STATE-SPACE LQG SELF-TUNING CONTROL OF FLEXIBLE STRUCTURES

by

Fusheng Ho

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

APPROVED:

William T. Baumann, Chairman

Hugh F. VanLandingham

Harry H. Robertshaw

John S. Bay

Harley H. Cudney

April, 1993

Blacksburg, Virginia

STATE-SPACE LQG SELF-TUNING CONTROL OF FLEXIBLE STRUCTURES

by

Fusheng Ho

Department of Electrical Engineering

William T. Baumann, Chairman

(ABSTRACT)

This dissertation presents a self-tuning regulator (STR) design method developed based upon a state-space linear quadratic Gaussian (LQG) control strategy for rejecting a disturbance in a flexible structure in the face of model uncertainty.

The parameters to be tuned are treated as additional state variables and are estimated recursively together with the system state that is needed for feedback. Also, the feedback gains are designed in the LQ framework based upon the estimated model parameters.

Two problems concerning the uncertainty of model parameters are recognized. First, we consider the uncertainty in the system matrix of the state-space model. The self-tuning regulator is implemented by computer and the control law is obtained based upon a discrete-time model; however, only selected continuous-time parameters with physical meanings to which the controller is highly sensitive are tuned. It is formulated as a nonlinear filtering problem such that both the estimated state and the unknown parameters can be obtained by an extended Kalman filter. The capability of this design method is experimentally demonstrated by applying it to the rejection of a disturbance in a simply-supported plate.

The other problem considered is that the location where the disturbance enters the system is unknown. This corresponds to an unknown disturbance influence matrix. Under the assumption that the system matrix is known and the disturbance can be measured, it is formulated as a linear filtering problem with an approximate discrete-time design model. Similarly, the estimated state for feedback and the unknown parameters are identified simultaneously and

recursively. Also, the feedback gains are calculated approximately by recursively solving the discrete-time control Riccati equation. The effectiveness of the controller is shown by applying it to a simply-supported plate, when the location of the disturbance is assumed unknown.

Since implementing LQG self-tuning controllers for vibration control systems requires significant real-time computation, methods that can reduce the computing load are examined. In addition, the possibility of extending the self-tuning to disturbance model parameters is explored.

Acknowledgments

I would like to express my sincere appreciation and thanks to my advisor, Dr. William T. Baumann, for his help in every aspect of my work toward the Ph.D. degree. It is lucky to have his guidance and I have benefited greatly from his advise.

Dr. Hugh F. VanLandingham has given me invaluable support at the most difficult time of my graduate studies, for which I present my cordial gratitude.

This work was supported by the Office of Naval Research under Grant No. N00014-92-J-1170. I am deeply grateful to Dr. Harry H. Robertshaw and Dr. William T. Baumann for offering me the opportunity to work on this project.

I also would like to thank Dr. John S. Bay, Dr. Harley H. Cudney and the rest of my committee for their time and efforts in making suggestion on behalf of my dissertation.

I want to dedicate this dissertation to my dearest family. Without the love and support from my parents, who always put children's education in the first priority, my education would have never been possible. The love and care of my wife, Hua-Fu, directly contribute to my work and my life. The constant encouragement from the parents of my wife is also very precious to me.

Contents

List of Figures

List of Tables

1. Introduction

1.1 Motivation of This Study

The modern control theory method of linear-quadratic-Gaussian (LQG) design, which stems from the combination of the solution to the linear optimal control problem and innovations made by Kalman and Bucy[1] in filtering theory, was developed in the sixties and was quite successful in the control of space vehicles. The optimal control problem and the optimal filtering problem were formulated in state space, employing the mathematical tools of differential equations and vector space. Based upon a linear state-space model, Gaussian noise, and a quadratic performance index, an LQG controller is said to be optimal in the sense that by feeding back the state and relocating the poles of the system, it minimizes the performance index. The success of LQG control in the sixties can be attributed to its immediate applicability to multivariable systems and the availability of accurate models and measurements.

It has been shown[2] that a multivariable system with full state available for feedback has guaranteed 60° phase and 6 dB gain margins with an LQG controller. However, if full-state feedback is not possible and a state estimator is used such that the estimated state can be fed back, then the guaranteed margins no longer exist[3], which means that the LQG controller may be rather sensitive to the uncertainty and variation of the models. Many methods have been proposed to recover or enhance the robustness of LQG controllers[4-8]; however, performance of those controllers must be sacrificed in order to maintain stability in the presence of uncertainty. Adaptive control comes into play when high performance of the controllers is required.

Flexible structures are usually described by linear, finite-dimensional, time-invariant (LFDTI) models. A major problem of controlling flexible structures by feedback approaches is the difficulty in obtaining accurate models. While the LFDTI models of flexible structures are examples of highly uncertain plants, the LQG feedback controllers are particularly sensitive to only a few physical

parameters of the models.[8]

The physical plants to be controlled are continuous-time systems; however, the controllers usually need to be implemented by computer and to be designed based upon discrete-time models because of their complexities. Since a discrete-time model is obtained from a continuous-time model by a nonlinear transformation, the uncertainty about one physical parameter is smeared over the entire discrete-time model.

Adaptive control can be generally viewed as a combination of a recursive identifier and a specified control strategy. Adaptive controllers that are implemented by computers are usually designed to identify discrete-time model parameters even though the systems to be controlled are continuous-time plants. Thus, the controllers need to assume that all the parameters in the discrete-time models are unknown and tune each one of them adaptively, although the continuous-time models may be partially known or only a small number of parameters is critical to the controller design.

If the control strategy adopted is the LQG design method, then it is usually necessary to estimate all the parameters in the discrete-time model, to reconstruct the state, and to calculate the optimal feedback gains based upon the estimated model in a single sampling period. All of these require intensive computation and make implementation of LQG controllers difficult, especially for systems of large dimension and fast sampling rate.

Since an LQG controller is particularly sensitive only to a small number of parameters in the continuous-time model and the model can be partly known, it would reduce the computing load if these continuous-time model parameters, instead of all the parameters of the discrete-time model, are tuned adaptively.

1.2 Research Objective

Our objective in this research is to explore methods of designing self-tuning regulators based upon the state-space LQG control strategy for structural vibration control. The focus is on reducing computing load to make the realization of a self-tuning LQG regulator possible. A major result is a self-tuning LQG

regulator that is able to selectively tune the parameters of a continuous-time model instead of all the parameters in a discrete-time model. The problems to be considered result from the variations of the most sensitive parameters in the system matrix of a continuous-time model and the uncertainty in the disturbance influence matrix. The basic idea is to treat parameters to be tuned as extra state variables and to use a state estimator to estimate the parameters and the state together.

Following this introduction chapter, feedback control of flexible structures by the optimal state-space LQG method is discussed in Chapter 2 which establishes the framework of our proposed methods for self-tuning controller design. The strengths of the LQG control method with an accurate model are shown and the stability characteristics of the closed-loop system with an incorrect model are investigated. The results suggest that only a small number of continuous-time model parameters need to be tuned adaptively.

