]$, and output, an estimate state $\hat{x}[k]$. Defining the error between the real state and the estimated one yields

$$e[k]=x[k]-\hat{x}[k] \text{ and } e[k+1]=x[k+1]-\hat{x}[k+1] \tag{5.28}$$

By subtracting Equation (5.27) from (5.1), the equation of estimate error of the observer becomes

$$\begin{aligned} e[k+1]=x[k+1]-\hat{x}[k+1]&=[Ax[k]+Bu[k]]-[(A-lC)\hat{x}[k]+Bu[k]+lCx[k]] \\ &=(A-lC)x[k]-(A-lC)\hat{x}[k] \end{aligned}$$

or

$$e[k+1]=(A-lC)e[k] \tag{5.29}$$

If all eigenvalues of $(A-lC)$ can be assigned arbitrarily, then the rate for $e[k]$ to approach zero or the rate for the estimated state to approach the real state can be controlled. This can be the case if the system in Equation (5.1) or pair (A,C) is observable.

5.5 Feedback from Estimated State

If the real state is not available, the estimated state variables are used for feedback.

$$u[k] = r[k] - k\hat{x}[k] \tag{5.30}$$

This connection is called the controller estimator configulation. Considering the feedback system and the observer yields

$$\begin{aligned} x[k+1] &= Ax[k] - Bkx[k] + Br[k] \\ \hat{x}[k+1] &= (A - lC)\hat{x}[k] - Bkx[k] + Br[k] + lCx[k] \end{aligned} \tag{5.31}$$

They can be combined as:

$$\begin{bmatrix} x[k+1] \\ \hat{x}[k+1] \end{bmatrix} = \begin{bmatrix} A & -Bk \\ lC & A - lC - Bk \end{bmatrix} \begin{bmatrix} x[k] \\ \hat{x}[k] \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} r[k]$$

$$y[k] = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x[k] \\ \hat{x}[k] \end{bmatrix} \tag{5.32}$$

Then the system is transform by the transformation

$$\begin{bmatrix} x[k] \\ e[k] \end{bmatrix} = \begin{bmatrix} x[k] \\ x[k] - \hat{x}[k] \end{bmatrix} = \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \begin{bmatrix} x[k] \\ \hat{x}[k] \end{bmatrix} = P \begin{bmatrix} x[k] \\ \hat{x}[k] \end{bmatrix} \tag{5.33}$$

where

$$P = P^{-1} = \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \tag{5.34}$$

The state equation by the transformation $\begin{bmatrix} x[k] \\ e[k] \end{bmatrix} = P \begin{bmatrix} x[k] \\ \hat{x}[k] \end{bmatrix}$ is

$$P^{-1} \begin{bmatrix} x[k+1] \\ e[k+1] \end{bmatrix} = \begin{bmatrix} A & -Bk \\ lC & A - lC - Bk \end{bmatrix} P^{-1} \begin{bmatrix} x[k] \\ e[k] \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} r[k]$$

$$\begin{bmatrix} x[k+1] \\ e[k+1] \end{bmatrix} = P\begin{bmatrix} A & -Bk \\ lC & A-lC-Bk \end{bmatrix}P^{-1}\begin{bmatrix} x[k] \\ e[k] \end{bmatrix} + P\begin{bmatrix} B \\ B \end{bmatrix}r[k]$$

$$\begin{bmatrix} x[k+1] \\ e[k+1] \end{bmatrix} = \begin{bmatrix} A-Bk & Bk \\ 0 & A-lC \end{bmatrix}\begin{bmatrix} x[k] \\ e[k] \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix}r[k]$$

$$y[k] = \begin{bmatrix} C & 0 \end{bmatrix}\begin{bmatrix} x[k] \\ e[k] \end{bmatrix} \tag{5.35}$$

The system in Equation (5.35) has the eigenvalues of the union of those of $(A-Bk)$ and $(A-lC)$. Thus inserting the state observer does not affect the eigenvalues of the original state feedback. And the eigenvalues of the observer are not affected by this connection. Moreover, the transfer function of the system is still as same as that of the system fed back by the real state vector.

$$G_f(z) = C(zI - A + Bk)^{-1}B$$

5.6 Conclusion.

This chapter has considered the design of state feedback and the observer. Next chapter, these control theory will be included with the RLS estimator to establish the adaptive state feedback.

Chapter 6

Adaptive State Feedback Control in Computer Simulation

Before applying the new control approach to the real dc motor, the perfomance of control should be verified in computer simulation in order to sum up the problems and find out their solution. In this chapter the simulation program is made for this purpose.

At the beginning of the chapter, the description of computer simulation is presented. Then the control method is explained in the next two sections. For the last section the experiments are conducted on the important aspects of control.

6.1 Computer Simulation

As mentioned above, this section discusses the way to simulate the dynamic of dc motor. Since this program simulation is made on the basis of state equations, it consists of 2 main parts: modeling the system in state space and transforming it to discrete form.

6.1.1 Modeling in State Space

The equivalent circuit of a dc motor, the permanent-magnet type, is shown in Figure 6.1. In this figure, the armature circuit is illustrated with an ideal voltage source E_a, inductor L_a and resistor R_a. This presentation is really the Thevinin equivalent circuit of the rotor structure.
The internal generated voltage in this machine is given by equation

$$E_a = k_e \omega(t) \tag{6.1}$$

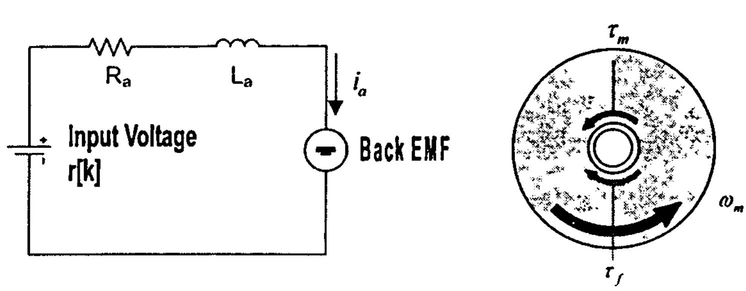

Figure 6.1 The equivalent circuit of DC motor and torque – acceleration relation.

The induced torque developed by the machine is given by

$$\tau_m = k_\tau i_a(t) \tag{6.2}$$

where k_e and k_τ are the back E.M.F constant and torque constant respectively. And, as the circuit in figure 6.1, the differential equation of the armature current $i_a(t)$ is

$$L_a \dot{i}_a(t) + R_a i_a(t) + E_a = V(t) \quad \text{,that can be rewritten as} \quad \dot{i}_a(t) = \frac{-R_a i_a(t) - E_a + V(t)}{L_a} \tag{6.4}$$

The angular acceleration of dc motor associated with the moment of inertia is given by

$$\dot{\omega}(t) = \frac{\sum \tau}{J} = \frac{\tau_m + \tau_f}{J_m + J_L} \tag{6.3}$$

Substituting Equation (6.1) into (6.3) and Equation (6.2) into (6.4), then

$$\dot{i}_a(t) = -\frac{1}{L_a} k_e \omega(t) - \frac{R_a}{L_a} i_a(t) + \frac{1}{L_a} V(t) \tag{6.5}$$

$$\dot{\omega}(t) = \frac{k_\tau}{J} i_a(t) + \frac{\tau_f}{J} \tag{6.6}$$

These last two equations can be combined together into a state space equation as

$$\begin{bmatrix} \dot{\omega}(t) \\ \dot{i}_a(t) \end{bmatrix} = \begin{bmatrix} 0 & \frac{k_\tau}{J} \\ \frac{-k_e}{L_a} & \frac{-R_a}{L_a} \end{bmatrix} \begin{bmatrix} \omega(t) \\ i_a(t) \end{bmatrix} + \begin{bmatrix} 0 & \frac{1}{J} \\ \frac{1}{L_a} & 0 \end{bmatrix} \begin{bmatrix} V(t) \\ \tau_f \end{bmatrix} \tag{6.5}$$

which has the state variable $x = [\omega(t) \quad i_a(t)]^T$ as and the input vector $u = [V(t) \quad \tau_f(t)]^T$ as . Since the output is the angular speed $\omega(t)$ so the output equation for state equation (6.7) is

$$y(t) = [1 \quad 0] \begin{bmatrix} \omega(t) \\ i_a(t) \end{bmatrix} \tag{6.8}$$

6.1.2 Discretization

Consider the solution of continuous time state equation.

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau \tag{6.9}$$

Suppose the input is generated by a digital to analog converter with the function of ZOH, then the input will be piecewise constant as

$$u(t) = u(kT) = u[k] \text{ for } kT \le t < (k+1)T \tag{6.10}$$

For $k = 0,1,2,\ldots$. This input changes the value only at the discrete time instants. For this input, the solution of Equation (6.9) at $t = kT$ and $t = (k+1)T$ becomes

$$x[k] := x(kT) = e^{AkT}x(0) + \int_0^{kT} e^{A(KT-\tau)}Bu(\tau)d\tau \tag{6.11}$$

and

$$x[k+1] := x((k+1)T) = e^{A(k+1)T}x(0) + \int_0^{(k+1)T} e^{A((k+1)T-\tau)}Bu(\tau)d\tau \tag{6.12}$$

Equation (6.12) can be written as

$$x[k+1] = e^{AT}\left[e^{A(k+1)T}x(0) + \int_0^{kT} e^{A(kT-\tau)}Bu(\tau)d\tau\right] + \int_{kT}^{(k+1)T} e^{A(kT+T-\tau)}Bu(\tau)d\tau \tag{6.13}$$

After substituting Equation (6.10) and (6.11) into (6.13) and introducing the new variable $\alpha = kT + T - \tau$ the Equation (6.13) becomes

$$x[k+1] = e^{AT}x[k] + \left(\int_0^T e^{A\alpha}d\alpha\right)Bu(k) \tag{6.14}$$

Thus, the discrete time state equation of the system, an input changes value only at discrete time instants, is the form of Equation (4.22) that is

$$x[k+1] = A_d x[k] + B_d u[k]$$
$$y[k] = C_d x[k] + D_d u[k]$$

with $A_d = e^{AT_s}$, $B_d = \left(\int_0^{T_s} e^{At}dt\right)B$, $C_d = C$, $D_d = D$ as shown in Equation (3.4).

The computation of B_d can be modified as Equation (6.15), if and only if A is a nonsingular matrix.

$$B_d = A^{-1}(A_d - I)B \tag{6.15}$$

This discrete time state space can be computed its solution, or response, in computer due to its recursive operation.

6.2 Adaptive State Feedback and Observer

The system parameter from RLS estimator in form of Equation (2.2b) can be rearranged in the transfer function form as.

$$G(z^{-1}) = z^{-d} \frac{b_0 + b_1 z^{-1} + \ldots + b_{n_b} z^{-n_b}}{1 + a_1 z^{-1} + \ldots + a_{n_a} z^{-n_a}} \tag{6.16}$$

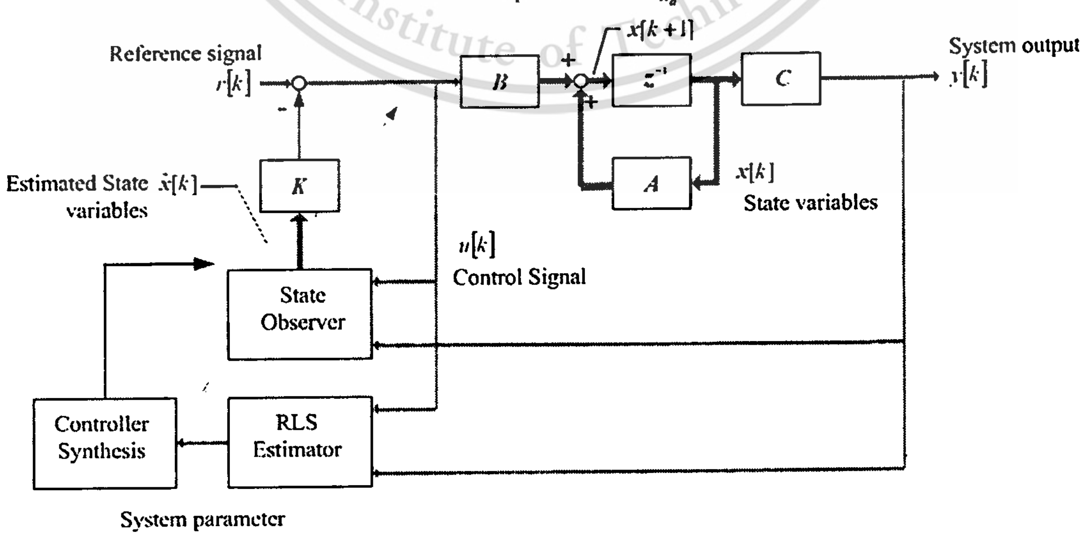

Figure 6.2 Adaptive state feedback controller.

where n_a, n_b and d are the estimator parameters properly chosen. And according to chapter 4, this transfer function can be transform into the state space representation as.

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ \vdots \\ x_{n_a-1}(k+1) \\ x_{n_a-1}(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_{n_a} & -a_{n_a-1} & -a_{n_a-2} & \cdots & -a_1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{n_a-1}(k) \\ x_{n_a}(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(k) \tag{6.17}$$

$$y(k) = \begin{bmatrix} b_{n_a} & b_{n_a-1} & \cdots & b_2 & b_1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_{n_a-1}(k) \\ x_{n_a}(k) \end{bmatrix} \tag{6.18}$$

where the coefficient b_0 is always zero because of the effect of ZOH. This state equation is used as the system reference and the model for the observer. The feedback gain $\boldsymbol{K}$ will be adapted according to this system parameter as shown in figure 6.2. At this point the feedback gain $\boldsymbol{K}$ is computed by

$$\begin{bmatrix} k_{n_a} & k_{n_a-1} & \ldots & k_1 \end{bmatrix} = \begin{bmatrix} \bar{a}_{n_a} - a_{n_a} & \bar{a}_{n_a-1} - a_{n_a-1} & \ldots & \bar{a}_1 - a_1 \end{bmatrix} \tag{6.19}$$

where the variable $\bar{a}_m$ is the coefficient of the closed loop polynomial which come from the desired closed loop poles. This can be easily understood that the feedback gain and the model of the observer are adapted to the estimated system. It is the reason to perform the experiments to test the accuracy of feedback as well as the observer. While, the observer gain l is set any real constant matrix.

6.2.1 Accuracy of Adaptive State Feedback

The state feedback is tested for accuracy in this section. To begin with, the estimated state vector is fed back from the observer through the feedback gain vector. This multiplied matrix becomes scalar and rounds up with the reference signal afterward.

The system transfer function is selected as $G(z^{-1}) = z^{-1}\dfrac{0.0562 + 0.0438z^{-1}}{1 - 1.4574z^{-1} + 0.4724z^{-2}}$

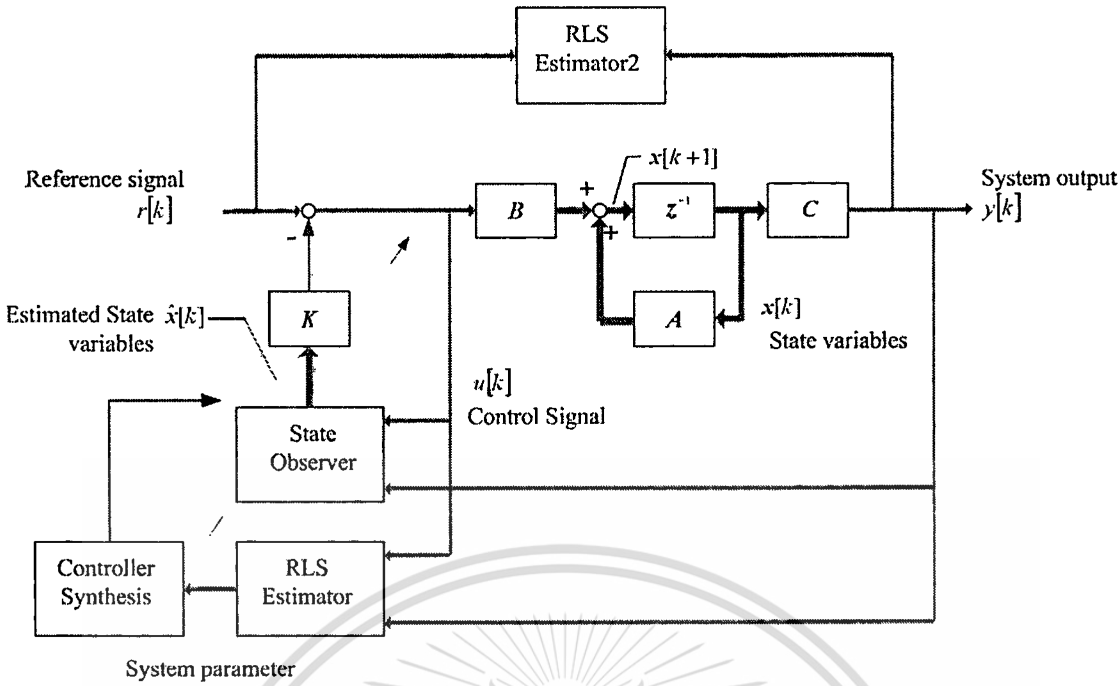

Figure 6.3 Block diagram of the accuracy test including 2 RLS estimators.

while the closed-loop polynomial is selected as

$$D(z^{-1}) = 1 - 1.77z^{-1} + 0.7812z^{-2}$$

The pulse signal with amplitude of 1 volt is supplied as the reference signal. There are two RLS estimators are used in this test, one for closed-loop system parameters, another one for open-loop system parameter.

Simulation results:

The results of this test are shown all together in Figure 6.4. These results indicate that the estimator, state observer, state feedback can do their responsibility exactly. Considering The estimated closed loop system parameter (Fig 6.4 (c)), feedback gain K (Fig 6.4 (a)), and open loop system parameter(Fig 6.4 (b)), they have the correct relation corresponding to state feedback theory. Moreover, the error between the actual output and observer's output decreases to zero within the first 50 samplings. This assures that the estimated state variables always equal to the real one after that.