In Chapter 3, a few basic system and signal models useful in control systems are introduced and several popular recursive estimation methods are reviewed in order to have a comparison made with the proposed methods.

A summary of several of the most applied adaptive control methods is included in Chapter 4 with a focus on the self-tuning regulators category, into which our proposed methods fit.

Presented in Chapter 5 is the proposed self-tuning design method that selectively tunes the most sensitive continuous-time model parameters, making use of the extended Kalman filter to solve a nonlinear filtering problem.

In order to reduce the computing load of the LQG self-tuner, Chapter 6 is devoted to a method that can simplify the algorithm in Chapter 5 and still maintain desired performance.

The problem of an unknown disturbance matrix is separated from the other problem and is dealt with in Chapter 7. Based upon the assumption that the disturbance influence matrix is the only unknown in the model and the disturbance can be measured, approximations are made on the discrete-time model such that the unknown parameters can be estimated by solving a linear filtering problem.

1.3 Contributions

The most important contribution of this research is the exploitation of the extended state model concept in designing LQG self-tuning regulators for partly unknown systems.

When the vibration control of a flexible structure is considered, the proposed LQG self-tuner is aimed at selectively tuning particularly sensitive parameters in the continuous-time system matrix. While an LQG self-tuning regulator requires intensive computation, this result surely reduces much of the computing load. In order to save more computing effort, a simplified algorithm for the Kalman filter is developed based upon the fact that part of the error covariance matrix monotonically reaches steady state and can be fixed to the steady state value. The validity of the controllers are demonstrated experimentally.

The computational method of obtaining a ramp invariant (RI) discrete-time model, developed by Bingulac and VanLandingham[9], is modified such that the linear relation between the quadruple of matrices of the continuous-time model and the quadruple of matrices of the discrete-time RI model is made explicit. This result allows us to use a linear estimator in dealing with the uncertainty in the disturbance influence matrix which occurs when the location where the disturbance enters the system is modeled. Under the assumption that the disturbance can be measured, the controller is shown to be effective if the sampling rate is much higher than the modeled modes and the disturbance frequencies.

2. Optimal Feedback Control of Flexible Structures

One of the effective approaches to active vibration control of flexible structures is via a feedback strategy. Mechanically flexible structures, which require an infinite number of modes to completely describe their behavior, are usually modeled by large finite-dimensional systems. Feedback control is aimed at precisely controlling a few critical modes of a large dimensional system with a controller of much smaller dimension[10]. In this chapter, the focus is on rejecting a point force disturbance in flexible structures by the method of linear optimal state feedback control theory combined with a Kalman filter state estimator. The disturbance is assumed to be low-frequency harmonic or narrowband, and the control force is of point force type. A discussion of the LQG regulator design and the digital implementation of the controller is provided. Effectiveness of the design method is shown by an example of rejecting a disturbance in a simply-supported plate. Both simulated and experimental results are presented. The issues of performance and stability are addressed and the tradeoff of the design is investigated at the end of the chapter.

2.1 LQG Regulator Design

Consider a general flexible structure which can be modeled by a first-order differential equation. Subject to the influence of the disturbance $d \in R^s$ and controlled by the actuator input $u_p \in R^m$, the state-space representation of the system is

$$\dot{x} = F\ x + G\ u_p + L\ d\ , \tag{2.1.1}$$

where $x \in R^{2n}$ is the state of the system in modal coordinates, n is the number of modes that are modeled, F is the *system matrix*, and we call G the *control influence matrix* and L the *disturbance influence matrix.*

The controlled output is chosen to be the modal accelerations and the output measurement equation, which contains direct feedthrough of the control and disturbance forces, is given by

$$y = C\ x + D\ u_p + E\ d + \theta\ , \tag{2.1.2}$$

where $y \in R^n$ is a vector of measured modal accelerations and $\theta \in R^n$ is a zero mean Gaussian measurement noise vector.

By the *factorization theorem* and the *representation theorem*[11], all stochastic processes with rational spectral densities can be thought of as generated from dynamical systems whose inputs are white noises. Thus, the disturbance input, which is assumed to be a colored noise source, is modeled by the output of a shaping filter driven by a Gaussian white noise. The frequency content of the disturbance will be determined by the dynamics of the shaping filter. The general state-space model of the shaping filter is

$$\dot{r} = F_w\ r + G_w\ v \tag{2.1.3}$$

$$d = C_w\ r\ , \tag{2.1.4}$$

where $r \in R^w$ is the state of the shaping filter, $v \rightarrow N(0, Q_c)$ is the driving noise source and d is the disturbance input colored noise source.

When a much smaller order controller with a finite number of control actuators and sensors is used to control a large order system, spillover is likely to be induced. The phenomenon of spillover, in which the energy intended to go into the controlled modes is also pumped into the uncontrolled modes, can lead to instabilities in the closed-loop system[12].

In order to overcome the problem of control spillover, a smoothing filter is augmented to the system to filter out the frequency content of the control signal that is not in the spectrum of the critical modes to be controlled. Assuming that the first few modes are most critical, a low-pass filter with corner frequency properly chosen will meet our requirements. Thus, the combined system shows low-pass characteristics and if the disturbance is constrained to be in the low

frequency region, the uncontrolled modes won't be excited by either control signal or disturbance signal and the instabilities from control and estimation spillover can be eliminated.

Let the dynamics of the smoothing filter be modeled by the equation

$$\dot{s} = F_{lp}\, s + G_{lp}\, u \tag{2.1.5}$$

$$u_p = C_{lp}\, s\ , \tag{2.1.6}$$

where $s \in R^t$ is the state of the smoothing filter, u is the input signal into the combined system and u_p is the actual control force on the structure.

Combining the dynamics of the structure, the disturbance and the smoothing filter results in an augmented model

$$\begin{bmatrix} \dot{x} \\ \dot{s} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} F & GC_{lp} & LC_w \\ 0 & F_{lp} & 0 \\ 0 & 0 & F_w \end{bmatrix} \begin{bmatrix} x \\ s \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ G_{lp} \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ G_w \end{bmatrix} v$$

$$= F_a\, x_a + G_a\, u + L_a\, v \tag{2.1.7}$$

$$y = \begin{bmatrix} C & DC_{lp} & EC_w \end{bmatrix} \begin{bmatrix} x \\ s \\ r \end{bmatrix} + \theta$$

$$= C_a\, x_a + \theta\ . \tag{2.1.8}$$

The quadratic cost function to be minimized is

$$J = \lim_{T \to \infty} \frac{1}{2T}\, \mathbb{E} \left\{ \int_{-T}^{T} (x_a{}^{\mathrm{T}}\, Q\, x_a + u^{\mathrm{T}}\, R\, u)\, dt \right\}, \tag{2.1.9}$$

where Q is a positive semi-definite *state weighting matrix* and R is *a control weighting matrix* that penalizes the amount of control effort used.

Based upon the cost function, the optimal control law has the form of state feedback

$$u = -K_{lq}\, x_a \,, \tag{2.1.10}$$

where $K_{lq} \in R^{m \times (2n+t+w)}$. The effect of the state feedback is to relocate the poles of the closed-loop system while minimizing the cost function and the minimized cost represents a compromise between the performance and the control effort used.

Since no state is measured, a full state estimator is required. The Kalman-Bucy filter, which is optimal in the presence of the specified measurement noise, is used and the equation is given by

$$\dot{\widehat{x}}_a = F_a\, \widehat{x}_a + G_a\, u + K_f(y - \widehat{y}) \,, \tag{2.1.11}$$

where K_f is the Kalman filter gain and $\widehat{x}_a$ is the estimated state and $\widehat{y} = C_a \widehat{x}_a$ is the estimated output. Based upon the *certainty equivalent principle*[11], the estimated state $\widehat{x}_a$ is used in Eq. (2.1.10) for x_a .

2.2 Digital Implementation of LQG Regulators

Because of the advances in digital technology, implementation of complicated controllers becomes possible. Although many methods can be used to obtain a digital controller that will approximate the performance of an existing analog controller, a digital controller designed in the discrete-time domain can take full advantage of digital control. This section is devoted to LQG regulator design in the discrete-time domain based upon discrete-time models.

One of the features in computer control systems is the analog-to-digital(A/D) and digital-to-analog(D/A) conversions involved. The A/D and D/A operations are usually implemented by the so called zero-order-hold(ZOH) device

and the whole system can be described by a ZOH discrete-time model. Assuming that the sampling period is T, this model can be derived from the continuous-time model as[13]

$$x_{a,\,k+1} = dF_a\, x_{a,\,k} + dG_a\, u_k + v_k \quad , \tag{2.2.1}$$

where

$$dF_a = \Phi(T) = \; e^{F_a\, T} \tag{2.2.2}$$

$$dG_a = \Gamma(T) = \int_0^T e^{F_a\, \xi}\, G_a\, d\xi \tag{2.2.3}$$

$$v_k = N(0, Q_d)\ , \tag{2.2.4}$$

and

$$Q_d = \int_0^T \Phi(\xi)\, L_a\, Q_c\, L_a{}^{\mathrm{T}}\, \Phi^{\mathrm{T}}(\xi)\, d\xi\ . \tag{2.2.5}$$

The continuous-time cost function J in Eq.(2.1.11) can be written as

$$J = \sum_{k\,=\,-\infty}^{\infty} \frac{1}{2T}\, \mathbb{E}\left\{ \int_{kT}^{(k+1)T} (x_a{}^{\mathrm{T}}\, Q\, x_a + u^{\mathrm{T}}\, R\, u)\, dt \right\}, \tag{2.2.6}$$

and a solution of Eq.(2.1.7) is given by

$$x_a(kT+\tau) = \Phi(\tau)\, x_a(kT) + u(kT) \int_0^\tau \Phi(\xi)\, G_a\, d\xi + \widetilde{v}\,(k,\tau)\ , \tag{2.2.7}$$

where

$$\widetilde{v}\,(k,\tau) = \int_0^\tau \Phi(\xi)\, L_a\, v(kT+\tau-\xi)\, d\xi\ . \tag{2.2.8}$$

Substitution of Eq.(2.2.7) into Eq.(2.2.6) gives the discrete-time cost

function

$$J = \sum_{k=-\infty}^{\infty} \frac{1}{2T}\, \mathbb{E}\{\, [x_{a,k}^{\mathrm{T}} \;\; u_k^{\mathrm{T}}] \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} x_{a,k} \\ u_k \end{bmatrix} \; + \; V \} \;, \tag{2.2.9}$$

where

$$\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} = \int_0^T \begin{bmatrix} \Phi^{\mathrm{T}}(\tau) & 0 \\ \Gamma^{\mathrm{T}}(\tau) & I \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \Phi(\tau) & \Gamma(\tau) \\ 0 & I \end{bmatrix} d\tau \tag{2.2.10}$$

$$\begin{aligned} V = 2 \int_0^T & \widetilde{v}^{\,\mathrm{T}}(k,\tau) \,[\; Q\; \Phi(\tau)\; x(kT) + Q\; u(kT)\; \Gamma(\tau) \;\;] \; d\tau \\ & + \int_0^T \;\; \widetilde{v}^{\,\mathrm{T}}(k,\tau)\; Q\; \widetilde{v}\;(k,\tau)\; d\tau \;. \end{aligned} \tag{2.2.11}$$

Since the expected value of the first integral in the right-hand side of Eq.(2.2.11) vanishes and the expected value of the second integral is independent of the optimal values of x_a and u, the cost function for the optimization problem is equivalent to

$$J_e = \sum_{k=-\infty}^{\infty} \frac{1}{2T}\, \mathbb{E}\{\, [x_{a,k}^{\mathrm{T}} \;\; u_k^{\mathrm{T}}] \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} x_{a,k} \\ u_k \end{bmatrix} \} \;. \tag{2.2.12}$$

While this equivalent cost function has cross weighting terms, transformations of variables can be made in the cost function and the state equation so that the cross terms disappear. The transformed system is given by

$$x_{a,k+1} = dF_{ae}\; x_{a,k} + dG_a\; u_{e,k} + v_k \tag{2.2.13}$$

$$J_e \;=\; \sum_{k=-\infty}^{\infty} \frac{1}{2T}\, \mathbb{E}\{\, [x_{a,k}^{\mathrm{T}} \;\; u_{e,k}^{\mathrm{T}}] \begin{bmatrix} Q_e & 0 \\ 0 & R_e \end{bmatrix} \begin{bmatrix} x_{a,k} \\ u_{e,k} \end{bmatrix} \} \;, \tag{2.2.14}$$

where

$$dF_{ae} = dF_a - dG_a\, Q_{22}^{-1}\, Q_{21} \tag{2.2.15}$$

$$u_{e,k} = u_k + Q_{22}^{-1}\, Q_{21}\, x_{a,k} \tag{2.2.16}$$

$$Q_e = Q_{11} - Q_{12}\, Q_{22}^{-1}\, Q_{21} \tag{2.2.17}$$

$$R_e = Q_{22}\ . \tag{2.2.18}$$

The optimal constant feedback gain for the transformed system is given by[14]

$$dK_{lqe} = (R_e + dG_a^T\, S\, dG_a)^{-1} dG_a^T\, S\, dF_{ae}\ , \tag{2.2.19}$$

where S is obtained by solving the steady-state discrete algebraic Riccati equation

$$0 = dF_{ae}^T (S^{-1} + dG_a\, R^{-1}\, dG_a)^{-1} dF_{ae} - S + Q_e\ , \tag{2.2.20}$$

and

$$u_{e,k} = -dK_{lqe}\, x_{a,k}\ . \tag{2.2.21}$$

Thus, while the discrete-time LQG regulator is implemented as a discrete-time Kalman filter and an optimal LQ state-feedback, a simple expression for the feedback signal for the original system is obtained from Eq.(2.2.16) as

$$u_k = -dK_{lq}\, x_{a,k}\ , \tag{2.2.22}$$

where

$$dK_{lq} = (R_e + dG_a^T\, S\, dG_a)^{-1} dG_a^T\, S\, dF_{ae} + Q_{22}^{-1} Q_{21}\ . \tag{2.2.23}$$