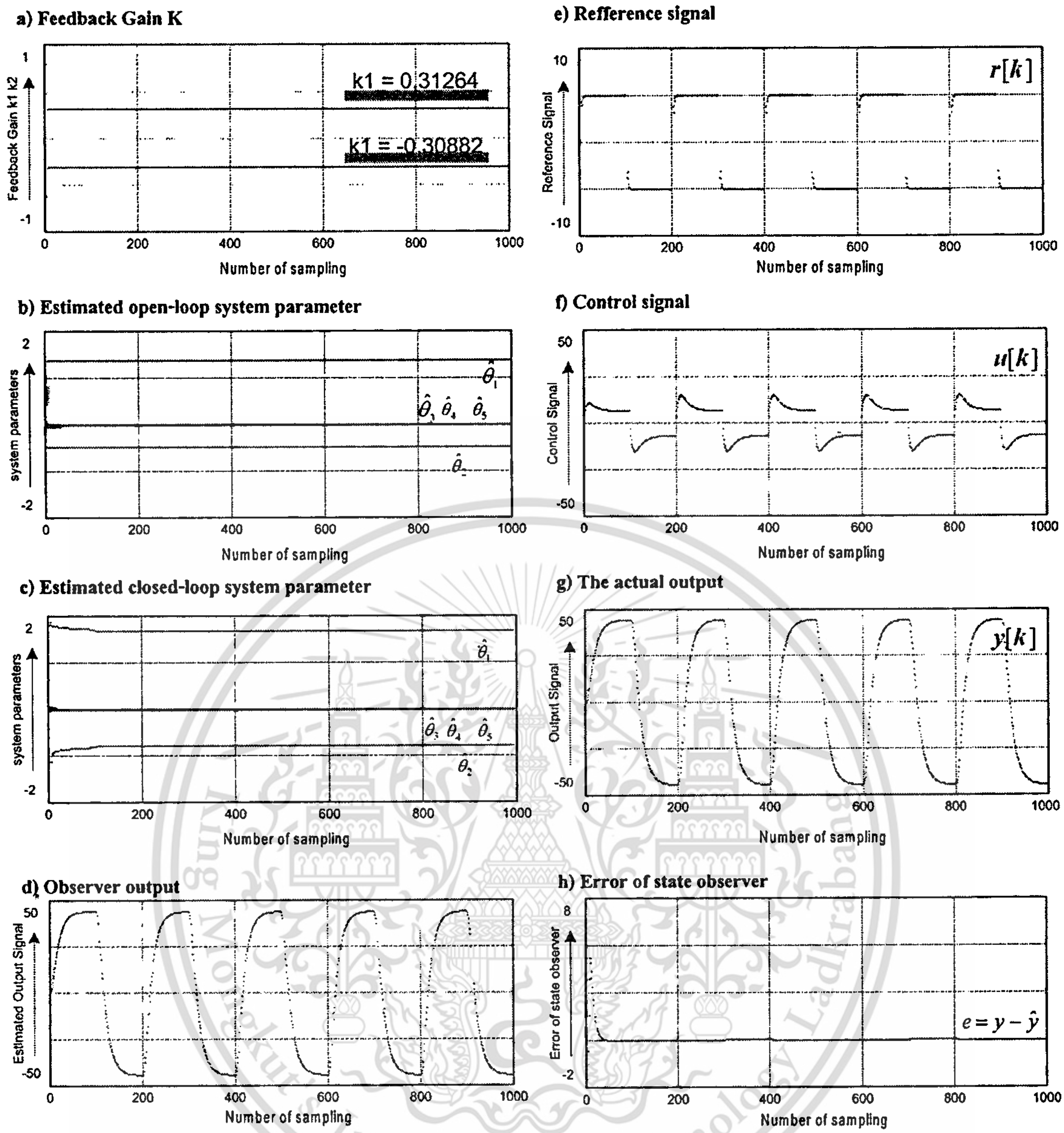

Figure 6.4 The experimental data of state feedback accuracy trial.

After that, in order to confirm the operation of state feedback, the control is adjusted for the other closed loop poles. Figure 6.5 shows that the response of system is faster when the closed loop poles are near zero (Figure 6.5 (a)) and slower when the closed loop poles approach unity (Figure 6.5 (d)).

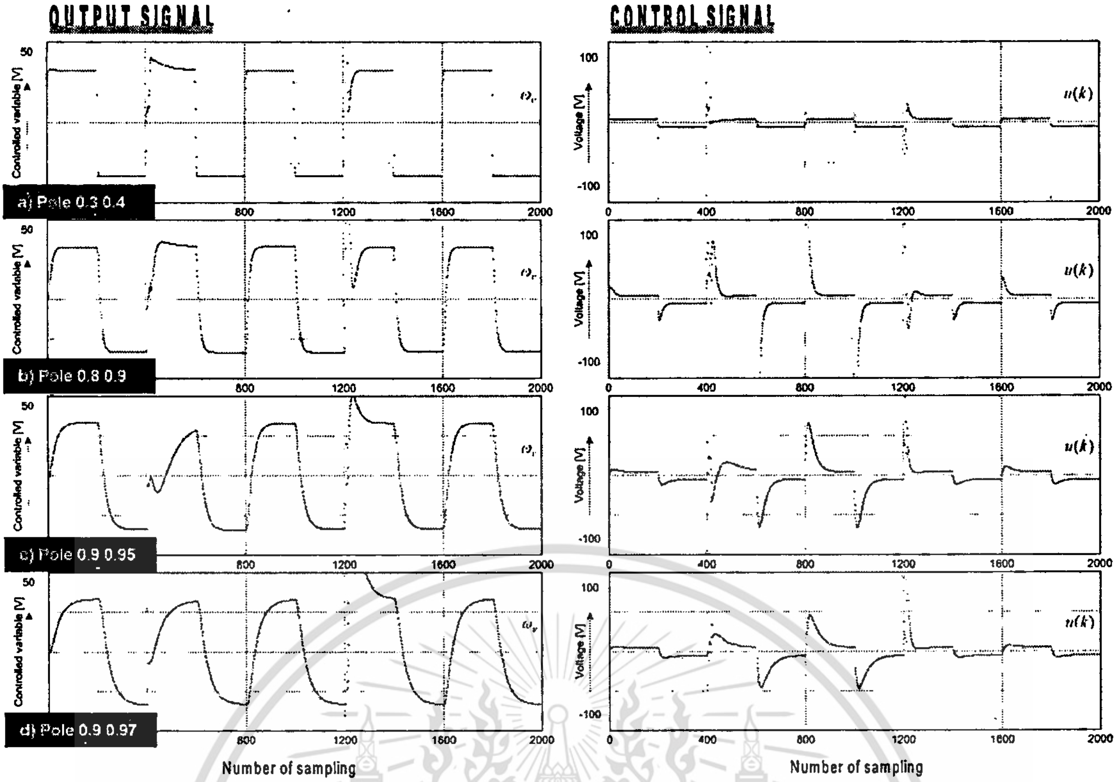

Figure 6.5 The experimental results of $y[k]$ and $u[k]$ of adaptive state feedback and feed-forward gain with different closed loop poles.

6.2.2 Adaptive Observer Gain l

In some case of very widely changing system, the fix rate of the error to approach zero may be necessary. From section 4, the system transfer function can be transform to the observable state equation as Equation (4.20) and (4.21). By this equation, it is possible to make the observer dispose of its error at any fix rate by means of adaptive observer gain l which can be adapted to the changing system. However, the quality to assign the poles of the control system is still needed. Therefore in order to establish this option, the properties of the controllability and observability must be obtained first.

This section will give the direct way to check the controllability and observability from the transfer function of the system. To simplify the discussion the system's order is 4 and the delay time is 0 then Equation (6.1) becomes

$$G(z^{-1}) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} + b_4 z^{-4}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} + a_4 z^{-4}} \tag{6.20a}$$

and the coprimness of the above fraction is defined by the matrix S.

$$S = \begin{bmatrix} a_4 & b_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_3 & b_3 & a_4 & b_4 & 0 & 0 & 0 & 0 \\ a_2 & b_2 & a_3 & b_3 & a_4 & b_4 & 0 & 0 \\ a_1 & b_1 & a_2 & b_2 & a_3 & b_3 & a_4 & b_4 \\ 1 & b_0 & a_1 & b_1 & a_2 & b_2 & a_3 & b_3 \\ 0 & 0 & 1 & b_0 & a_1 & b_1 & a_2 & b_2 \\ 0 & 0 & 0 & 0 & 1 & b_0 & a_1 & b_1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & b_0 \end{bmatrix} \qquad (6.20b)$$

That is the system is coprime or controllable and observable if the matrix S is nonsingular.

6.3 Reference Model for Tracking Problem

For the tracking problem, the reference model, called tracking model, is introduced in front of the closed loop system as shown in figure 6.6. It is desired that the system output follows the desired trajectory that is the output of tracking model. The tracking model can be determined by

$$y^*[k] = Gm(z^{-1})r[k] = z^{-(d+1)} \frac{Bm(z^{-1})}{Am(z^{-1})} r[k] \qquad (6.21)$$

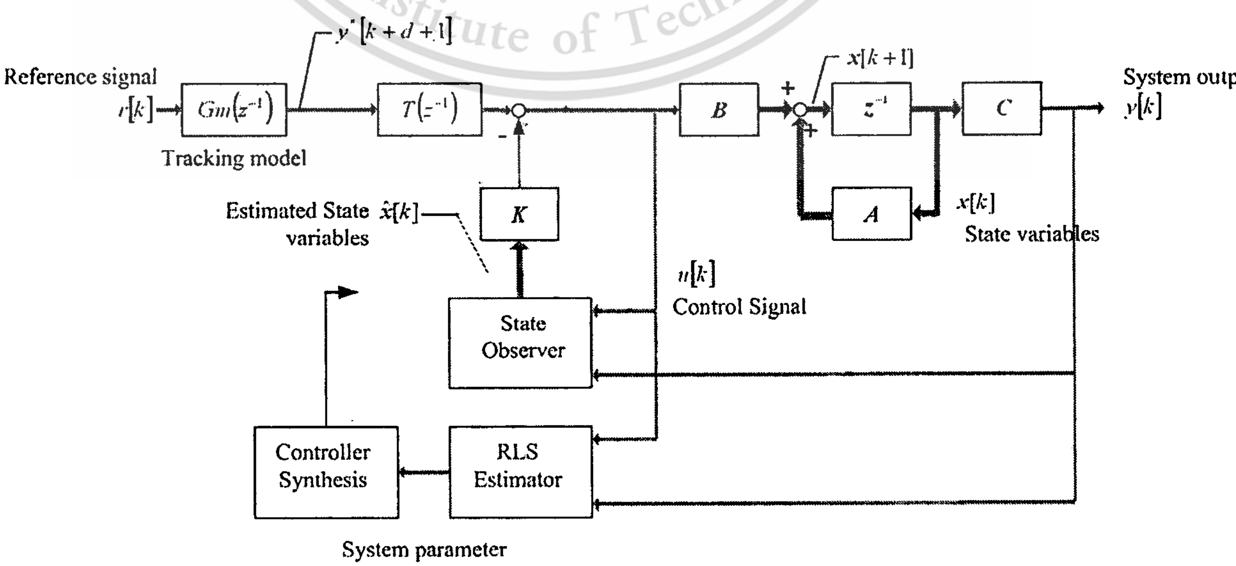

Figure 6.6 The adaptive state feedback control with tracking model and pre filter.

The coefficients of Equation (6.21) are defined corresponding to the satisfied performance (rise time and overshoot). The pre-filter $T(z^{-1})$ can be determined by

$$T(z^{-1}) = \frac{D(z^{-1})}{B(1)} \text{ or } T(z^{-1}) = \frac{D(1)}{B(1)} \tag{6.22}$$

where

$$D(z^{-1}) = 1 + \overline{\alpha}_1 z^{-1} + \cdots + \overline{\alpha}_{Na} z^{-Na} \tag{6.23}$$

6.4 Adaptive State Feedback Simulation Example

In this section, the adaptive state feedback control for dc motor is tested for its efficiency in the various issues: tracking, regulation and disturbance rejection. Therefore the experiments will comprise 3 tests as follows.

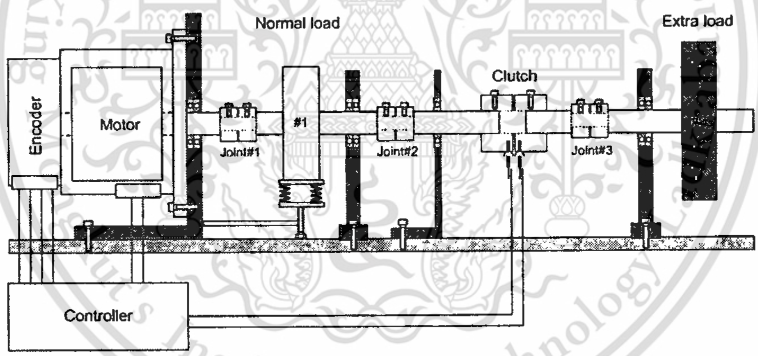

Figure 6.7 Dc motor system with electronic clutch is used to connect the extra load.

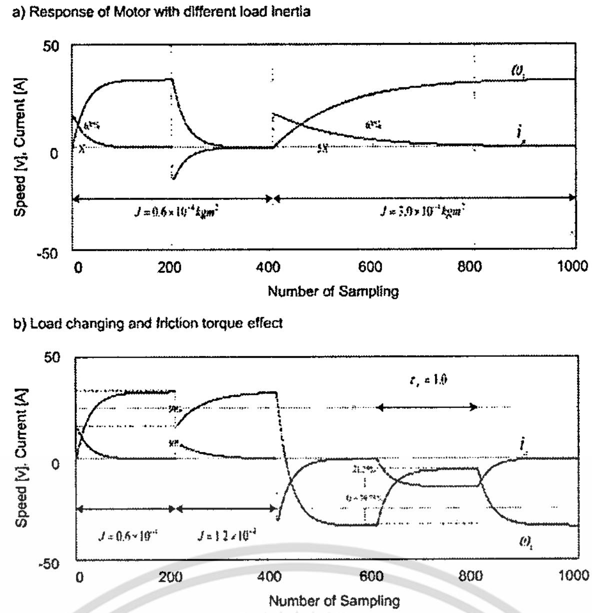

Figure 6.8 The speed and armature current of simulated dc motor under particular condition.

6.4.1 Tracking Efficiency Test

In order to check the tracking performance of control system, the response in transient must be sequently observed. So the reference signal of square wave form is supplied into the control system to excite the response during program running. In the same time the load inertia is changed by connecting the extra load. The dc motor system and its dynamic are shown in Figure 6.7 and 6.8, respectively.

The net moment of inertia switches between 0.6 x 10^{-4} kgm^2 and 3.0 x 10^{-4} kgm^2 at the 400^{th} and 1200^{th} sampling. The tracking and regulation (closed loop polynomial) are set up as follows:

Tracking performance:

$$Gm(z^{-1}) = z^{-1}\frac{0.0562z^{-1} + 0.0438z^{-2}}{1 - 1.4574z^{-1} + 0.4724z^{-2}}$$

Regulation performance:

$$D(z^{-1}) = 1 - 0.7z^{-1} + 0.12z^{2}$$

Simulation Results:

The controlled output in Figure 6.9 (c) entirely has the same profile as the desired response with the exception of load changing point (see at Figure 6.9 (e)). At this point, they are

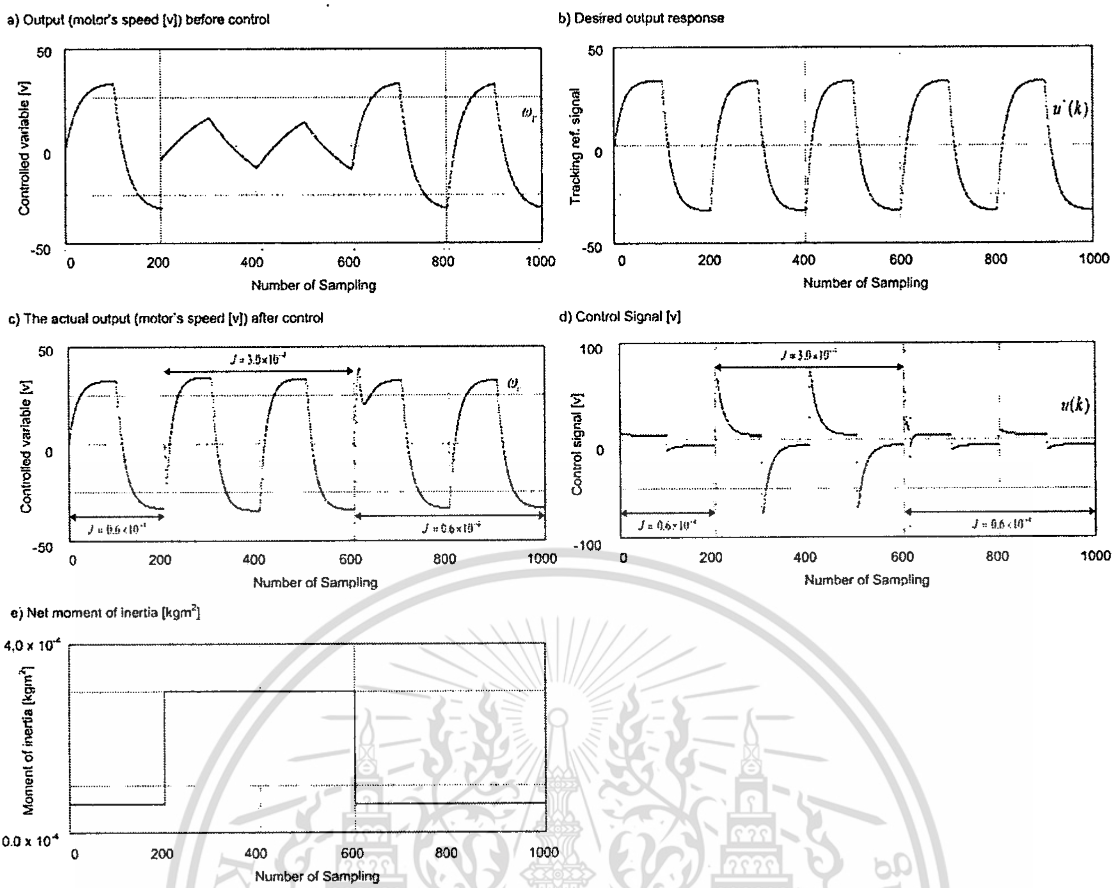

Figure 6.9 The experimental outcomes of tracking efficiency test.

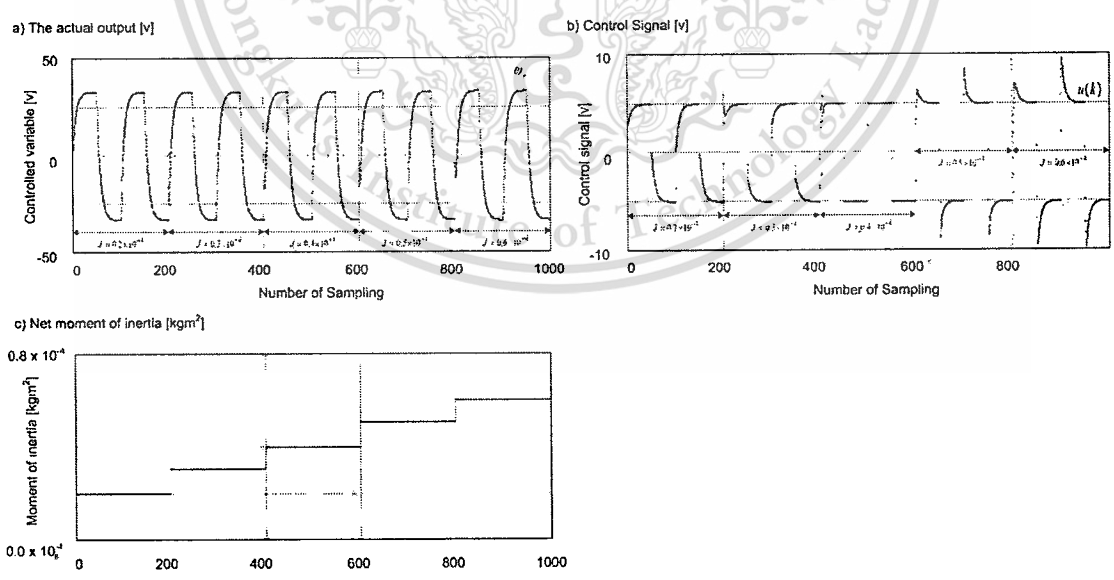

Figure 6.10 The experimental outcomes of tracking efficiency test with the moment of inertia 0.2 x 10^{-4} kgm^2 to 0.6 x 10^{-4} kgm^2.

slightly different. While there is no occurrence of fluctuation in the control signal as shown in Figure 6.9 (d).