The discrete-time Kalman filter is used to obtain the estimated state for

feedback. In order to account for the computation delay in a closed-loop system, the Kalman filter is arranged to be a one-step-ahead predictor. Assuming that $\widehat{x}_{a,k+1/k}$ is the estimate of x_a at the time instant $k+1$ given the information up to k, the optimal full-state estimator has the form

$$\widehat{x}_{a,k+1/k} = dF_a\, \widehat{x}_{a,k/k-1} + dG_a\, u_k + dF_a\, dK_f\, [y_k - \widehat{y}_k]\,, \tag{2.2.24}$$

and the optimal constant Kalman filter gain is given by

$$dK_f = P\, C_a^T\, R_f^{-1}\,, \tag{2.2.25}$$

where R_f is measurement error covariance matrix and

$$P = M - MC_a^{\mathrm{T}}(C_aMC_a^{\mathrm{T}} + R_f)^{-1}C_aM\,, \tag{2.2.26}$$

is the error covariance matrix after the measurement update. The steady state covariance of the state estimate before measurement, M, can be obtained by solving the algebraic Riccati equation

$$0 = dF_a\,(M^{-\mathrm{T}} + C_a^{\mathrm{T}}\, R_f^{-1} Ca)^{-1} dF_a^{\mathrm{T}} - M^{\mathrm{T}} + Q_d\,. \tag{2.2.27}$$

The computational algorithm is given as follows:

$$\widehat{x}_{a,k/k-1} = dF_a\ \widehat{x}_{a,k-1/k-1} + dG_a\, u_{k-1} \tag{2.2.28}$$

$$u_k = -K_{lq}\, \widehat{x}_{a,k/k-1} \tag{2.2.29}$$

$$\widehat{y}_k = C_a\, \widehat{x}_{a,k/k-1} \tag{2.2.30}$$

$$\widehat{x}_{a,k/k} = \widehat{x}_{a,k/k-1} + dK_f\,(y_k - \widehat{y}_k)\,, \tag{2.2.31}$$

and the structure of the LQG compensator is shown in Figure 2.1 .

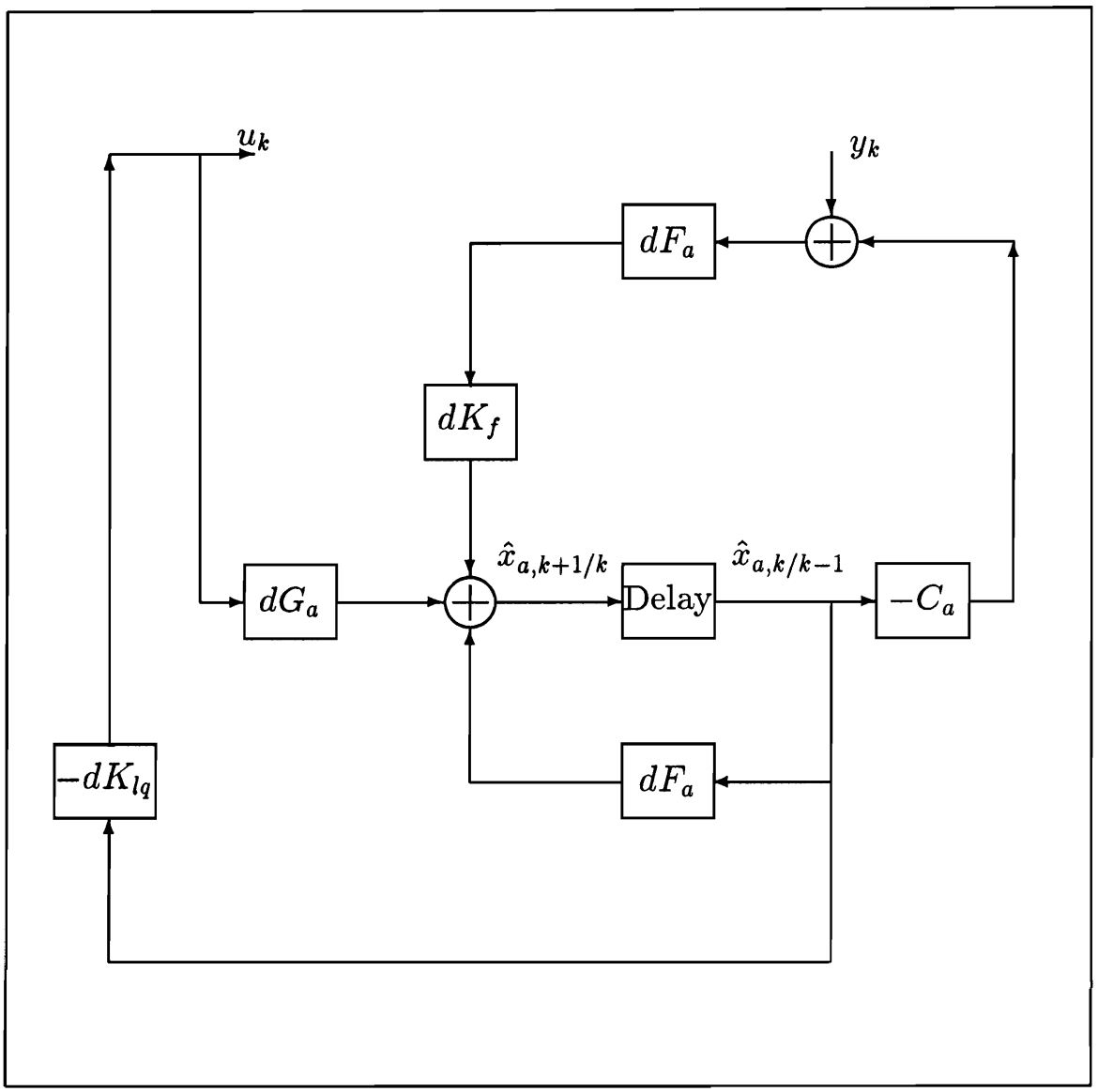


Figure 2.1: Structure of LQG Compensator

2.3 Simply-Supported Plate Experiment

Disturbance rejection for a simply-supported plate is considered in this section. The plate is excited with a disturbance by a point force shaker. A second shaker at different location is used to control the vibration of the plate. A transputer system is used for data acquisition and control. For a more detailed description of the experimental setup, see [15].

Both finite-element and modal analyses were performed[15] to obtain the state-space model of the simply-supported plate. The first nine natural frequencies and damping ratios are given in Table 2.1. The system matrix has the form

$$F = \begin{bmatrix} 0 & I \\ -\Omega^2 & -2\mathcal{Z}\Omega \end{bmatrix},$$

where

$$\Omega = diag\{\omega_1, \omega_2, \cdots, \omega_n\}$$

$$\mathcal{Z} = diag\{\zeta_1, \zeta_2, \cdots, \zeta_n\},$$

and the disturbance and control influence matrices are given by

$$G = \begin{bmatrix} 0 \\ g \end{bmatrix}, \qquad L = \begin{bmatrix} 0 \\ l \end{bmatrix},$$

where g and l are the eigenvectors corresponding to the control and disturbance inputs.

Twelve accelerometers are used as sensors for modal accelerations. According to the locations of these accelerometers, a 12×9 eigenvector matrix was obtained experimentally as

$$\Phi = \begin{bmatrix}
0.2068 & -0.5350 & 0.4716 & -0.6245 & -0.4200 & -0.3708 & -0.5257 & -0.6133 & -0.2620 \\
0.4290 & -0.7178 & 0.1104 & -0.1516 & -0.6814 & 0.4496 & -0.0798 & 0.6420 & -0.4180 \\
0.3072 & -0.4771 & -0.5241 & 0.6138 & -0.4607 & -0.3951 & 0.4640 & -0.6078 & -0.2086 \\
0.4001 & -0.2333 & 0.8233 & -0.1619 & 0.3468 & -0.6213 & 0.5346 & -0.2504 & 0.4473 \\
0.7033 & -0.3062 & 0.1353 & -0.0554 & 0.5246 & 0.6389 & 0.1319 & 0.3292 & 0.6070 \\
0.4829 & -0.1404 & -0.9120 & 0.4227 & 0.3599 & -0.6038 & -0.6515 & -0.2957 & 0.3680 \\
0.4468 & 0.2215 & 0.7428 & 0.6210 & 0.3438 & -0.6273 & 0.6039 & 0.3454 & -0.3710 \\
0.7033 & 0.3664 & 0.1091 & 0.2185 & 0.5573 & 0.6715 & 0.1343 & -0.4100 & -0.5521 \\
0.4480 & 0.2853 & -0.8733 & -0.4771 & 0.3848 & -0.6201 & -0.6411 & 0.3428 & -0.3629 \\
0.2803 & 0.4538 & 0.5085 & 1.0755 & -0.5249 & -0.3369 & -0.7840 & 0.6357 & 0.1768 \\
0.4365 & 0.7018 & 0.0766 & 0.2210 & -0.7149 & 0.4368 & -0.1769 & -0.8063 & 0.3551 \\
0.1884 & 0.4729 & -0.5331 & -1.0680 & -0.4109 & -0.3970 & 0.7980 & 0.5814 & 0.1820
\end{bmatrix} ,$$

where Φ_{ij} is the contribution of acceleration of jth mode to the ith accelerometer. In order to obtain modal accelerations from the output of these twelve accelerometers, the least squares technique is used since more equations than unknowns are available to get an exact solution and the result is written as

$$y_m = (\Phi^{\mathrm{T}}\, \Phi)^{-1} \Phi^{\mathrm{T}}\, y_s \ ,$$

where $y_m \in R^9$, is the modal output and $y_s \in R^{12}$, is the sensor output.

Since the disturbance shaker and the control shaker are each chosen to be placed at the same location as one of the sensors, g and l can be obtained by extracting rows from the modal matrix Φ. As a result, g^{T} is equal to the 8th row of Φ and l^{T} is equal to 5th row of Φ. In our example, only the first two modes of vibration are used to model the plate and g and l are truncated to 2×1 matrices. Also, matrix D is actually g and E is equal to l.

The disturbance shaker is driven either by a 60 Hz harmonic signal or by a narrow-band signal with 60 Hz center frequency. The 60 Hz harmonic signal is produced by a signal generator and the narrowband signal is obtained by feeding a ZOH computer generated random signal through a narrowband shaping filter. The shaping filter is modeled by 1% damping and the matrices in Eq. (2.1.3) become

Table 2.1: Identified Natural Frequencies and Damping Ratios for Simply-Supported Plate

Mode	Frequency(Hz)	Damping Ratio
1	49.45	0.0077228
2	108.96	0.0171460
3	130.25	0.0083185
4	188.53	0.0027311
5	203.25	0.0027250
6	265.62	0.0023876
7	285.78	0.0012244
8	326.08	0.0013216
9	338.30	0.0022209

$$F_w = \begin{bmatrix} 0 & 1 \\ -\omega_w^2 & -2\zeta_w\omega_w \end{bmatrix}, \quad G_w = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C_w = \begin{bmatrix} 0 & 1 \end{bmatrix},$$

where $\omega_w = 2\pi \cdot 60$ rad/sec and $\zeta_w = 0.01$ and the frequency response of the shaping filter model and the autospectrum of the narrowband signal actually generated are shown in Figure 2.2 and Figure 2.3 respectively.

A unity gain Sallen-Key low-pass filter with corner frequency at 120 Hz is used as the smoothing filter. The matrices in Eq. (2.1.5) then become

$$F_{lp} = \begin{bmatrix} 0 & 1 \\ -\omega_{lp}^2 & -2\zeta_{lp}\omega_{lp} \end{bmatrix}, \quad G_{lp} = \begin{bmatrix} 0 \\ \omega_{lp}^2 \end{bmatrix}, \quad C_{lp} = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

where $\omega_{lp} = 2\pi \cdot 120$ rad/sec and $\zeta_{lp} = 0.707$, and the frequency response of the smoothing filter model is plotted in Figure 2.4.

Since only one control shaker is used and there is only one disturbance source, the problem becomes single-input-multiple-output(SIMO) and the control efforts weighting matrix R becomes a scalar ρ, which determines how much control effort can be applied.

In order to suppress the vibration of the first mode as much as possible, we choose to penalize the total energy in mode one[8]. The state weighting matrix becomes

$$Q = \begin{bmatrix} \omega_1^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