Figure 6.10 (a) demonstrates the response of dc motor whose the moment of inertia gradually changes from 0.2 x 10^{-4} kgm^2 to 0.6 x 10^{-4} kgm^2(Figure 6.10 (c)). The set point response is determined as same as the response of 0.4 x 10^{-4} kgm^2 moment of inertia. Considering Figure 6.10 (b), the control signal is above the steady state level when the load is heavier than 0.4 x 10^{-4} kgm^2. In the same way, the control signal is below the steady state level when the load is lighter than 0.4 x 10^{-4} kgm^2.

6.4.2 Regulation Efficientcy Test

When the response is stable, the regulation performance is checked by activating the clutch to connect the extra load and suddenly deactivated afterward.

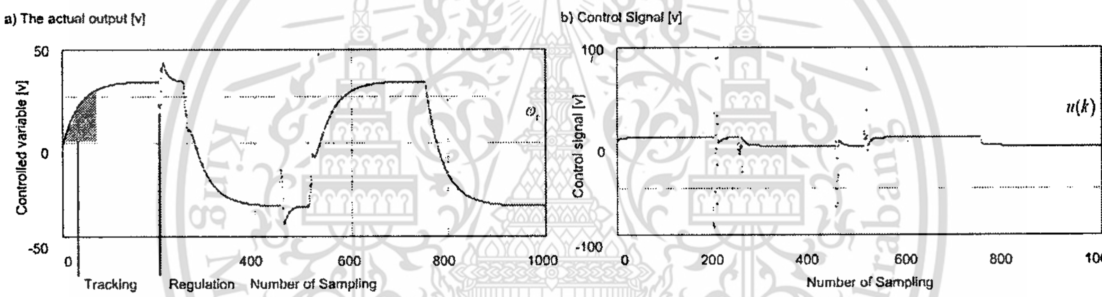

Figure 6.11 (a) Plant output and (b) control signal of the regulation test.

Simulation Results:

The actual output deviates immediately and then come back to the stable point again with the regulation rate as shown in Figure 6.11 (a) and the control signal in Figure 6.11 (b).

6.4.3 Disturbance Rejection Test

This experiment is performed under the condition that the output is stable and the friction torque of the system varies in range of 0 – 2 Nm in square wave form as in figure 6.12(c).

Simulation Results:

The actual output deviates every time that the friction torque alters. But, the controller have the ability of disturbance rejection, it is able to control the speed return to the stable point by itself approximately within the period of 40 samplings.

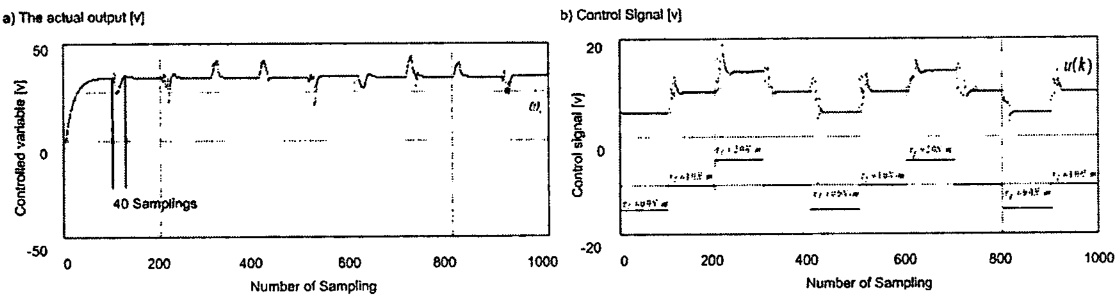

Figure 6.12 (a) Plant output and (b) control signal of the disturbance rejection test.

6.5 Efficiency Comparison Test

In this section, the adaptive state feedback control is compared with the other control methods especially with the adaptive control using the R-S-T compensators. Any advantage, drawback and features of state feedback will be concluded into the table form at the end of section.

6.5.1 Classical Method

The classical adaptive control method (further information given in Appendix A) in figure 6.13 is applied in this test in order to verify the response and the control signal with the same tracking and regulation performance as the section 6.4.1.

Simulation Results:

Although the plant output and control signal in Figure 6.14 (c) and (d) are almost close to those from the state feedback control, when investigate the output at the load changing points, it inferior to the one from state feedback method. In addition, the control signal of this method also has the apparent fluctuation in the transient.

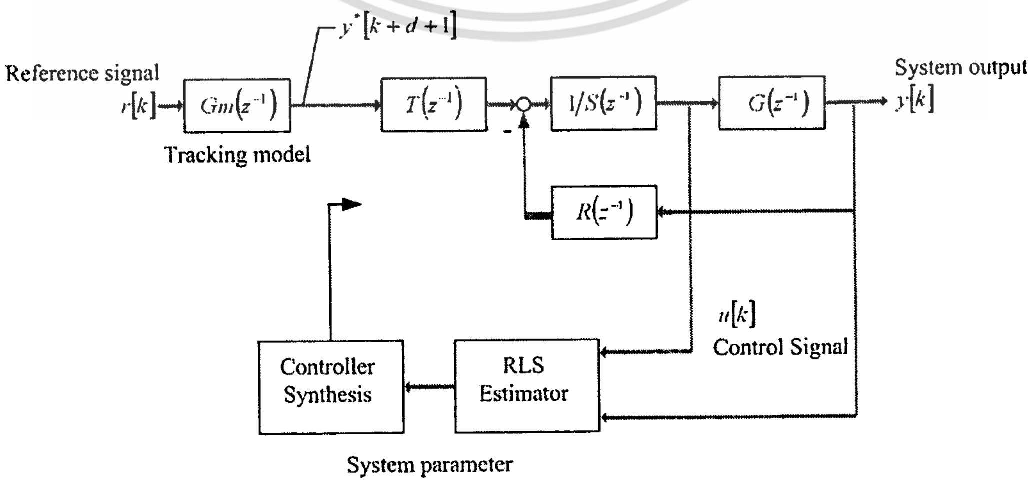

Figure 6.13 R-S-T tri-branch classical adaptive control.

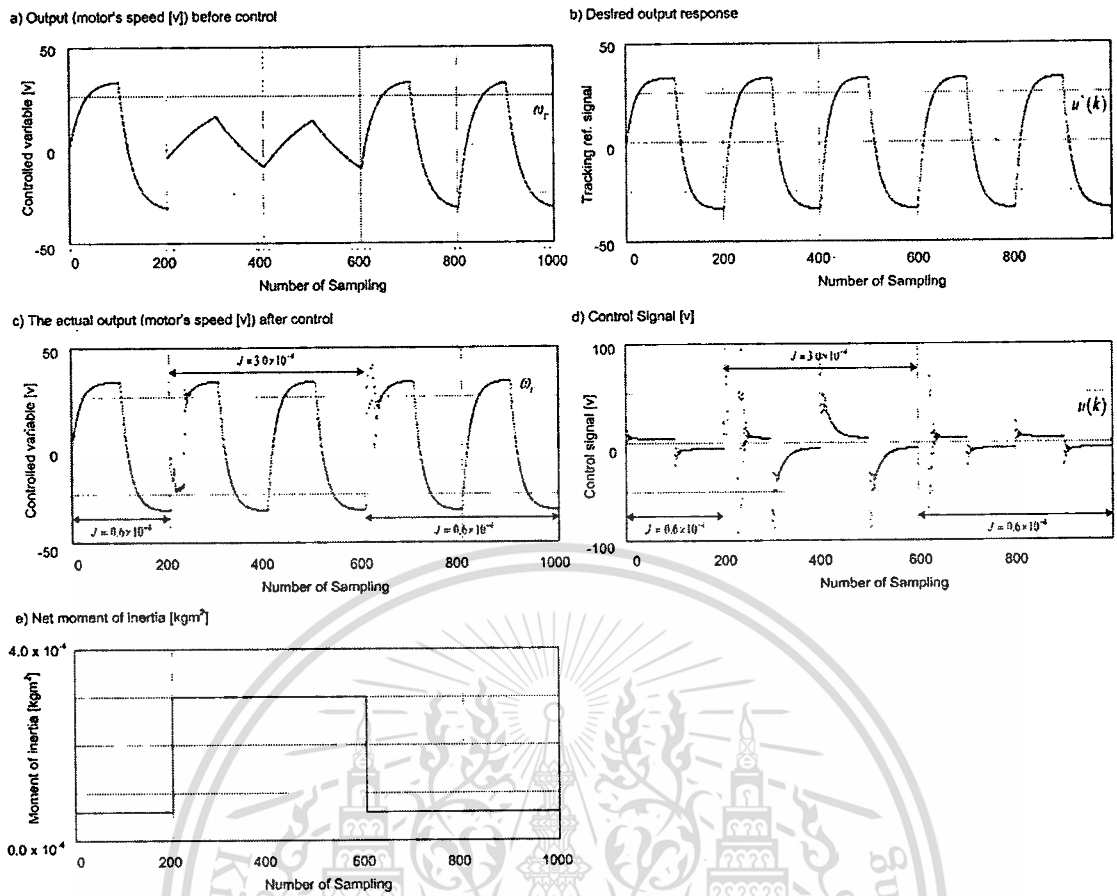

Figure 6.14 The experimental outcomes of R-S-T tri-branch classical adaptive control.

In order to differentiate between compensator and state feedback method more clearly, the further experiments are performed under the given conditions of various tracking and regulation performance. Figure 6.14 through 6.17 show the output signal and control signal of these experiments.

Figure 6.16 and Figure 6.17 show that the control signal of the classical method which has the compensator usage has more fluctuation when the regulation rate is set for faster response. On the contrary, there is no fluctuation of control signal the state feedback method for both regulation rate and tracking rate adjustment as indicated in Figure 6.14 and Figure 6.15.

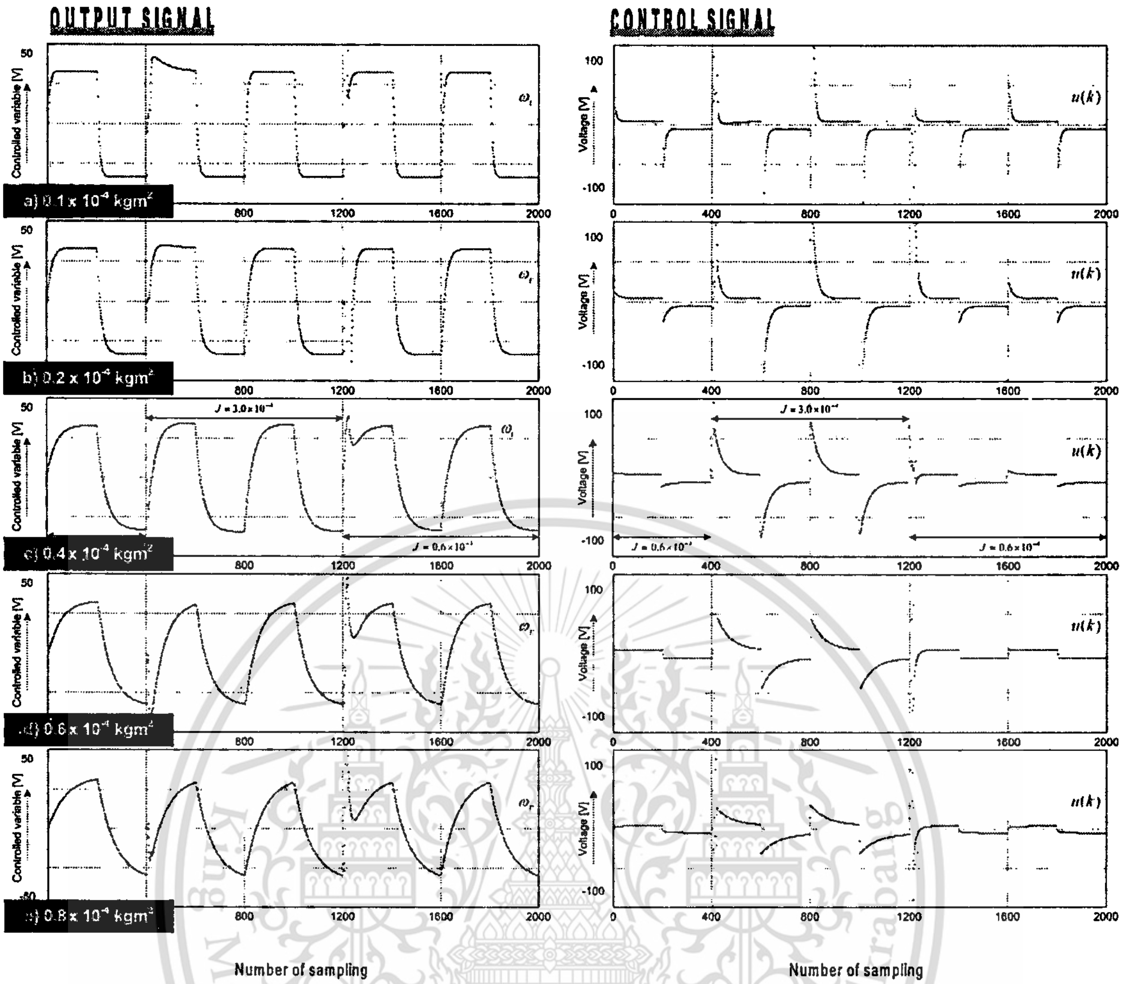

Figure 6.15 The experimental results of $y[k]$ and $u[k]$ of adaptive state feedback with different tracking rate as the response of motor with load varying from 0.2 x 10^{-4} kgm^2 to 0.6 x 10^{-4} kgm^2.

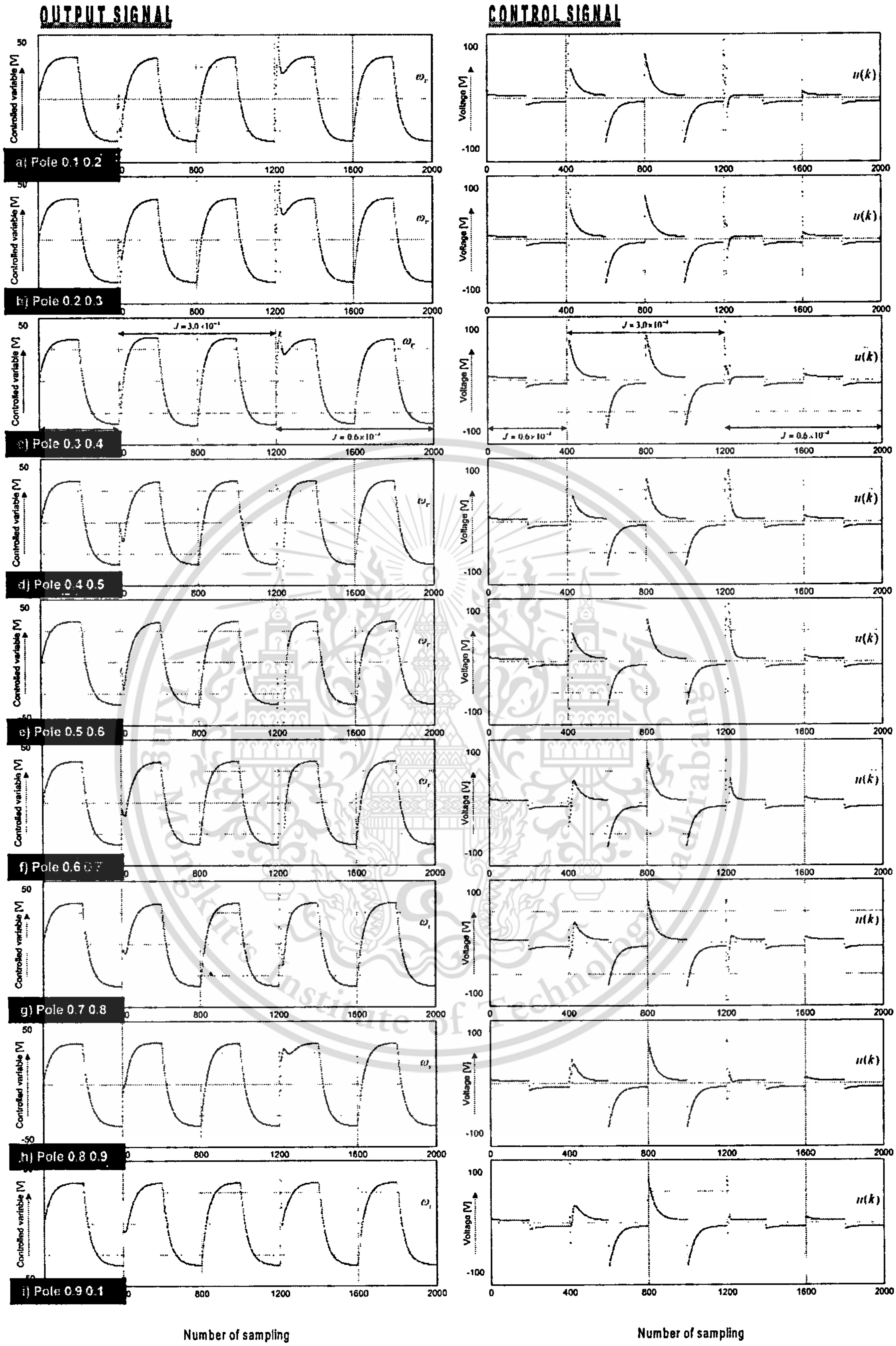

Figure 6.16 The experimental results of $y[k]$ and $u[k]$ of adaptive state feedback with different closed loop poles from 0.0 to 0.9 (regulation rate).

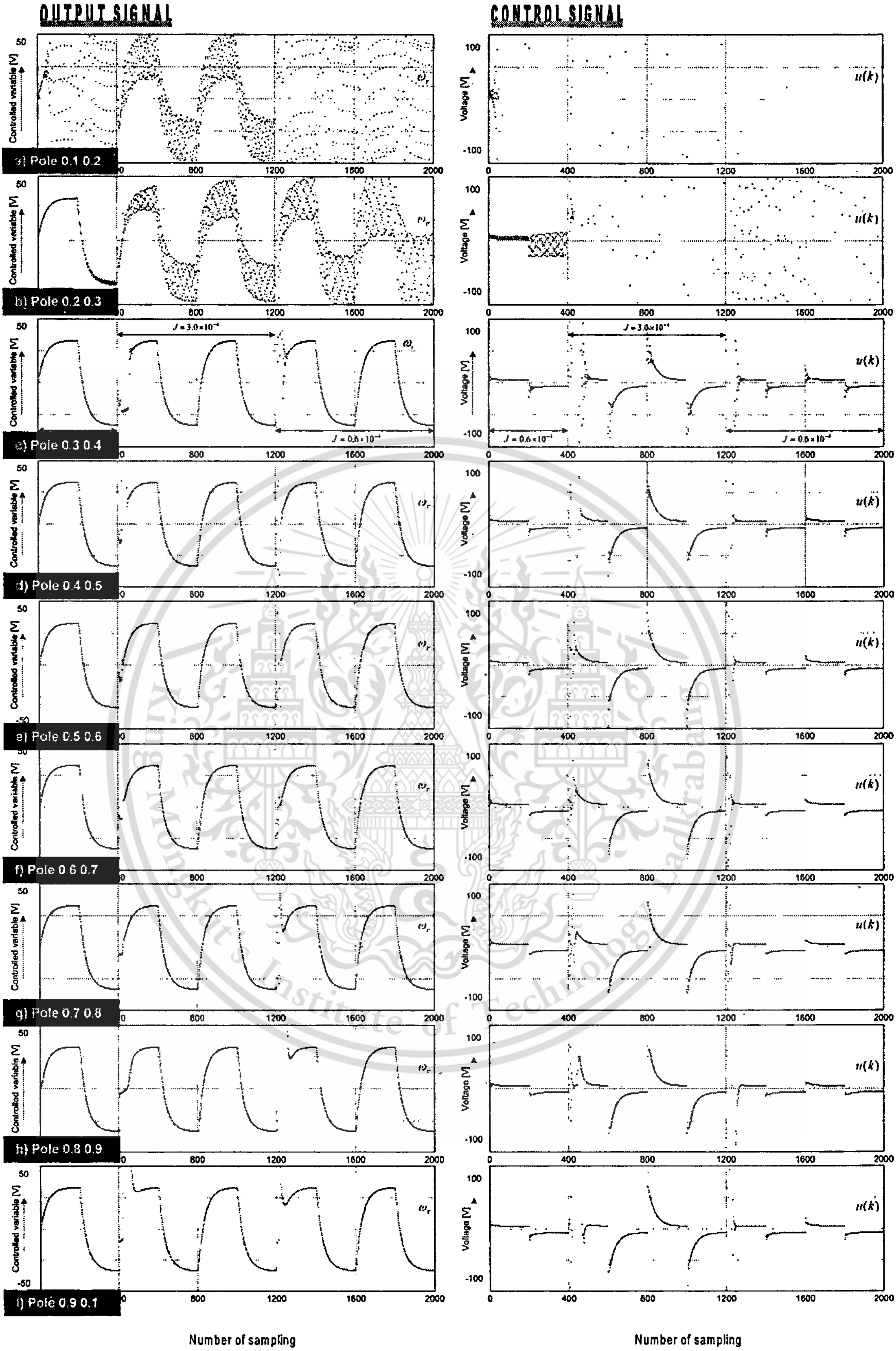

Figure 6.17 The experimental results $y[k]$ and $u[k]$ of normal R-S-T compensators with different closed loop poles of 0.0 to 0.9 (regulation rate).

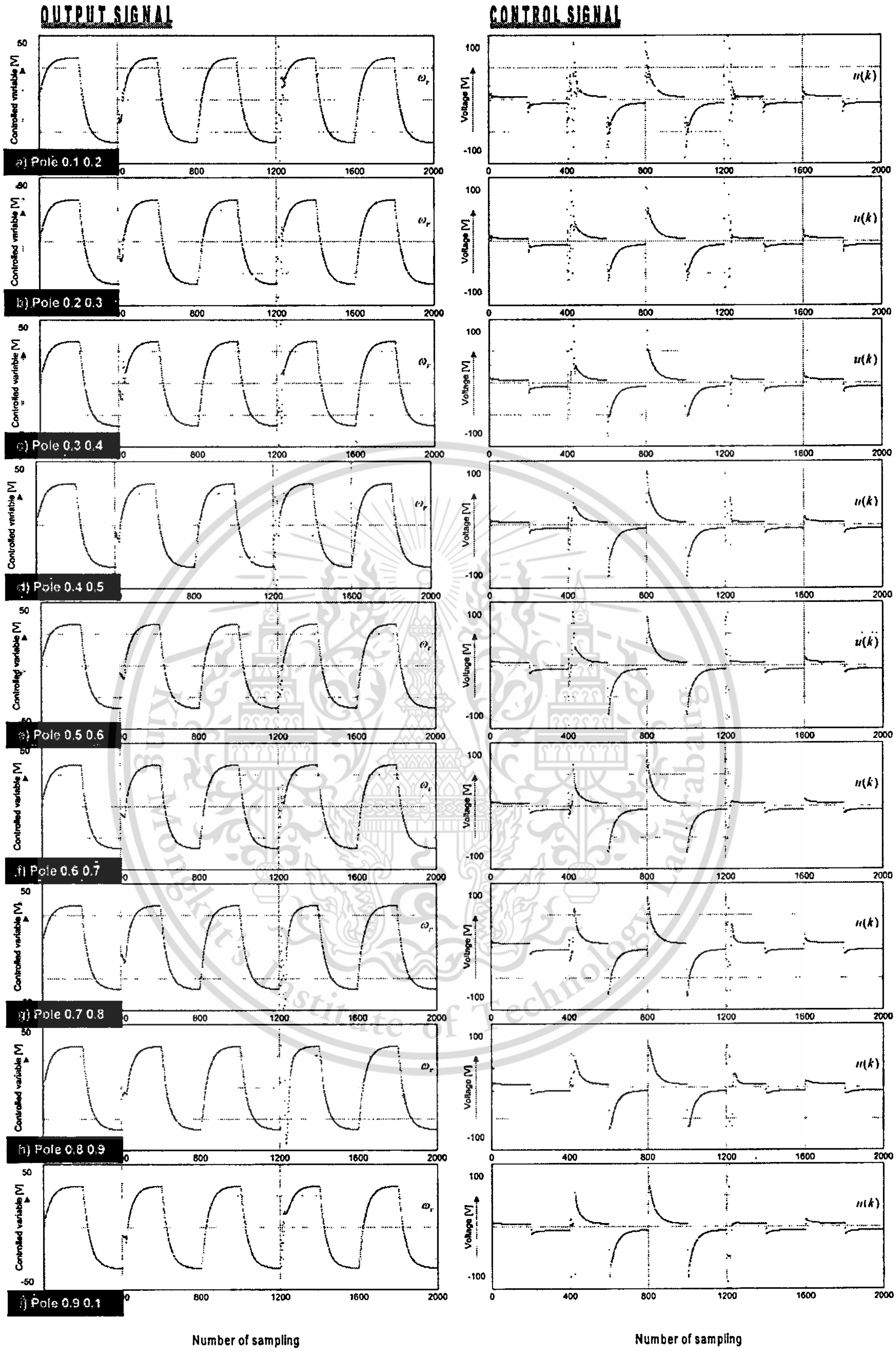

Figure 6.18 The experimental results of $y[k]$ and $u[k]$ of R-S-T independent tracking and regulation with different closed loop poles of 0.0 to 0.9 (regulation rate).

6.5 Conclusion

Taking every experiment results into account, the adaptive state feedback could be the useful method for the DC motor system. So in next Chapter, the practical dc motor experiments are performed.

Chapter 7

Adaptive State Feedback in Practical DC Motor

The performance of the adaptive state feedback is provided in the previous section. With many simulation examples and comparison test with the classical method, we have shown that this method is useful and superior to the old one. Here it is prepared to apply this mathod to the real DC Motor.

The characteristic of dc motor is given in the section 7.1, while every following section is concerned with experiment and analysis.

7.1 DC Motor System Specification

Table 7.1 The design constant of dc motor and symbols of variables: L_a = armature inductance, R_a = armature resistance, K_t = torque constant, K_e = back E.M.F. constant, etc.

Symbol	Description	Value	Unit
- L_a	: armature inductance	6	H
- R_a	: armature resistance	4.3	Ω
- K_t	: torque constant	7.154×10^{-2}	Nm/A
- K_e	: back E.M.F. constant	7.162×10^{-2}	$V\,sec/rad$
- K_p	: power amplifier gain	15	-
- J_m	: moment of rotor inertia	0.4×10^{-4}	Kgm^2
- J_l	: moment of load inertia	0.2×10^{-4}	Kgm^2
- S_r	: conversion constant ω to ω_r.	3.183×10^{-2}	$V\,sec/rad$
- R_r	: resistance for current detection	0.2	Ω
- τ_f	: friction torque	0.001	Nm
- r	: apply voltage	-	V
- i	: armature current	-	A
- ω	: rotor speed	-	Rad/sec
- ω_r	: voltage output of rotor speed	-	V

The dc motor set used in the experiment is shown in Figure 7.1. The DC Motor set consists of an amplifier (Kp=15), an encoder (1000P/R), an electronic clutch (24 vdc) and a dc motor. The major constants of dc motor set are shown in Table 7.1. The heavy load is connected by the electric clutch at the motor shaft. When the sampling time is selected as $T_S = 0.005$ [sec], the discrete time model of plant $G_l(z^{-1})$ (light load) and $G_h(z^{-1})$ (heavy load) are estimated by RLS estimator, respectively, as shown in Equation (7.1) and (7.2).

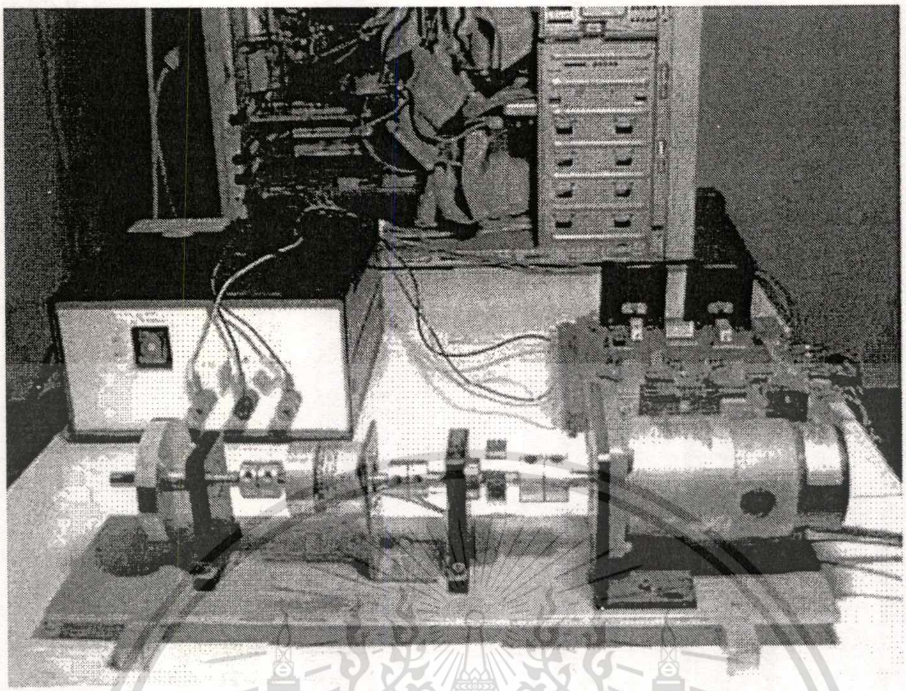

Figure 7.1 DC Motor set for performing the speed control.

$$G_l\left(z^{-1}\right)=z^{-1}\frac{0.011543+0.17697z^{-1}}{1-1.33901z^{-1}+0.52905z^{-2}} \tag{7.1}$$

$$G_h\left(z^{-1}\right)=z^{-1}\frac{0.004525+0.05581z^{-1}}{1-1.77579z^{-1}+0.83703z^{-2}} \tag{7.2}$$

and the response of the system $G_l\left(z^{-1}\right)$ and $G_h\left(z^{-1}\right)$ are plotted in Figure 7.2 (a) and Figure 7.2 (b) respectively. They are approximately equal except only that the first one $G_l\left(z^{-1}\right)$ has the lower overshoot than the latter $G_h\left(z^{-1}\right)$.

7.2 State Feedback Test

Adaptive state feedback is tested for its operation of pole placement as well as the RLS estimation and state variable generation in this section. Since the test of feedback is done once by observing the real closed loop polynomial in the simulation program, so in this section, the examination is under consideration of its actual response, by comparison with the selected polynomial.

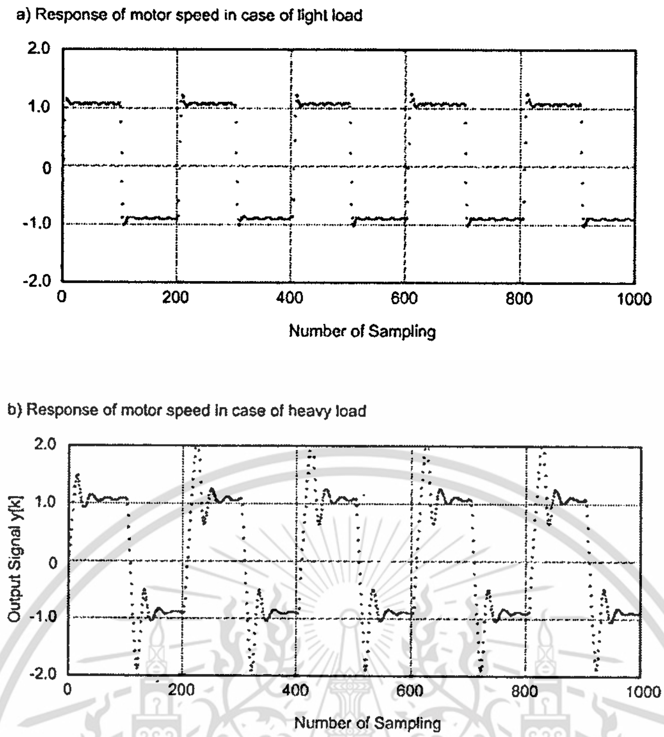

Figure 7.2 The response of motor: a) in case of light load $G_l(z^{-1})$, b) in case of heavy load $G_h(z^{-1})$.

The control method is state feedback without tracking model. The sampling time is chosen as 0.005 sec. and the program period is the period of 1000 samplings. The motor's load is constant.

The experiment examples are performed by the different selection of closed loop polynomials. For the reason of simple inspecting, they are all defined by various damping ratio (0.1 – 1.0) as same as the continuous time second order system in order to obtain the response with low and high overshoot.

Experimental result:

Figure 7.3 shows the outcomes of this experiment. It agrees with expected results and theory. The overshoot increases and swings higher when damping ratio is closer to zero (0.1). This indicates that state feedback able to control the plant response as designed effectively.

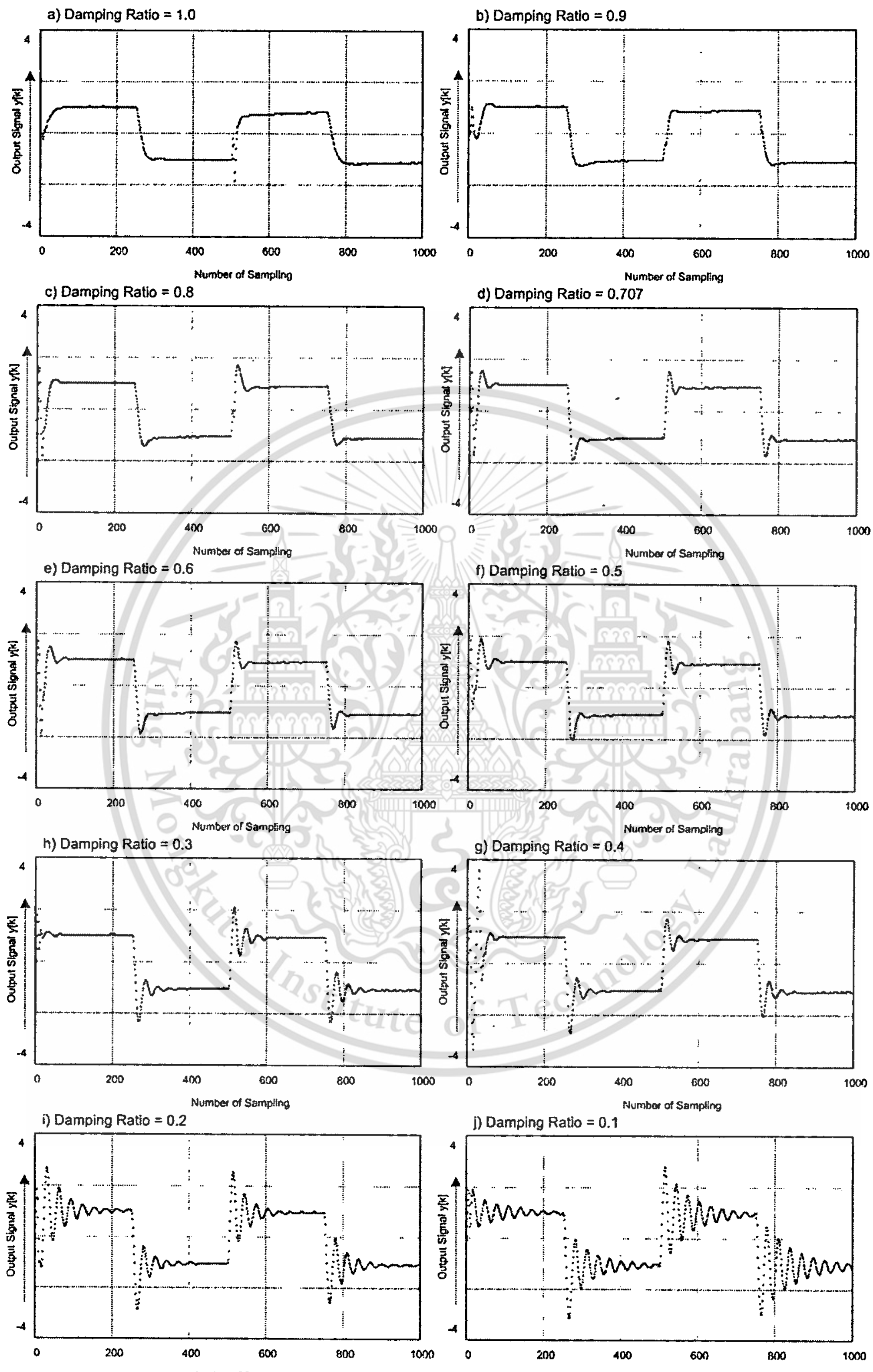

Figure 7.3 The response of closed-loop system with various damping ratio values (0.1 – 1.0).

7.3 Adaptive State Feedback Examples

In this section, the control method of adaptive state feedback is investigated in case of load, control performance and observer gain change.

7.3.1 Load Change

The control method is selected according to figure 6.6, State feedback with tracking model. The closed loop polynomial is chosen as $D(z^{-1})=1-0.3z^{-1}+0.02z^{-2}$. The motor's load changes at the 500[th] sampling. In Example 1, the system parameter changes from light $G_l(z^{-1})$ to heavy $G_h(z^{-1})$ and, in Example 2, the system parameter changes in the opposite way from heavy $G_h(z^{-1})$ to light $G_l(z^{-1})$.

The tracking model is selected as

$$Gm(z^{-1})=z^{-1}\frac{0.012+0.008z^{-1}}{1-1.7z^{-1}+0.82z^{-2}}$$

Experimental result:

Considering Figure 7.4, the output signal $y[k]$(7.4(a)) follows the desired response very closely, while all related factors (control signal, observer output, etc) are in normal condition even by the time load changes. The errors of RLS estimator and observer are equal to zero across the program length. This confirms that the estimated state variables are generated correctly. With the correct value of state variables, the closed loop system has the new behavior as designed.

And, when consider the results of Example 2 in Figure 7.5, the motor can respond as good as in example 1. From these two examples, it can be concluded that load change does not have any adverse effect on the motor's speed.