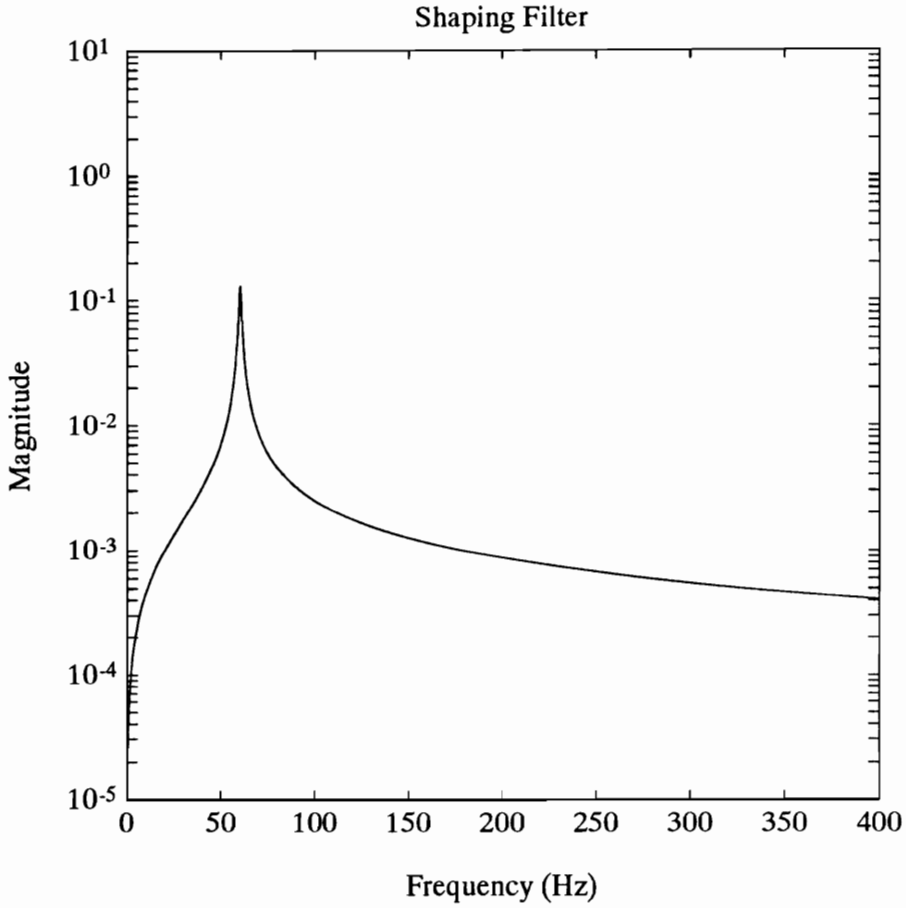


Figure 2.2: Frequency Response of Shaping Filter

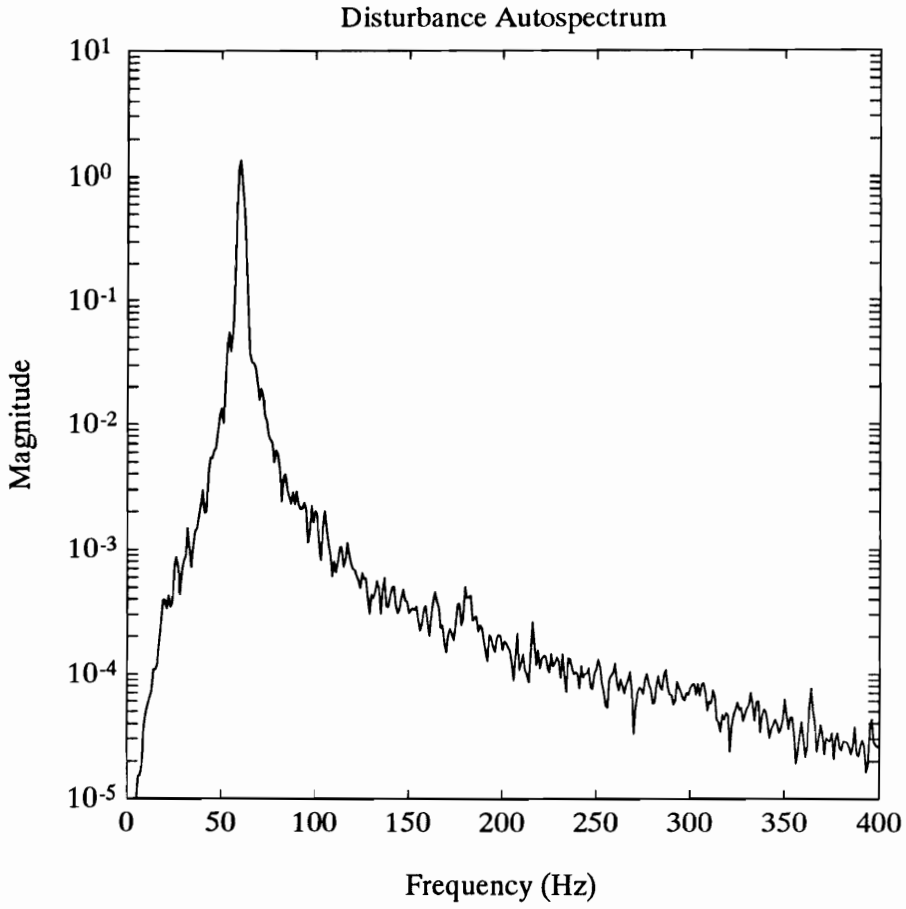


Figure 2.3: Narrowband Disturbance Autospectrum

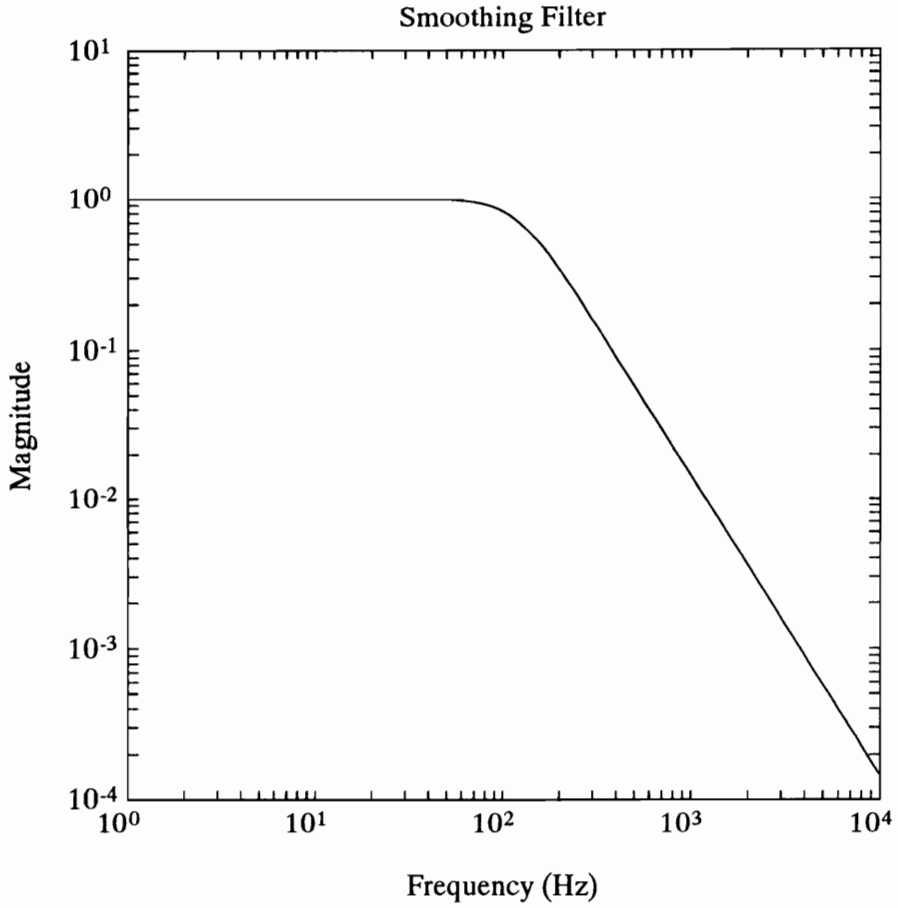


Figure 2.4: Frequency Response of Smoothing Filter

where ω_1 is the first mode natural frequency.

The sampling frequency for the transputer data acquisition and control system can go as high as 4360 Hz for this two mode controller. Simulated results of 60 Hz disturbance rejection of the controller are plotted in Figures 2.5–2.6, which show the effect of choosing different ρ values. Also, plotted in Figures 2.7–2.8 are the results of narrowband disturbance rejection with different ρ values. With a smaller value of ρ, more control effort is used and more rejection of disturbance is obtained.