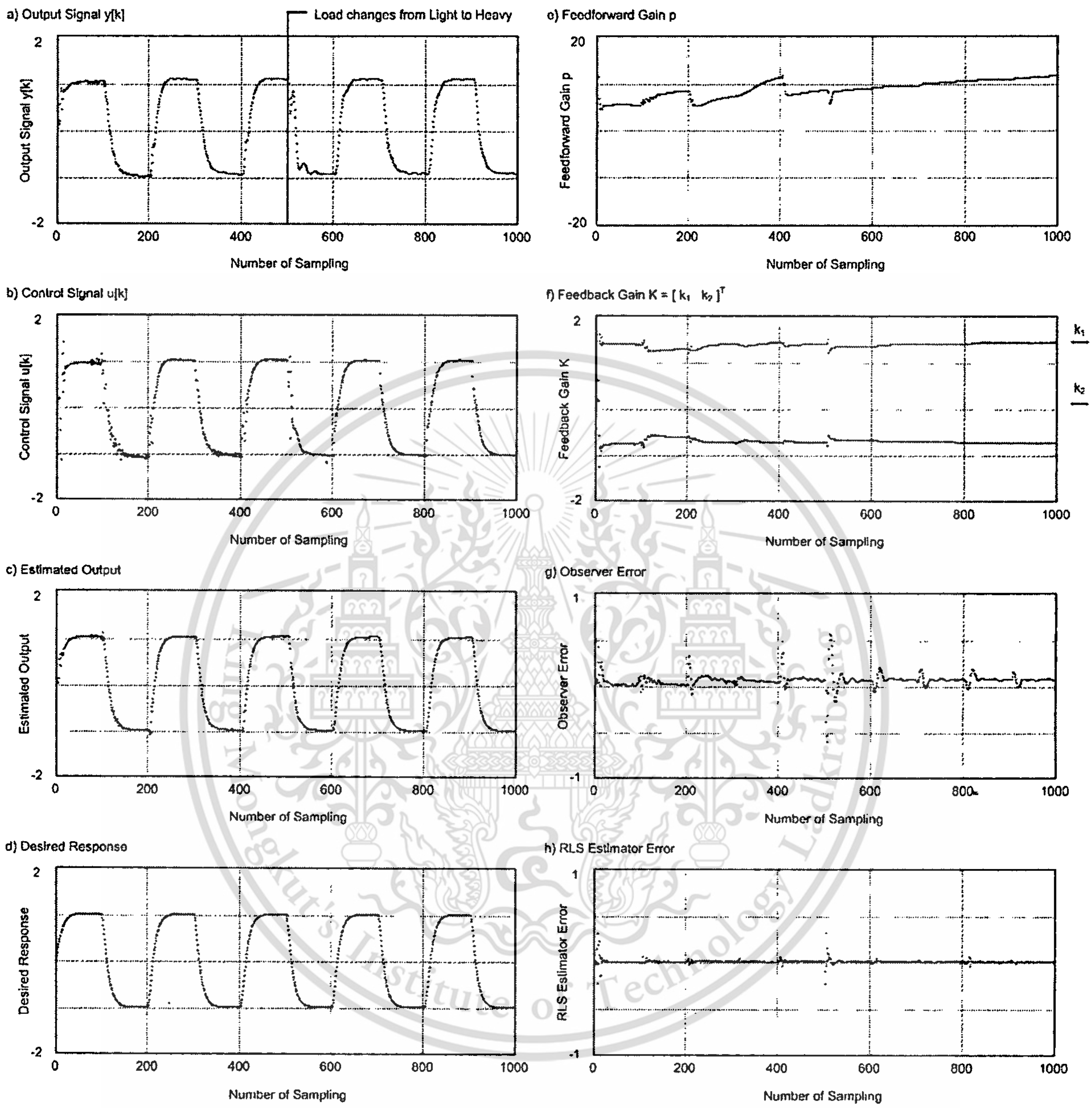

Figure 7.4 The experimental results of example 1 (Light to Heavy).

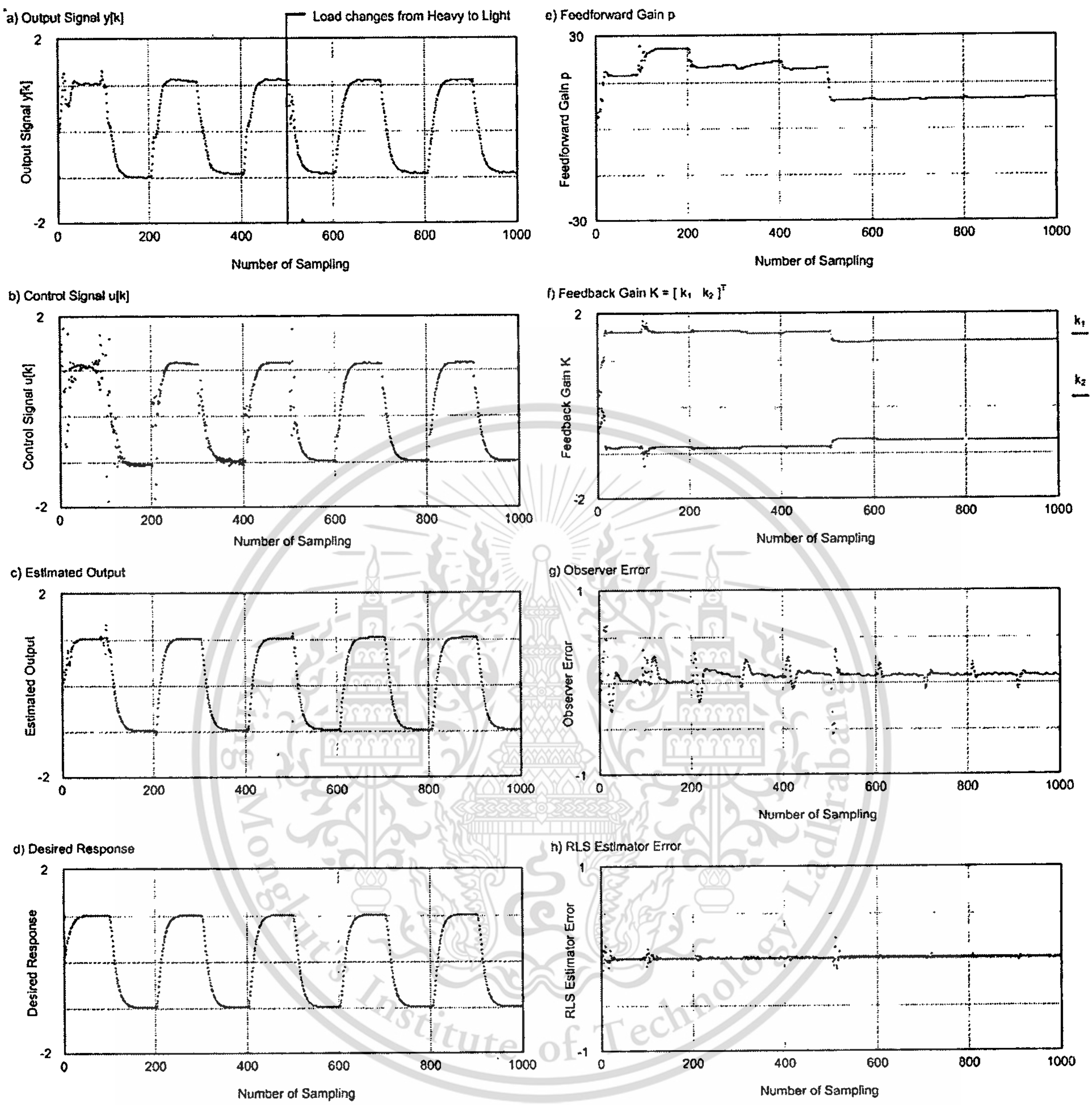

Figure 7.5 The experimental results of example 2 (Heavy to Light).

7.3.2 Constant Tracking Model with Different Regulation Rate at Sampling Time 0.005 sec

The effect of the regulation rate on the control performance is checked in this section as same as in the simulation program. The state feedback with tracking model is used for control method. The tracking model is set at an any constant rate. Only the regulation rate (pair of closed-loop poles) is changed from the value of 0.1 to 0.9.

The tracking model is defined as

$$Gm(z^{-1}) = z^{-1}\frac{0.012 + 0.008z^{-1}}{1 - 1.7z^{-1} + 0.82z^{-2}}$$

The pairs of poles are used as follows: (0.1 0.2), (0.3 0.4), (0.5 0.6), (0.7 0.8) and (0.9 0.1).

The motor's load changes at the 500th sampling from light $G_l(z^{-1})$ to heavy $G_h(z^{-1})$.

Experimental result:

According to Figure 7.6, the control performance with different value of closed loop poles has the same efficiency. The motor's output can track the desired response without the problem in control signal for every pair of poles.

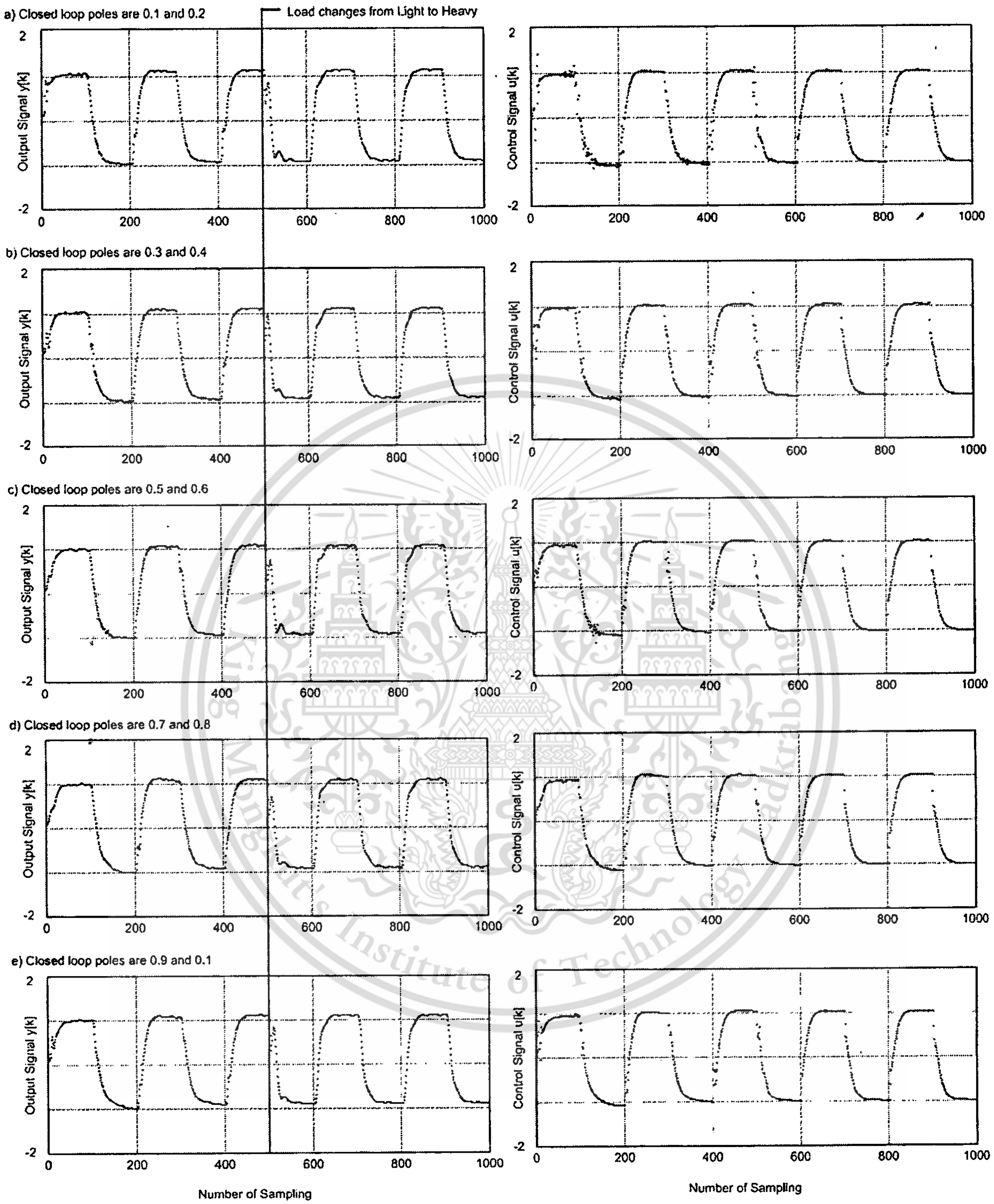

Figure 7.6 The experimental results $y[k]$ and $u[k]$ of adaptive state feedback with different closed loop poles (regulation).

7.3.3 Constant Regulation with Different Tracking Rate at Sampling Time 0.005 sec

The effect of the tracking rate on the control performance now is checked in this subsection as same as the experiment in the simulation program. The state feedback with tracking model is used for control method. The regulation rate is constant at poles 0.1 and 0.2.

The tracking rate changes from slow to fast rate of these polynomials as follows.

Example1. $Gm(z^{-1}) = z^{-1}\dfrac{0.012+0.008z^{-1}}{1-1.7z^{-1}+0.82z^{-2}}$

Example2. $Gm(z^{-1}) = z^{-1}\dfrac{0.019+0.011z^{-1}}{1-1.7z^{-1}+0.82z^{-2}}$

Example3. $Gm(z^{-1}) = z^{-1}\dfrac{0.039+0.021z^{-1}}{1-1.5z^{-1}+0.56z^{-2}}$

Example4. $Gm(z^{-1}) = z^{-1}\dfrac{0.049+0.031z^{-1}}{1-1.4z^{-1}+0.48z^{-2}}$

The motor's load changes at the 500^{th} sampling from light $G_l(z^{-1})$ to heavy $G_h(z^{-1})$.

Experimental result:

According to Figure 7.7 through Figure 7.11 (Example 1 to Example 4 respectively), the control performance with different value of tracking rate has the same efficiency in case of light load. The motor's output can track the desired response no matter faster or slower rate. While in case of heavy load, the motor speed has the unpleasant overshoot when the tracking rate is faster. This problem is caused by the error between the real state variables and estimated ones which can be verify from their output. It can be solved by the observer gain adjustment that will be tested in the next subsection.

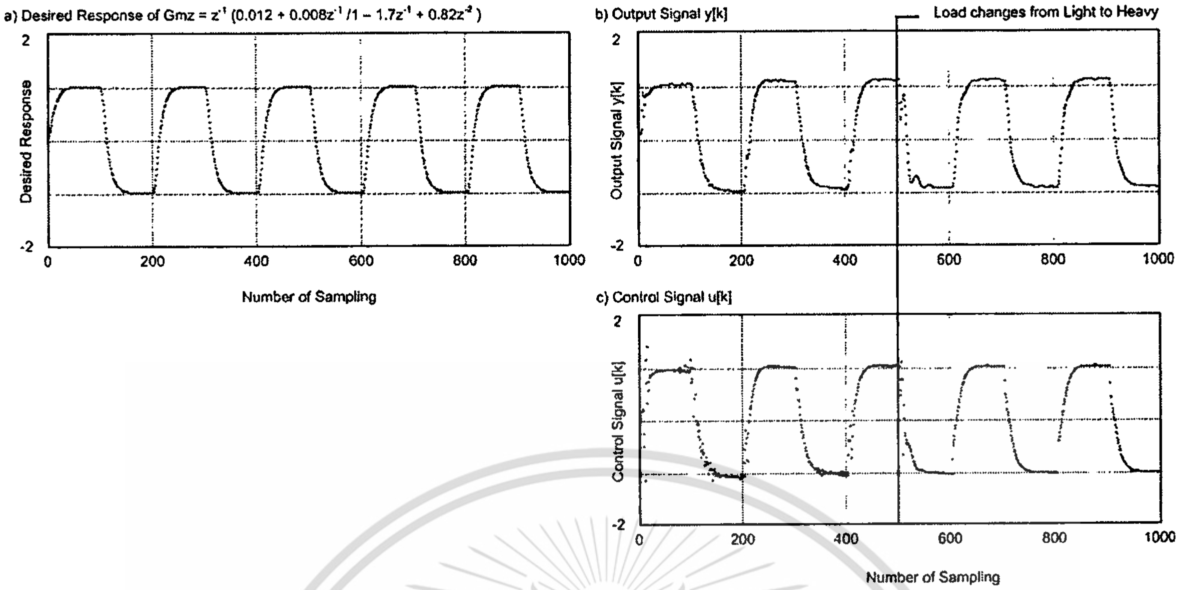

Figure7.7 The experimental results $y[k]$ and $u[k]$ of Example1 (Control with different tracking rate).

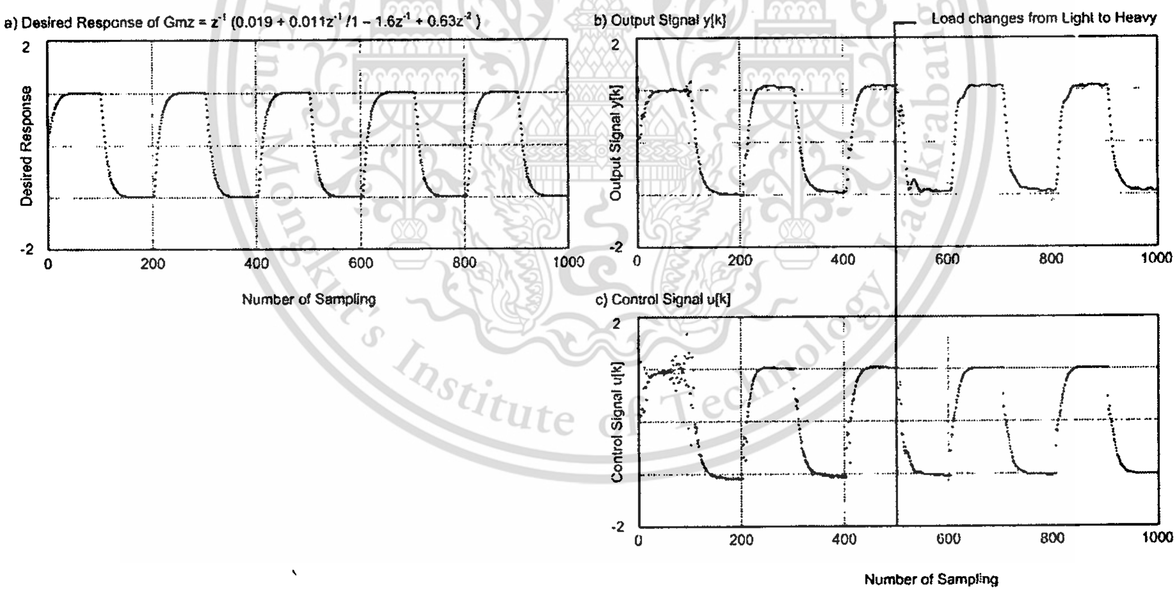

Figure7.8 The experimental results $y[k]$ and $u[k]$ of Example2 (Control with different tracking rate).

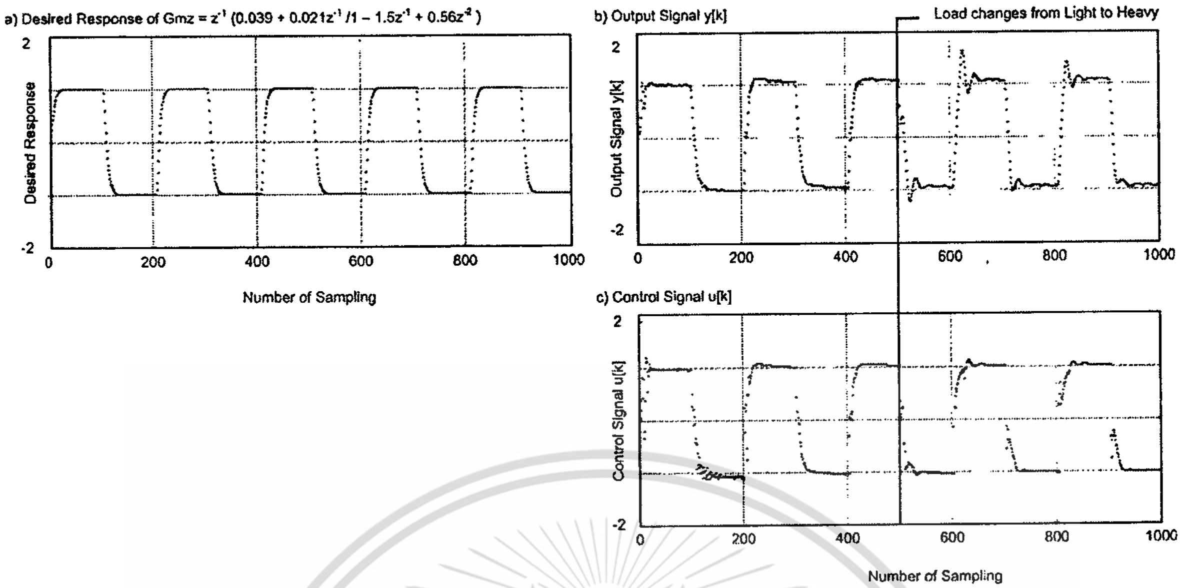

Figure7.9 The experimental results $y[k]$ and $u[k]$ of Example3 (Control with different tracking rate).

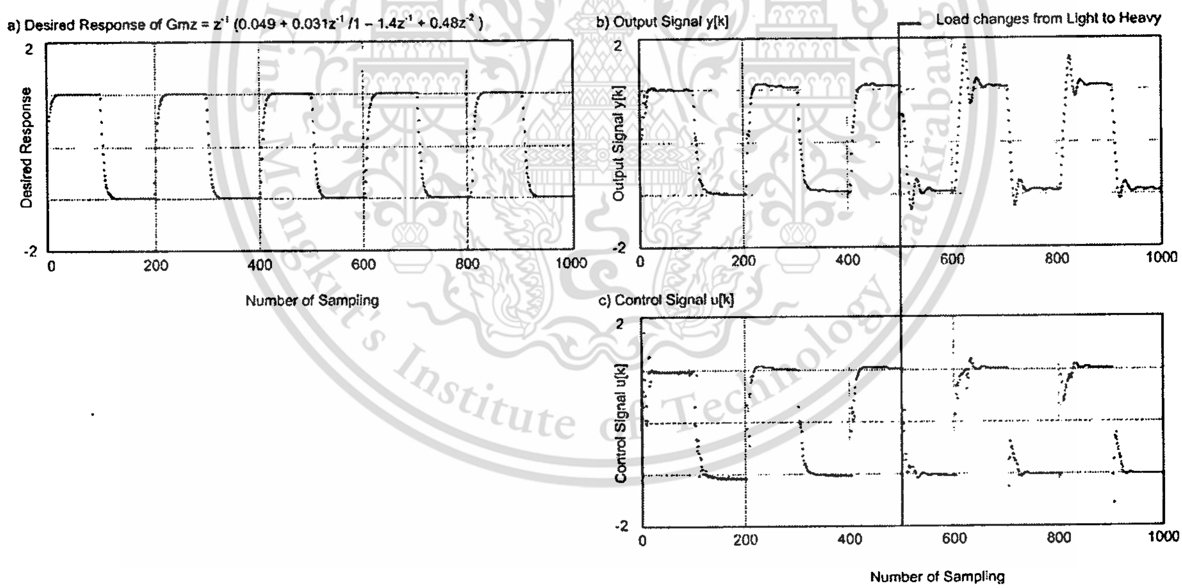

Figure7.10 The experimental results $y[k]$ and $u[k]$ of Example4 (Control with different tracking rate).

7.3.4 Observer Gain l Test

The response is worse when the desired response is faster as the results of figure 7.25. This problem happens because the value of state variables are not exact at every point. Therefore the observer gain l is adjusted to give the faster rate of $(A-lC)$ ([1 1]^T, [10 10]^T, [12 12]^T and [13 13]^T)in this section.

where $D(z^{-1}) = 1 - 0.3z^{-1} + 0.02z^{-2}$ and $Gm(z^{-1}) = z^{-1}(0.049 + 0.031z^{-1})/1 - 1.4z^{-1} + 0.48z^{-2}$. The motor's load changes at the 500th sampling from light $G_l(z^{-1})$ to heavy $G_h(z^{-1})$.

Experimental result:

After the value of observer gain is higher, the output has no longer the overshoot in the period of heavy load. In case of light load, however, the output of the first cycle becomes worse. The abrupt change occurs in this area as indicated in Figure 7.11 through 7.14. Since, when we use the observer gain as [1 1]^T, therefore it is obvious that it is caused by the inappropriate observer gain (in this case it is too high for light load condition).

For that reason, the observer gain is switched between [1 1]^T and [13 13]^T for light load and heavy load. As shown in Figure 7.15(b), the output now has no unexpected fluctuation and overshoot.

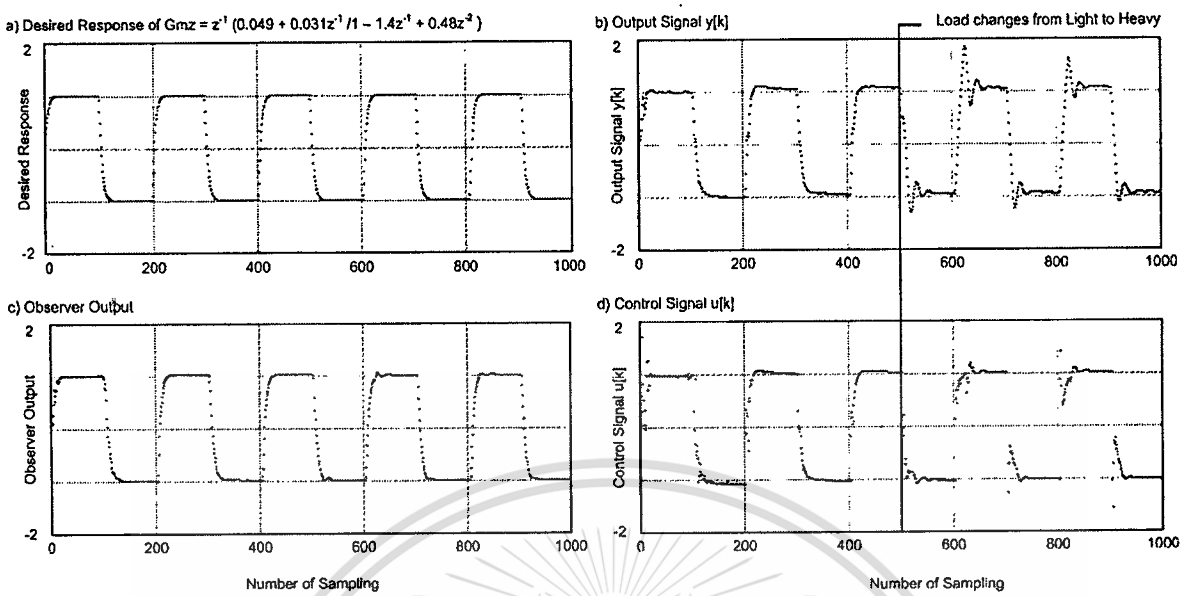

Figure7.11 The experimental results of adaptive state feedback observer gain $l = [1\ 1]^T$.

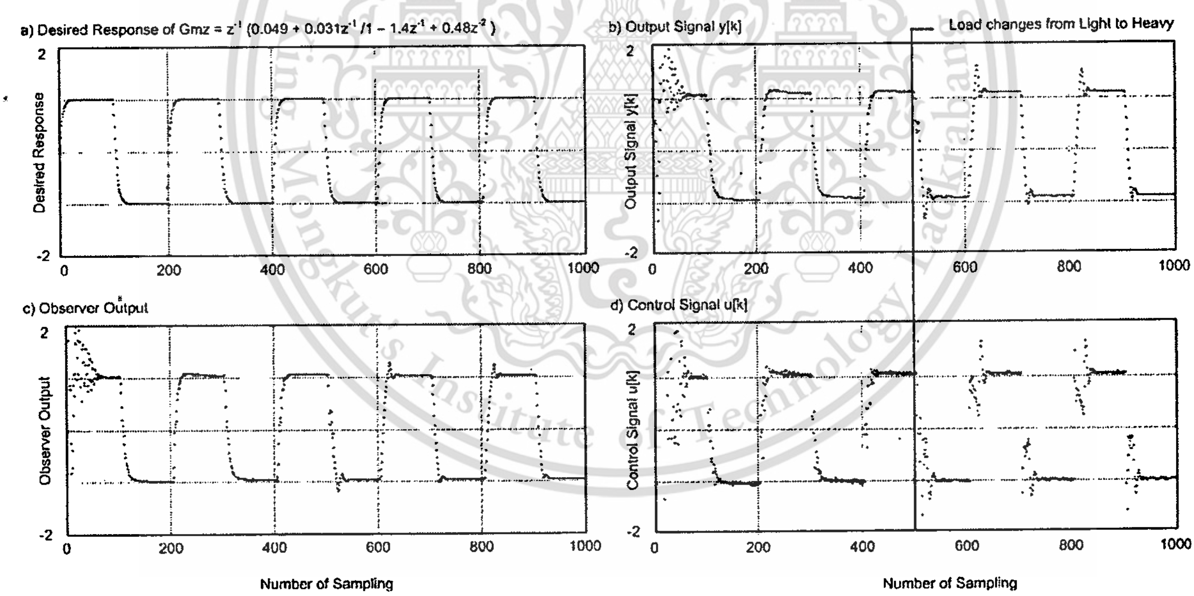

Figure7.12 The experimental results of adaptive state feedback observer gain $l = [10\ 10]^T$.

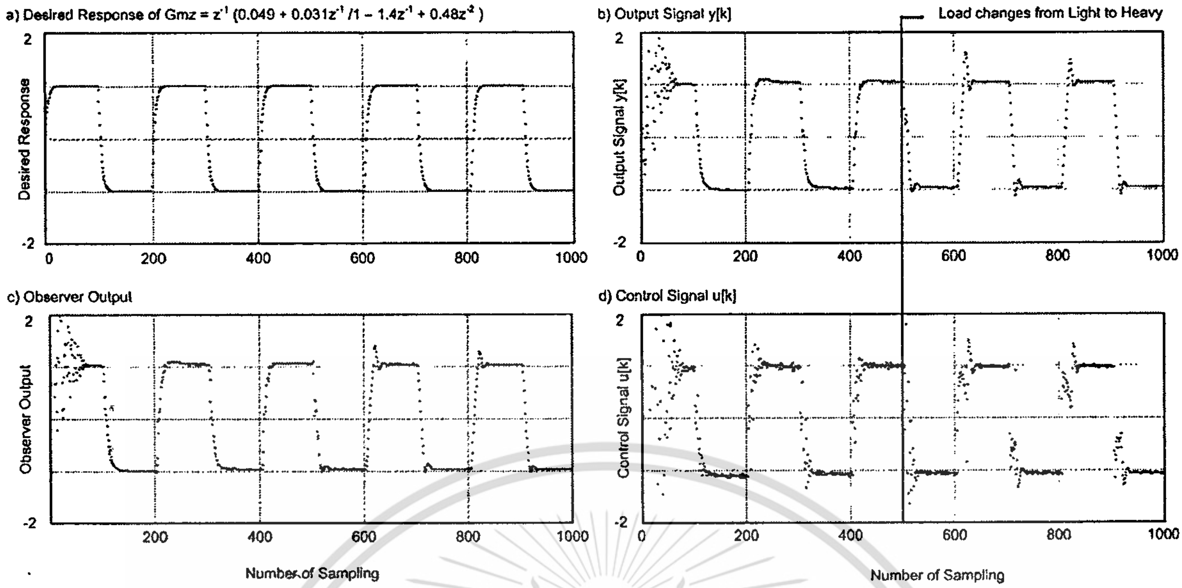

Figure7.13 The experimental results of adaptive state feedback observer gain l = [12 12]^T.

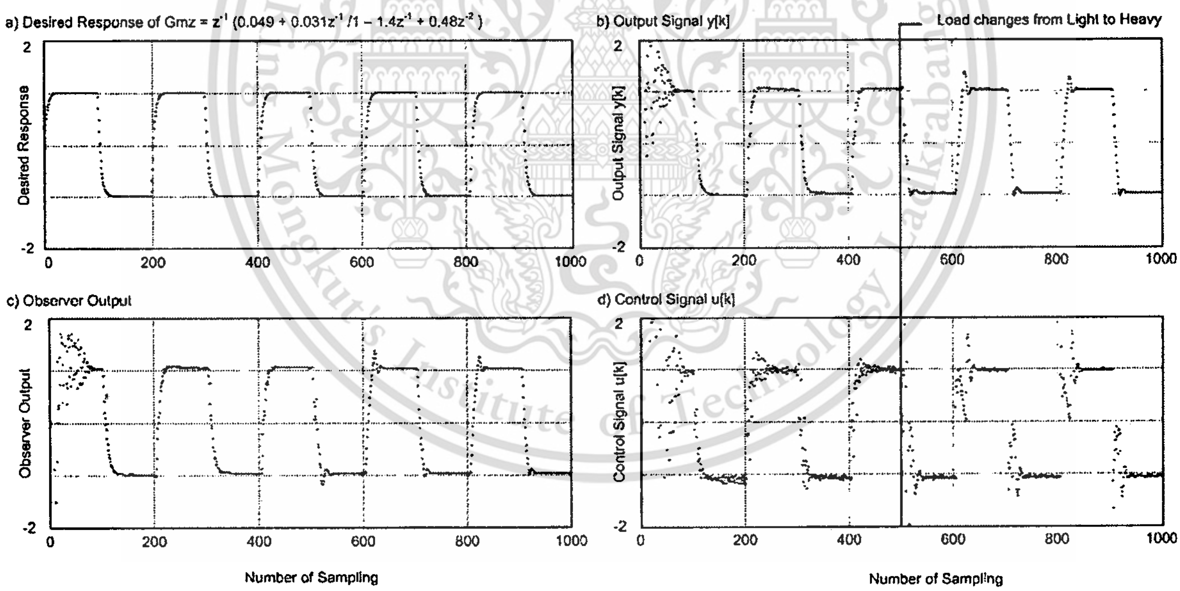

Figure7.14 The experimental results of adaptive state feedback observer gain l = [13 13]^T.

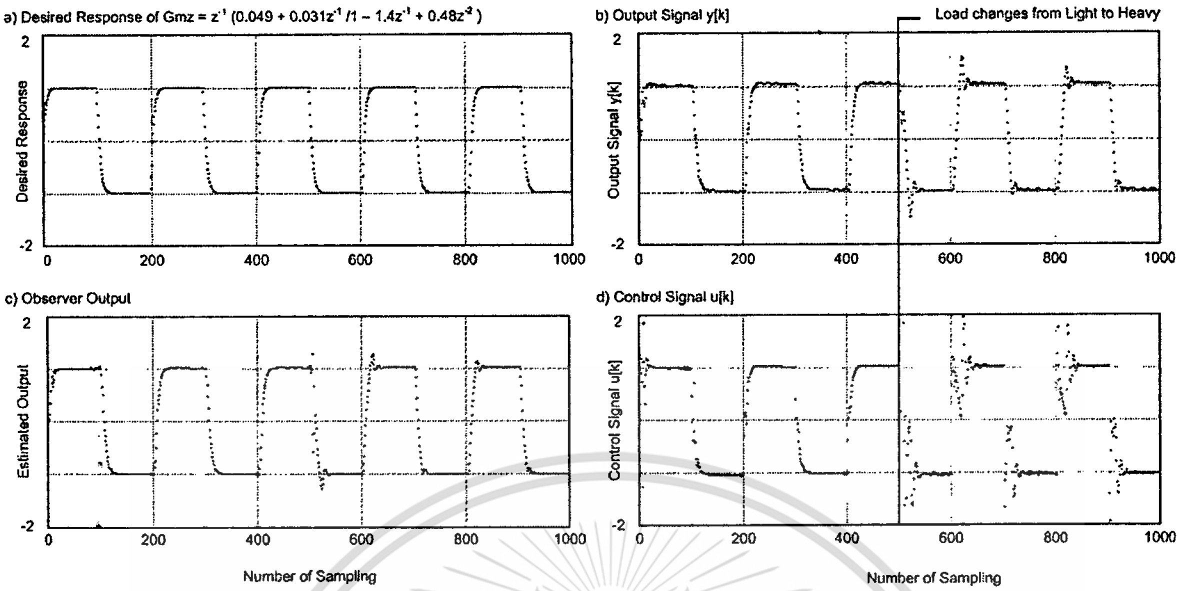

Figure7.15 The experimental results of adaptive state feedback observer gain l switches between $[1\ 1]^T$ and $[13\ 13]^T$.

7.4 Conclusion

The chapter has presented the adaptive state feedback experiments on the practical DC motor with different parameter. The response of the motor is stable even though when the system parameters change. This demonstrates the utmost advantage of the adaptive control that can keep the entire response of the system. The next chapter will disscuss every experiments of this chapter again and gives some suggestions.

Chapter 8

Conclusion and Suggestion

The adaptive state feedback of dc motor speed control has been introduced for tracking and regulation problem in the thesis. The pole placement method with state feedback is used to shift the position of closed loop poles to the satisfying position. Since the situation is assumed that the system is unaccessable (unable to measure the internal states), the state observer becomes the requirment of such a system. This observer can exactly do its task as the state variable generation on condition that its mathematical model is quite right as the real system parameter. Furthermore, this model is used for the state feedback gain selection criterion. The controller compares it with the designed one and compute the proper value for the feedback gain. With this reason the system identification is the most important part in the adaptive control which should be studied profoundly for its properties and using. And since this study involves the use of state variables in pole placement, the basic of state space analysis and control design shold be learnt.

Chapter 2 provides the recursive least square algorithm. It discusses the theory of least square to estimate the system parameters and its properties at first. Then the recursive least squarestimation, allowing the update estimated model at each sampling, is given as well as the alternative algorithm, Recursive Extended Least Squares.

Chapter 3, the using recursive estimation, helps explain the correct direction of RLS algorithm with the real situation. The initializing parameter and specifying estimator, the operation level and effect of covariance are described. At the end of chapter, some examples of estimation in the progam simulation and dc motor are given. As the experiment results the RLS estimator can identify the parameters of the system well even it in case of system parameter change.

In Chapter 4, section 4.1 state space representation demonstrates how the transfer function is transformed into the state space equation. Controllability and observablity are also presented.