When the controller is applied to the actual plate, the Kalman filter is designed based upon the process noise variance

$$Q_c = 500 ,$$

and the measurement noise error covariance matrix

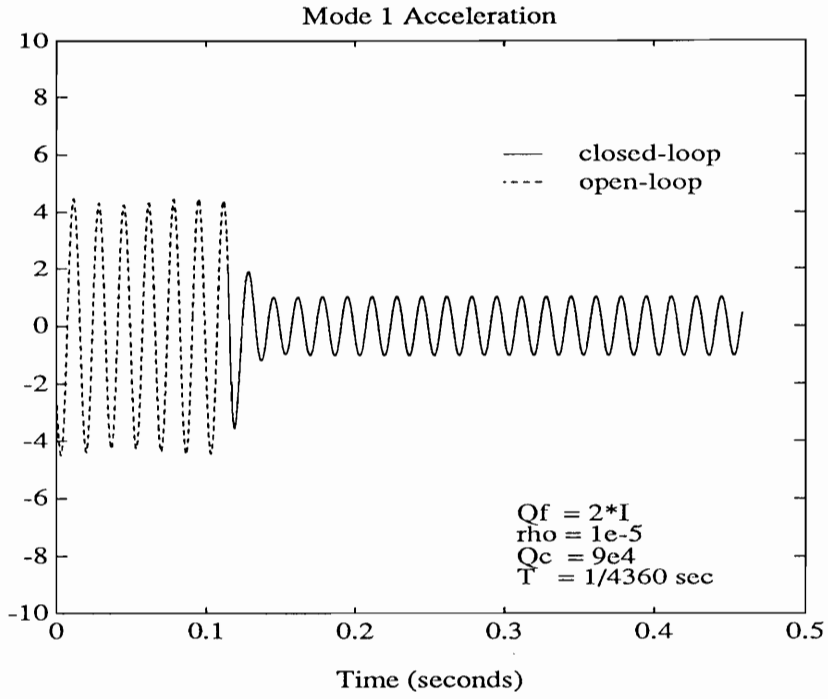
$$R_f = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}.$$

Figures 2.9–2.10 are the experimental results of 60 Hz disturbance rejection with different ρ values. Similarly, Figures 2.11–2.12 are results of narrowband rejection.

A comparison of the simulated and experimental results demonstrates the effectiveness of this controller design method as long as the structure is accurately modeled.

2.4 Performance and Stability

An LQG regulator is optimal in the sense that it minimizes a quadratic cost function. It is sometimes helpful to know what the exact value of the cost is for a certain design. In this section, we show a way to calculate the cost and explain how model uncertainty affects the stability in terms of the cost.



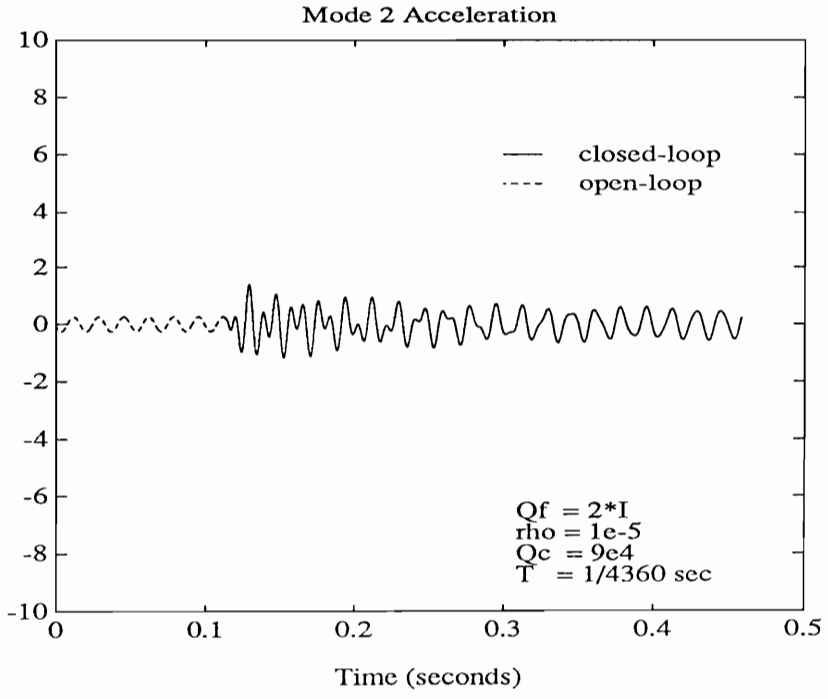
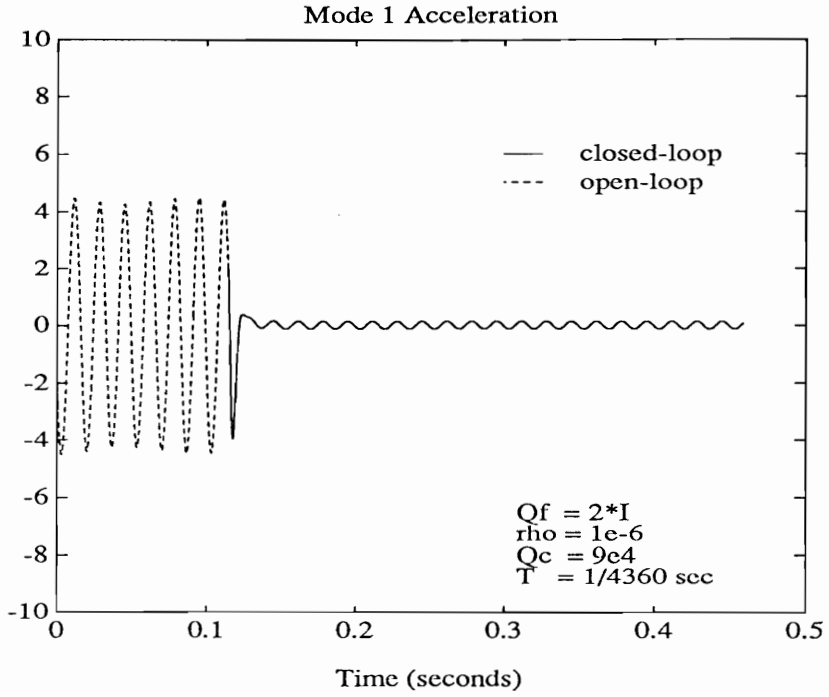


Figure 2.5: LQG Disturbance Rejection (Simulated, 60 Hz, $\rho = 10^{-5}$)