Chapter 5, design of control system in state space, considers the concept of state feedback as pole placement. It shows that the position of closed loop poles can be arbitrarily chosen if the system is controllable. This means the system is able adjusted to any control demand. The theory of closed loop state observer is explained that it can dispose of the error of state variables because of initial state setting. So feedback the estimated state variables from the observer is usable. It is not quite as great as the direct feedback from real state variables of plant though. At this point the way to construct the adaptive state feedback is clearified. the RLS estimator, state feedback and state observer are joined together. The function of this combination is named "the adaptive state feedback".

In Chapter 6 the way to make the simulation program is given in the first section: modeling in state space and discretization. This progam simulation will be used for every experiment in this chapter. The accuracy of adaptive state feedback is verified in example of section 6.2.1. It indicates that the closed loop polynomial has the same value as setting.

Concerning the tracking problem, sometimes it is needed being independent of the regulation one. For that reson, the reference model or tracking model is applied to control system together with a pre-filter. They are placed in front of the feedback point. The refference model produces the signal that has the preferable trajectory to the original of the system. While the pre-filter works at compensating for the closed loop system gain like feedforward gain. It is also adjusted to the system parameters accordingly.

In this research, the proposed method, adaptive state feedback, is tested in the computer simulation of dc motor system before applying it to the real dc motor. The programe simulation provides the approximately exact response of the motor's speed and current under given conditions. The tracking efficientcy test is the fisrt test. The square wave is supplied and the load is changing during control. The output will respond as same as desired one over the program time except at changing point. The output is subjected to the load changing for a while at this moment. Then it completely disappeares at the subsequence pulse. This happens because when system parameter changes the estimator need a short time to correct its model. Secondly, the regulation efficientcy test is performed by increasing the load inertia when the output begins stable. The speed of motor deviate from the stable point and then it is recovered by the regulation at the setting rate.

The results of the experiments on tracking and regulating performance indicate that it is practicable to control the dc motor as same as the classical method which has the compensators. However, it is also able to deal with the fluctuation problem in control signal that happens to the use of compensators.

Finally, in Chapter 7 the adaptive state feedback is used for controling the real dc motor speed. The load is increased by adding the additional disk. It is coupled to the shaft of motor when the electric clutch is engaged immediately the system parameter changes. The range of parameter changing depends on the extent of disk or the moment of inertia [N.m]. So there can be two options that can make the system changing: system 1 $G_l(z^{-1})$ (clutch off J = 0.6 x 10^{-4} kgm^2) to system 2 $G_h(z^{-1})$ (clutch on J = 3.0 x 10^{-4} kgm^2) or vice versa. The experimental results in practical dc motor show the good performance of the proposed method in both cases. Although there is a little overshoot in case that the motor is controlled to respond more rapidly, the overall performance is still acceptable. As many and different experimental results of the adaptive state feedback, the thesis conclude that it is one of effective methods for the dc motor speed control.

In this research, the efficiency of closed loop system, however, is still manually chosen by the designer. This duty requires the knowledge of control theory in order to gain the suitable efficiency for each system. Therefore it can make trouble for unskilled users and the automatic function of optimal pole selection for each system may become the interesting issue.

And, since this thesis focuses only on the adaptive controller of dc motor, the experiment of adaptive state feedback should be more widely performed in various plants as well.

The simulation used in Chapter 3 and 6 is written on visual basic 6 program, while the program communicating the 12 bit A/D and D/A board for dc motor drive and data acquisition in Chapter 7 is on C++ programming language.

REFERENCES

[1] Astrom K. J. and Wittenmark B. **Adaptive Control.** Addison Wesley. 1989.

[2] Landau I. D. **System Identification and Control Design.** Prentice Hall. 1990.

[3] Soderstrom T. and Stoica P. **System Identification.** Prentice Hall. 1989.

[4] Ljung L. **System Identification Theory for the User.** Prentice Hall. 1987.

[5] Narendra K.S. and Annaswamy A.M. **Stable Adaptive Systems.** Prentice Hall. 1989

[6] Chi-Tsong Chen. **Linear System Theory and Design.** Third Edition. 1999.

[7] Panicom S., Hirata H., Julsereewong P. and Ratiroch-anant P. "Auto-tuning Strongly Stable Adaptive Control System of DC Motor Speed." **ECTI-CON.** 2005. pp. 137-140.

[8] Ratiroch-anant P., Anabuki M. and Hirata H. "Self-Tunung Control for Rotational Inverted Pendulum by Eigenvalue Approach." **IEEE TENCON.** 2004.

[9] Seung Min Baek and Tae Yong Kuc. "An Adaptive PID Learning Control of DC Motors." **IEEE.** 1997.

[10] Andrzej Dzielinski and Dominik Sierociuk. "Adaptive Control of Fractional Order Discrete State-Space System." **(CIMCA-IAWTIC'05), IEEE.** 2005.

[11] "A new Approach to State Estimation in Deterministic Digital Control Systems." **NASA Technical Paper.** 1987.

[12] Wen-Jeng Liu. "Design of the Observer Feedback Gain for Two-Dimensional Discrete Systems." **IEEE Signal. Processing Letter.**, vol. 11, no. 4, April 2004. pp. 413-415.

[13] Trinh H. and Aldeen M. "A Memoryless State Observer for Discrete Time-Delay Systems." **IEEE Transactions on Automatic Control.,** vol. 42, no. 11, November 1997. pp. 1572-1577.

[14] Weiwen Wang and Zhiqiang Gao. **A Comparison Study of Advanced State Observer Design Techniques.**

[15] Fabiano Salvadori, Mauricio de Campos, Cassiano Rech, Gideon Villar Leandro, Rodrigo Padilha Vieira and Alexandre Cunha Oliveira. "Optimal Discrete-Time Complete Order State Observer Applied to Asynchronous Machine Drive Systems." **IEEE.** 2006. pp. 108-113.

Appendix A

A.1 Poles Placement

The R-S-T tri-branches digital controller is shown in figure A.1, $r(k)$ is the reference input, $u(k)$ is control signal and $y(k)$ is the output of the plant. Since this method does not introduce the additional zero, so it is unnecessary to concern about the overshoot of the response. The plant transfer function is

$$G_p\left(q^{-1}\right)=\frac{q^{-d}B\left(q^{-1}\right)}{A\left(q^{-1}\right)}, d = \text{delay time.} \tag{A.1}$$

The series compensator $S\left(q^{-1}\right)$ is in the forward part, the parallel compensator $R\left(q^{-1}\right)$ is in the feedback part, and $T\left(q^{-1}\right)$ is the pre-filter. The closed loop transfer function is

$$H_{CL}\left(q^{-1}\right)=\frac{q^{-d}T\left(q^{-1}\right)B\left(q^{-1}\right)}{A\left(q^{-1}\right)S\left(q^{-1}\right)+q^{-d}B\left(q^{-1}\right)R\left(q^{-1}\right)} \tag{A.2}$$

in which

$$A\left(q^{-1}\right)=1+a_1q^{-1}+\ldots+a_nq^{-n} \tag{A.3}$$

$$B\left(q^{-1}\right)=b_0+b_1q^{-1}+\ldots+b_mq^{-m} \tag{A.4}$$

where n and m are the order of $A\left(q^{-1}\right)$ and $B\left(q^{-1}\right)$ respectively. The closed loop polynomial is

$$\begin{aligned} D\left(q^{-1}\right) &= A\left(q^{-1}\right)S\left(q^{-1}\right)+q^{-d}B\left(q^{-1}\right)R\left(q^{-1}\right) \\ &= 1+d_1q^{-1}+\ldots+d_{nd}q^{-nd} \end{aligned} \tag{A.5}$$

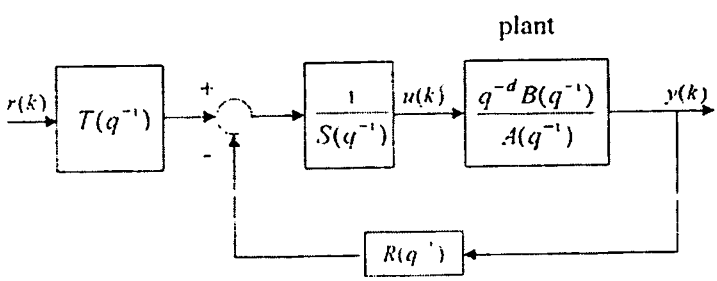

Figure A.1 Pole placement with R-S-T controller.

This equation is known as the Diophantine equation or Bezout identity. The coefficient of $D(q^{-1})$ is linear combination of constant coefficients of $B(q^{-1})$ and $A(q^{-1})$ and unknown coefficient of $S(q^{-1})$ and $R(q^{-1})$ which are given as

$$S(q^{-1}) = 1 + s_1 q^{-1} + \ldots + s_{ns} q^{-ns} \tag{A.6}$$

$$R(q^{-1}) = r_0 + r_1 q^{-1} + \ldots + r_{nr} q^{-nr} \tag{A.7}$$

where nd, ns and nr are the order of $D(q^{-1})$, $S(q^{-1})$ and $R(q^{-1})$ respectively. From Equation (A.5), (A.6) and (A.7) the coefficients can be matched and the Diophantine equation can be calculated from the following matrix equation:

$$\begin{bmatrix} 1 & 0 & . & . & . & 0 & b_0 & 0 & . & . & . & 0 \\ a_1 & 1 & . & . & . & . & b_1 & b_0 & . & . & . & . \\ a_2 & a_1 & & & & . & b_2 & b_1 & . & . & . & . \\ . & . & & & & . & . & . & & & & . \\ . & . & & & & 0 & . & . & & & & . \\ a_{n-1} & . & . & . & . & 1 & b_{m-1} & . & & & & b_0 \\ a_n & a_{n-1} & . & . & . & a_1 & b_m & b_{m-1} & . & . & . & b_1 \\ 0 & a_n & . & . & . & a_2 & 0 & b_m & . & . & . & b_2 \\ . & 0 & & & & . & . & 0 & & & & . \\ . & . & & & & . & . & . & & & & . \\ . & . & . & . & . & a_{n-1} & . & . & . & . & . & . \\ 0 & . & . & . & 0 & a_n & 0 & . & . & . & . & b_m \end{bmatrix} \begin{bmatrix} 1 \\ s_1 \\ . \\ . \\ . \\ s_{ns} \\ r_0 \\ r_1 \\ . \\ . \\ . \\ r_{nr} \end{bmatrix} = \begin{bmatrix} 1 \\ d_1 \\ . \\ . \\ . \\ d_{nd} \end{bmatrix} \tag{A.7a}$$

$$MX = D \tag{A.8}$$

Matrix M is known as the Sylvester matrix, D is the vector of the coefficient of Equation (A.5) and X is a vector of unknown controller parameters, $D(q^{-1})$ is specified by the designer and M is a constant matrix. The solvability of the Diophantine equation can be considered from the Sylvester matrix M.

First, the plant polynomial must co-prime, it means that $B(q^{-1})$ and $A(q^{-1})$ must have no common factors. If such common factor exist, then the roots also appear as the root of the Diophantine equation. Thus, the system is forced to accept the roots as the pole and resulting in constrained pole placement. Furthermore, coprimeness of $B(q^{-1})$ and $A(q^{-1})$ ensures that the

Sylvester matrix is nonsingular. Therefore, if the Sylvester matrix is nonsingular and square, then the controller parameters can be calculated as follows:

$$X = M^{-1}D \tag{A.9}$$

A.1.1 Order of R-S-T Controller

The dimension of the Sylvester matrix depends on the order of the controller. By choosing the proper controller order such that the Sylvester matrix is square, the controller parameter can be uniquely solved if $A(q^{-1})$ and $B(q^{-1})$ do not have any common factors. However, the non-uniqueness solution will be due to the orders of $S(q^{-1})$ and $R(q^{-1})$ where difference choices of ns and nr will give difference $S(q^{-1})$ and $R(q^{-1})$.

To condition the problem for a unique solution, the number of equation and number of unknown in Equation (A.8) are equated. While the number of unknowns is the number of coefficients in the compensator $S(q^{-1})$ and $R(q^{-1})$, the number of equations is simply the number of coefficients in the $D(q^{-1})$ or it is the order of $A(q^{-1})$ plus the order of $S(q^{-1})$. Therefore,

$$num - unknow = ns + nr + 1 \tag{A.10}$$

$$num - equation = n + ns \tag{A.11a}$$

$$num - equation = nd \tag{A.11b}$$

Since both the plant and controller transfer function are assumed to be proper, the order of the $D(q^{-1})$ can be represented by nd = n + ns. By equating the number of equations in Equation (A.11a) to the number of unknowns, the following relationship is obtained

$$ns - nr + 1 = n + ns \tag{A.12}$$

$$nr = n - 1 \tag{A.13}$$

$$ns \geq nr \tag{A.14}$$

Hence, for a unique solution, the order of $R(q^{-1})$ must be one less than the order of $A(q^{-1})$, ie. $nr = n - 1$. Meanwhile, properness of the controller must be maintained by satisfying $ns \geq nr$.

The fact that ns is cancelled in Equation (A.12) implies that $S(q^{-1})$ can be chosen arbitrarily provided that the controller is proper. From Equation (A.5) ns depends on the value of d, such as

$$ns \le m + d + 1 \tag{A.15a}$$

From Equation (A.11b), $nd - n = ns$, then

$$nd \le n + m + d - 1 \tag{A.15b}$$

Given the 4[th] order plant, for example, if d = 2, then. The Sylvester matrix will be squared with the dimension of r x r, ie $r = n + m - 9$, for 3[rd] order of $R(q^{-1})$ and a 4[th] order of $S(q^{-1})$.

A.2 Regulation and Tracking

The closed loop poles which are the roots of the polynomial $D(q^{-1})$ defines the regulation performances of control system. $D(q^{-1})$ is specified directly from the desired performance which is the disturbance rejection speed of the null reference output response. By doing so, $D(q^{-1})$ can be written in the form of binomial as follows

$$D(q^{-1}) = (1 - \alpha q^{-1})^{nd} \tag{A.16}$$

Where a is the stable poles of the discrete time close-loop system and ensuring the condition

$$0 < \alpha < 1 \tag{A.17}$$

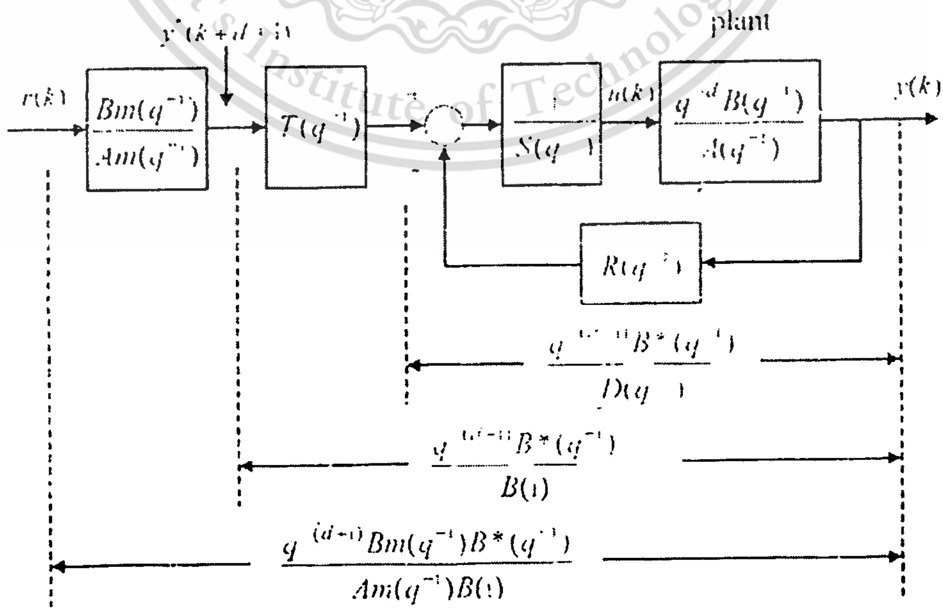

Figure A.2 Regulation and tracking performance.

Then, the controller $S(q^{-1})$ and $R(q^{-1})$ can be designed under the desired regulation performance. When the reference in put changes, it is desired that the system output follows a desired trajectory $y^*(k)$ .This trajectory may be stored or generated each time the reference is changed by using a dynamic model, this is known as "tracking", see figure A.2. the dynamic model $\frac{Bm(q^{-1})}{Am(q^{-1})}$. can be determined from desired performances (rise time, overshoot and settling time) by passing the step function, for example, through a second order normalized continuous time model and transforming into pulse transfer function. To approach the reference trajectory and the delay d can be compensated, then

$$y^*(k) = \frac{q^{-(d+1)}Bm(q^{-1})}{Am(q^{-1})}r(k). \tag{A.18}$$

where

$$Bm(q^{-1}) = bm_0 + bm_1q^{-1} + bm_2q^{-2} + \ldots$$
$$Am(q^{-1}) = 1 + am_1q^{-1}am_2q^{-2} + \ldots$$

and $q^{-d}B(q^{-1})$ should be rewritten as $q^{-(d+1)}B^*(q^{-1}), B(q^{-1}) = q^{-1}B^*(q^{-1})$

For this, first $y^*(k+d+1)$ is generated from $r(k)$ as shown below

$$y^*(k+d+1) = \frac{Bm(q^{-1})}{Am(q^{-1})}r(k). \tag{A.19}$$

Ideal tracking is achieved by selecting the pre-filter $T(q^{-1})$ properly to ensure a unitary steady stae gain of the overall system and the compensation of the regulation dynamics $T(q^{-1})$.

This leads to the choice:

$$T(q^{-1}) = GD(q^{-1}) \tag{A.20}$$

In which

$$G = \begin{cases} \frac{1}{B(1)}; B(1) \neq 0 \\ 1 \end{cases} \tag{A.21}$$

The control law equation becomes

$$S(q^{-1})u(k)+R(q^{-1})y(k)=T(q^{-1})y^{*}(k+d+1) \tag{A.22}$$

Then

$$u(k)=\frac{T(q^{-1})y^{*}(k+d+1)-R(q^{-1})y(k)}{S(q^{-1})} \tag{A.23}$$

which yields the complete pole placement as shown in figure 2.2. The transfer function between the reference and the output is

$$H_{CL}(q^{-1})=\frac{q^{-(d+1)}Bm(q^{-1})}{Am(q^{-1})}\times\frac{B^{*}(q^{-1})}{B(1)} \tag{A.24}$$

A.3 Tracking and Regulation with Independent Objectives or MRAC

In some case the designer needs to obtain the desired tracking performance (changing of the rejection of disturbance reference or noise) independent of the desired regulation performance (changing of the reference). For example, it is necessary in a situation for which the desired regulation response time is significantly smaller than the desired response time for a change in reference or vice versa. This case results in the simplification of the zero $B(q^{-1})$ of the discrete time plant model. By doing so, the strategy can only be applied to discrete time models with stable zeros which implies that the plan must be a minimum phase. The structure of this method is shown in figure A.3.

A.3.1 Regulation

The regulation performance is specified by $D(q^{-1})$. To compensate $B^{*}(q^{-1})$, $S(q^{-1})$should has $B^{*}(q^{-1})$and in order to compute $R(q^{-1})$and $S(q^{-1})$, the following equation must be solved

$$B^{*}(q^{-1})D(q^{-1})=A(q^{-1})S(q^{-1})+q^{-(d+1)}B^{*}(q^{-1})R(q^{-1}). \tag{A.25}$$

In this equation, $S(q^{-1})$should has $B^{*}(q^{-1})$

$$S\left(q^{-1}\right)=B^{*}\left(q^{-1}\right)S'\left(q^{-1}\right) \tag{A.26}$$

In which

$$S'\left(q^{-1}\right)=1+s_1'q^{-1}+\ldots+s_d'q^{-d} \tag{A.27}$$

From Equation (A.25) and (A.26), one obtains

$$D\left(q^{-1}\right)=A\left(q^{-1}\right)S'\left(q^{-1}\right)+q^{-(d+1)}R\left(q^{-1}\right) \tag{A.28}$$

This equation has unique solution for

$$\deg D\left(q^{-1}\right)\leq n+d, \deg S'\left(q^{-1}\right)=d, R\left(q^{-1}\right)=1 \tag{A.29}$$

In which

$$R\left(q^{-1}\right)=r_0+r_1q^{-1}+\ldots+r_{nr}q^{-nr}$$

A.3.2 Tracking

From figure A.3 the tracking performance is define by the transfer function

$$\frac{y(k)}{r(k)}=\frac{q^{-(d+1)}Bm\left(q^{-1}\right)}{Am\left(q^{-1}\right)}. \tag{A.30}$$

The output of the tracking (dynamic) model specifies the desired trajectory $y^{*}(k)$ with d + 1 steps in advance to accomplish the asymptotically tracking. To ensure this, the prefilter will be selected as

$$T\left(q^{-1}\right)=D\left(q^{-1}\right) \tag{A.31}$$

Then, from figure 2.3, the controller equation can be obtained:

$$S\left(q^{-1}\right)u(k)+R\left(q^{-1}\right)y(k)=D\left(q^{-1}\right)y^{*}(k+d+1) \tag{A.32}$$

And $u(k)$ can be written as

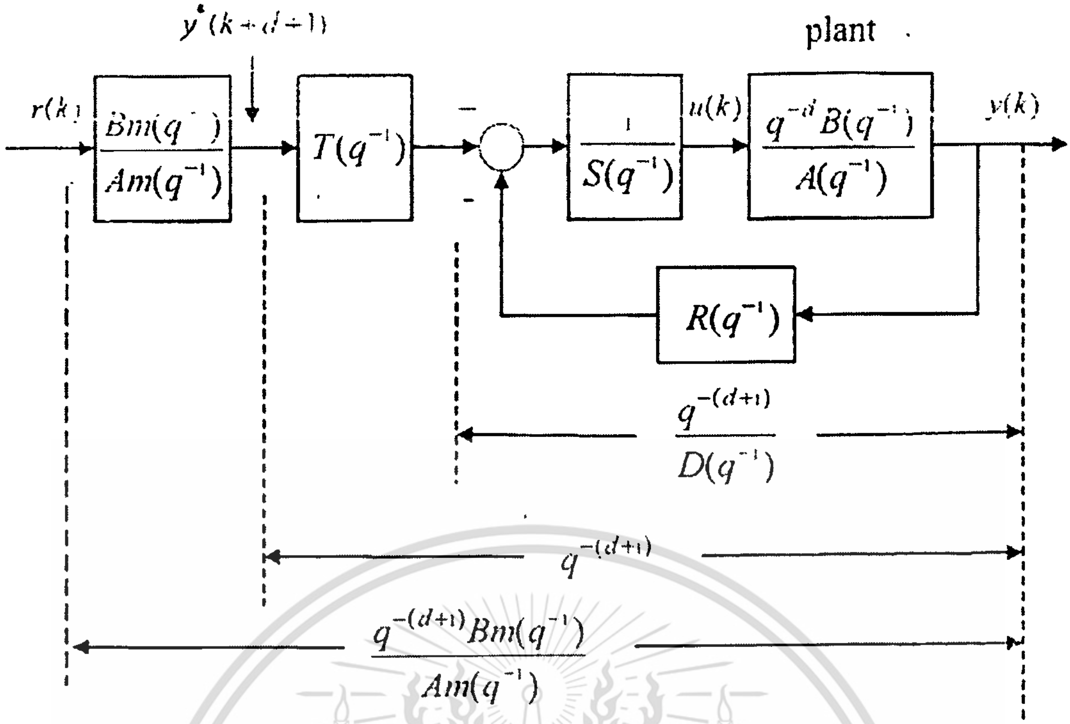

Figure A.3 Regulation and tracking with independent objectives.

$$u(k)=\frac{D\left(q^{-1}\right)y^{*}(k+d+1)-R\left(q^{-1}\right)y(k)}{S\left(q^{-1}\right)} \tag{A.33}$$

A.4 Conclusion

This appendix presents digital control design. The controller has a tri-branches structure known as R-S-T canonical form, in which the control law equation is

From the pole placement strategy, the digital controller is designed in two steps

1. the computation of the polynomials $S\left(q^{-1}\right)$ and $R\left(q^{-1}\right)$ through the Diophantine equation in order to match the desired regulation performances.
2. The computation of the pre-filter $T\left(q^{-1}\right)$ in order to approach the desired tracking performances.

The orders of $S\left(q^{-1}\right)$ and $R\left(q^{-1}\right)$ polynomials are depend only on the complexity of the plant model.

The tracking and regulation with independent objectives design strategy (MRAC) can exactly match the desire tracking and the regulation performance, but the discrete time plant model must have stable zeros.

Appendix B

B.1 Deterministic signals

B.1.1 Offset

The simplest kind of signal is a constant value d. A system with offset d be represented by the equation

$$s(t) = d \tag{B.1}$$

When present in a control system, the offset is removed by integral action in a feedback controller, but some self tuning schemes estimate d and subtract out its effect by modifying the control signal. In a signal processing application, a constant signal level is often present due to the way in which the sensing instrumentation is designed. In addition, a constant signal is often used as a test of system functionality when no other signal is present.

B.1.2 Drift

A generalization of the constant offset is the drift signal where the offset becomes a function of time. in many situations drift can be modeled by a polynomial function of time D(t).

$$s(t) = D(t) = d_0 + d_1 t + \ldots + d_n t^n . \tag{B.2}$$

A first order drift polynomial (nd = 1) will cover many situations, especially the common case where the output offset is slowly changing at a constant rate. Also it is convenient to assume that D(t) is filtered by the $A(q^{-1})$ object of equation (B.3), so that the effective drift is $D(t)/A(q^{-1})$.

$$\begin{aligned} B(q^{-1}) &= b_0 + b_1 q^{-1} + \ldots + b_m q^{-m} \\ A(q^{-1}) &= 1 + a_1 q^{-1} + \ldots + a_n q^{-n} \end{aligned} \tag{B.3}$$

A further form of time dependent drift occurs when the offset is periodic. Periodic drift can be represented in a model by the appropriate periodic function (e.g. sinusoidal, square wave, triangular wave). For example, a sine wave drift can be modeled as

$$D(t) = d_s \sin(\Omega Tt) + d_c \cos(\Omega Tt)\,. \tag{B.4}$$

The parameter Ω determines the period of the oscillation and the quantities ds and dc jointly determine the amplitude and phase. In self tuning schemes ds, dc are usually unknown but the period Ω is known. The self-tuning algorithm must then determine ds,dc the influence of $D(t)$ can then be measured and removed by modulating the control signal u(t) in a control situation. In a signal processor a model of $D(t)$ can be subtracted from the signal $s(t)$. The situation is more difficult if the parameter Ω is unknown.

A deterministic signal may not be an explicit function of the time index, (although this is usual) but may be expressed in terms of some other independent variable. For example, if the offset is known to vary as a linear function of a variable w (which might be the operating temperature of a process or attitude of an aircraft), then an appropriate offset model is

$$s(w) = D(w) = d_0 + d_1 w + \ldots + d_n w^n \tag{B.5}$$

B.1.3 Measurable signal sources

Often, system output or signal source is influenced by a known source $v_c(\tau)$ via a disturbance transfer function. The disturbance can be represented by a discrete time transfer function acting upon a sequence $v(t)$ representing a sampled version of the known source. Thus a measurable disturbance is represented by

$$s(t) = \frac{D}{A} v(t) \tag{B.6}$$

where

$$D = d_0 + d_1 q^{-1} + \ldots + d_{nd} q^{-nd}$$
$$A = 1 + a_1 q^{-1} + \ldots + a_{na} q^{-na}$$

Note the following points concerning $D = d_0 + d_1 q^{-1} + \ldots + d_{nd} q^{-nd}$:

(a) Some of the leading coefficients $d_0, d_1 \ldots$ may be zero, corresponding to a time delay in the disturbance transfer function.

(b) Unless the continuous time signal $v_c(\tau)$. is z transformable, then the disturbance representation (B.6) is theoretically in correct. In practice, however, the errors caused are small compared with the improved performance obtained by modeling the disturbance.

A known disturbance is usually removed in a conventional control scheme by feed forward control. This is also the case in self tuning with the added feature that the coefficients of D are estimated and may be used to subtract out the influence of $v(t)$ by modifying u(t). A similar subtraction procedure is used in signal processors for interference cancellation. In this situation the signal $v(t)$ represents some measurable function of a process which is interfering with a desired signal.

B.1.4 Random signals

An important class of random signals is that associated with small unpredictable changes in a system and unobservable noise like disturbances. Such disturbances can be aggregated and modeled by a single noise source which is often assumed to be stationary, gaussian noise sequence. Such a sequence can be represented by a white guassian noise sequence $e(t)$ with zero mean and variance δ_e^2 which has been passed through a stable linear time invariant filter $N(q^{-1})$.

If $s(t)$ is the noise sequence as measured at the output of N then $s(t)$ will have a power spectrum (or spectral density function) given by

$$S_{ss}(\omega) = N(\exp(j\omega))N(\exp(-j\omega))\delta_e^2 \tag{B.7}$$

Note that q is evaluated on the unit circle $q = \exp(-j\omega)$.

Since $S_{ss}(\omega)$ is the only quantity we can measure which will give us information about N, we are free to construct any N which will match the power spectrum. If the power spectrum can be approximated by a rational polynomial function of $\cos(\omega)$, then it is always possible to obtain a noise transfer function N which is stable (all poles inside the unit circle) and has all its zeroes on or inside the unit circle. Zero on the unit circle are not interest here and in their absence the noise transfer function will be inverse stable (or minimum phase by analogy with the corresponding term for continuous time systems), with all zeroes in the left half-plane.

In general, a stationary random signal source is represented by the stable transfer function model:

$$s(t) = \frac{C}{A} e(t) \tag{B.8}$$

where

$$C = c_0 + c_1 q^{-1} + \ldots + c_{nc} q^{-nc}$$
$$A = 1 + a_1 q^{-1} + \ldots + a_{na} q^{-na}$$

The above model for a random process is termed an auto regressive moving average (ARMA) model.

The stationary noise model of equation B.8 is not always sufficient to represent random drift disturbances which influence some systems. In such circumstances, a random noise model is postulated in the form of a stationary increment to $s(t)$:

$$s(t) = s(t-1) + \frac{C}{A} e(t) \tag{B.9a}$$

or

$$s(t) = \frac{C}{\Delta A} e(t) \tag{B.9b}$$

Where $\Delta = 1 - q^{-1}$, the first difference operator. The incremental assumption builds an integrator into the noise model and consequently this form is termed an auto regressive integrated moving average (ARIMA) representation. In the special case $C = 1 = A$, the signal is an integral of the white noise and is often referred to random walk or Brownian motion.

B.1.5 Overall signal model

The representations of deterministic, random and measurable disturbances can be drawn together to form a composite signal model which will meet most signal representation situations. Specifically, by combining the equations (B.2), (B.6) and (B.8) we can write:

$$s(t) = \frac{D(t)}{A} + \frac{D}{A}v(t) + \frac{C}{A}e(t) \tag{B.10}$$

Using this model it is possible to represent almost any signal form which might be encountered in self-tuning studies.

Appendix C

C.1 Probability

An event A is any outcome of interest from an experiment. The probability $P(A)$ (or prob (A)) of this event can be thought of as the limit, as the experiment is repeated indefinitely, of the ratio of the number of times A occurs to the number of trials. This definitely implies that

$$0 \leq p(A) \leq 1 \tag{C.1}$$

If $A_1, A_2, \ldots,$ are events, no two of which can occur together (mutually exclusive), then the event that A_1 or A_2 or … occurs is given by

$$P(A_1 \text{ or } A_2 \text{ or} \ldots) = P(A_1) + P(A_2) + \cdots \tag{C.2}$$

and it assumed that

$$P(A_1) + P(A_2) + \cdots = 1 \tag{C.3}$$

if the list of events covers all possible outcomes of the experiment.

If two outcomes A, B do not preclude each other, then

$$P(A \text{ or } B) = P(A) + P(B) + P(A \text{ and } B) \tag{C.4}$$

in an obvious notation. Clearly the last term vanishes for exclusive events to yield a special case of (C.2).

A key concept in estimation theory is that of conditional probability. The probability of A occurring, given that B has occurred, is denoted by $P(A|B)$ and is defined by

$$P(A \text{ and } B) = P(A|B)P(B) \tag{C.5}$$

If A, B are independent events, so that the occurrence of B has no effect on the likelihood of A occurring, then we can replace $P(A|B)$ by the unconditional probability $P(A)$ so that C.5 becomes

$$P(A \text{ and } B) = P(A)P(B) \quad \text{(C.6)}$$

The event 'A and B' is the same as 'B and A' and (C.5) leads to the useful relationship

$$P(B|A) = P(A|B)P(B) / P(A) \quad \text{(C.7)}$$

C.2 Random Variables

The events generally of interest to engineers are the occurrence of specific numerical values of physical variables (temperature, voltage, etc). A variable whose value is determined by the (uncertain) outcome of an experiment is called a random variable (abbreviated to *rv*). The probabilistic behavior of a *rv* X is completely specified by its probability distribution function (*pdf*) $F_X(x)$ or its probability density function $f_X(x)$ defined by

$$F_X(x) = P(X \le x) = \int_{-\infty}^{x} f_x(u)du \quad \text{(C.8)}$$

so that $f_X(x)$ is the gradient of $F_X(x)$. It follows that

$$\int_{-\infty}^{\infty} f_x(x)dx = 1 \quad \text{(C.9)}$$

Note that $F_X(x)$ is a probability but that $f_X(x)$, although positive, can take large values. It also follows that

$$P(x \le X \le x + \delta x) \approx f_x(x)\delta x \quad \text{(C.10)}$$

for small δx, provided the density function is bounded. This is true of a continuously distributed *rv* for which the probability is zero of any specific value occurring. If X takes a set of discrete values of $x_1, x_2, \ldots,$ with probabilities $p(x_1), p(x_2), \ldots,$ then the density function can be cast in the form

$$f_X(x) = \sum_i p(x_i)\delta(x - x_i) \tag{C.11}$$

where $\delta(\cdot)$ is the direct delta function. In general, a density function may be partly continuous and partly of the blip structure (C.11)

For two *rvs* X, Y the probabilistic structure is specified by the *joint pdf* $F_{XY}(x,y)$ or joint density $f_{XY}(x,y)$ defined by

$$F_{XY}(x,y) = P(X \le x \text{ and } Y \le y) = \int_{-\infty}^{x}\int_{-\infty}^{y} f_{XY}(u,v)\,du\,dv \tag{C.12}$$

It follows that, for X alone,

$$F_X(x) = F_{XY}(x,\infty); \qquad f_X(x) = \int_{-\infty}^{\infty} f_{XY}(x,y)\,dy \tag{C.13}$$

and similarly for Y. the extension to more than two *rvs* is straightforward. If X, Y are independent *rvs*, then Equation (C.5) and (C.12) imply that

$$F_{XY}(x,y) = F_X(x)F_Y(y); \qquad f_{XY}(x,y) = f_X(x)f_Y(y) \tag{C.14}$$

C.3 Expectations and Moments

The expectation (average, mean) or first moment of a *rv* X is defined as the weighted average

$$E_X(X) = m_X = \int_{-\infty}^{\infty} x f_X(x)\,dx \tag{C.15}$$

A measure of the dispersion or size of fluctuation of X around its mean value is given by the second central moment or *variance* σ_{XX} (σ_X^2 or *var* X) defined by

$$\sigma_{XY} = E_X[X - E_X[X]]^2 = E_X[X^2] - (E_X[x])^2 \tag{C.16}$$

The standard error (or standard deviation) σ_X is defined as the positive square root of the variance and has the same dimensions as X. the moments give useful but only partial information about the pdf in general.

A useful (but often conservative) relationship is given by the Chebyhev inequality

$$P(|X| \geq \varepsilon) \leq \frac{1}{\varepsilon^2} E_X[X^2] \quad (C.17)$$

where ε is any positive constant.

The extension to moments involving more than one *rv* is straightforward, by simply calculating the weighted average using the appropriate joint density. This leads to

$$E_{XY}[g(X)+h(Y)] = E_X[g(X)] + E_Y[h(Y)] \quad (C.18)$$

and, if X, Y are independent,

$$E_{XY}[g(X)h(Y)] = E_X[g(X)]E_Y[h(Y)] \quad (C.19)$$

where g, h are arbitrary functions (subject to the existence of the moments). A useful second order moment that gives some indication of the degree to which X and Y are related is the covariance

$$\sigma_{XY} = E_{XY}[(X-m_X)(Y-m_Y)] = E_{XY}[XY] - m_X m_Y \quad (C.20)$$

The *rvs* are uncorrelated if σ_{XY} is zero and it is easy to see from (C.19) that this is implied by independence. Uncorrelated *rvs*, however, are not necessarily independent.

A useful measure of correlation is given by the dimensionless scalar correlation coefficient

$$p_{XY} = \sigma_{XY} / \sigma_X \sigma_Y \quad (C.21)$$

which always lies between -1 and +1, taking one of the extreme values if X and Y are linearly related.

Generalizing (C.20) to the case of n *rvs* $X = (X_1, X_2, \ldots, X_n)^T$ leads to the covariance matrix $(\mathrm{cov}\, X)$ whose (i, j) element is the covariance between X_i and $X_j (i \neq j)$ and whose (i, i) diagonal element is var X_i.

C.4 Stochastic Convergence

The estimates generated by a recursive estimation algorithm form a sequence of *rvs* and it is usually of interest to know how the sequence behaves and in particular whether it converges in some sense. In general, it is too restrictive to demand that every realization of the sequence converges, but there are a number of weaker forms of convergence, two of which are relatively easy to use.

The sequence {x(t)} tends to the *rv* x in the mean-square sense (ms) if

$$E|x(t) - x|^2 \to 0 \qquad \text{as} \quad t \to \infty \tag{C.22}$$

The sequence {x(t)} tends to the *rv* x in probability if, for any positive, ε,

$$P[|x(t) - x| > \varepsilon] \to 0 \qquad \text{as} \quad t \to \infty \tag{C.23}$$

written

$$P \lim x(t) = x \qquad \text{as} \quad t \to \infty \tag{C.24}$$

Using the Chebyshev inequality (C.17) it follows that ms convergence implies convergence in probability. The converse is not true.

AUTHOR BIOGRAPHY

Name – Family Name	Ngantawee Rujirakamthornchai
Birth date	July 23, 1980
Address	410 Charansanitwong Road Bangplad Bangkok 10700
	Tel. 4353554
Educations	2002 Bachelor Degree of Control Engineering from King Mongkut's Institute of Technology Ladkrabang
Skilled	General control system
	Electrical and automation design
	Process control
Work Experiences	2002 – 2005 Service Engineer
	2005 – 2007 Service Manager in the company involving food process industrial and machinery.