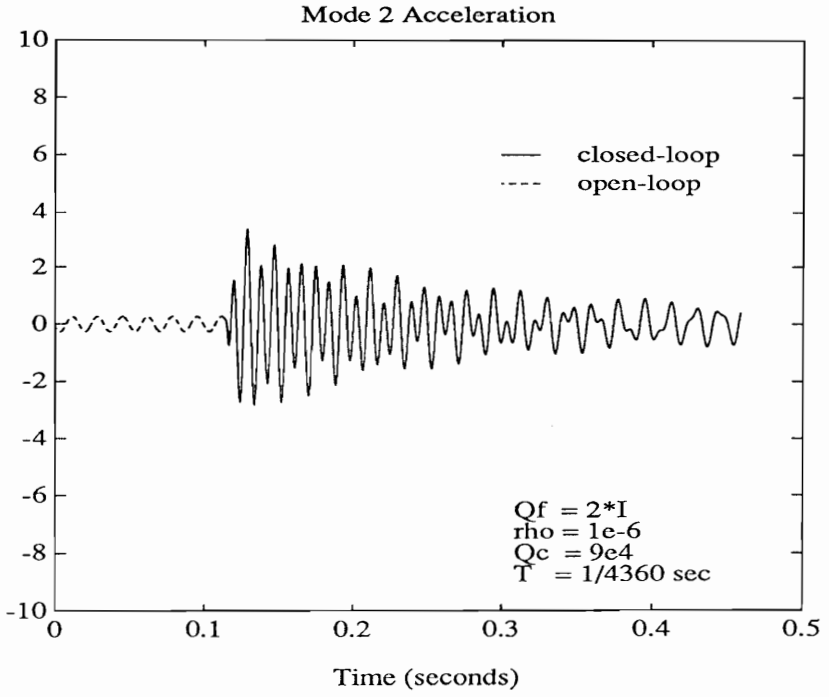
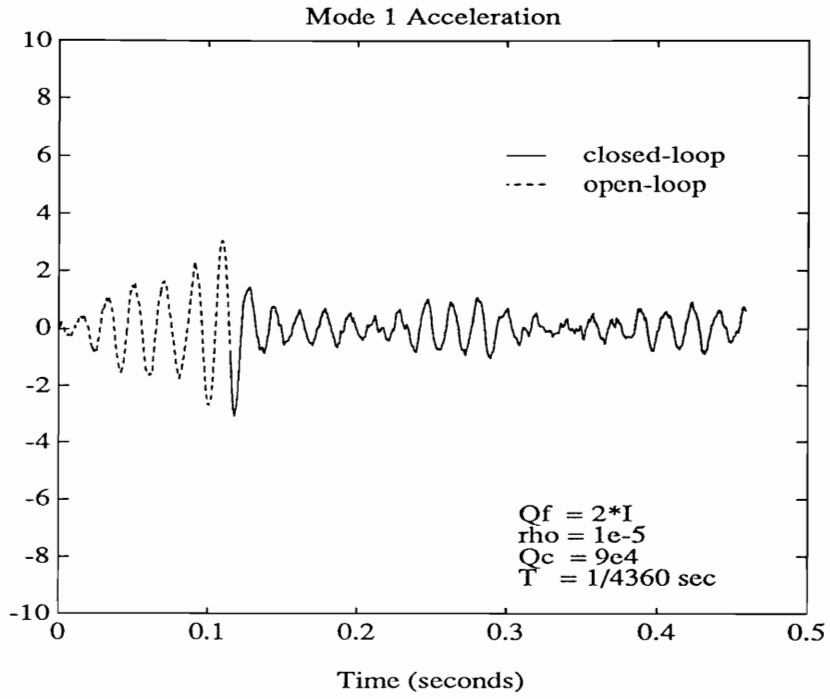


Figure 2.6: LQG Disturbance Rejection (Simulated, 60 Hz, $\rho = 10^{-6}$)



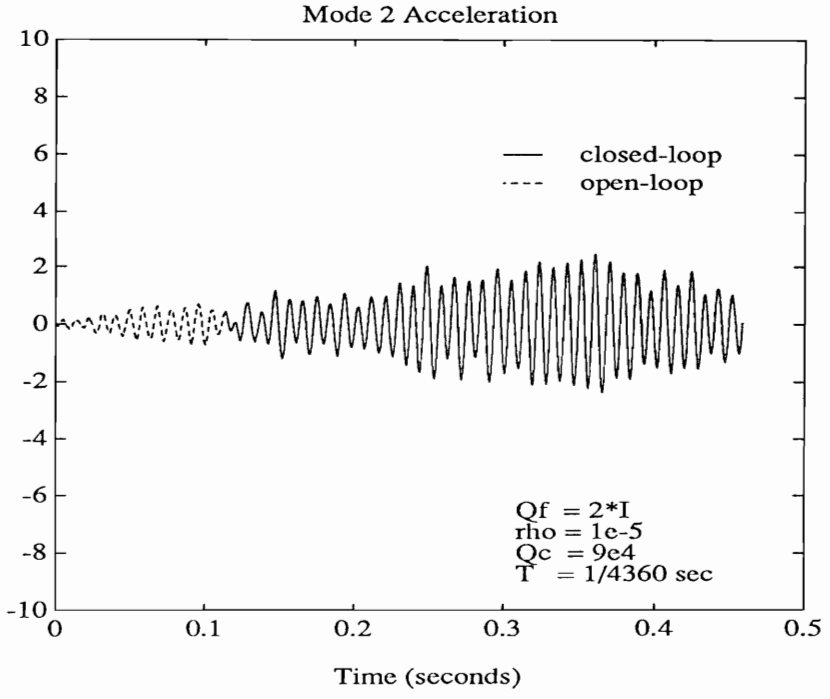
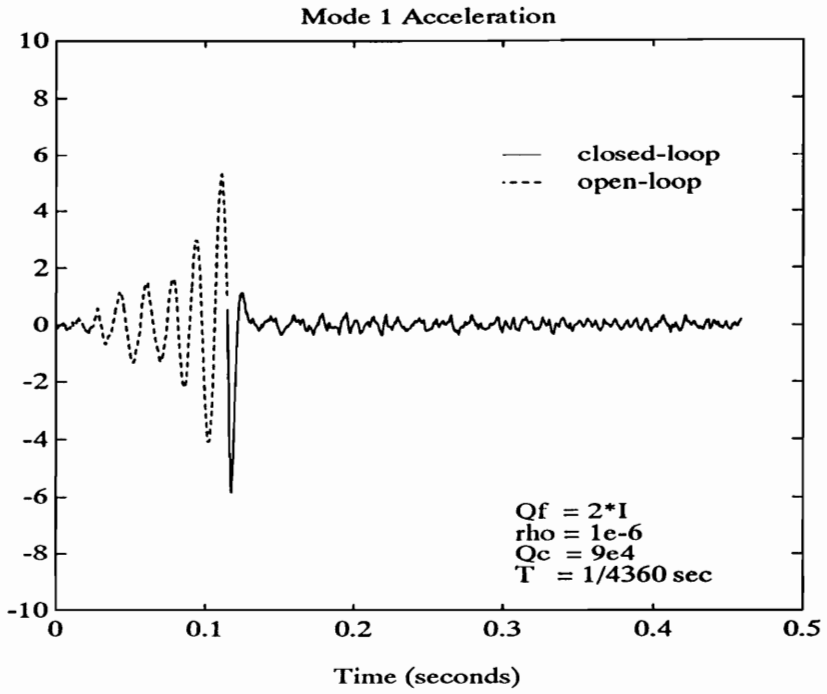


Figure 2.7: LQG Disturbance Rejection (Simulated, Narrowband, $\rho = 10^{-5}$)



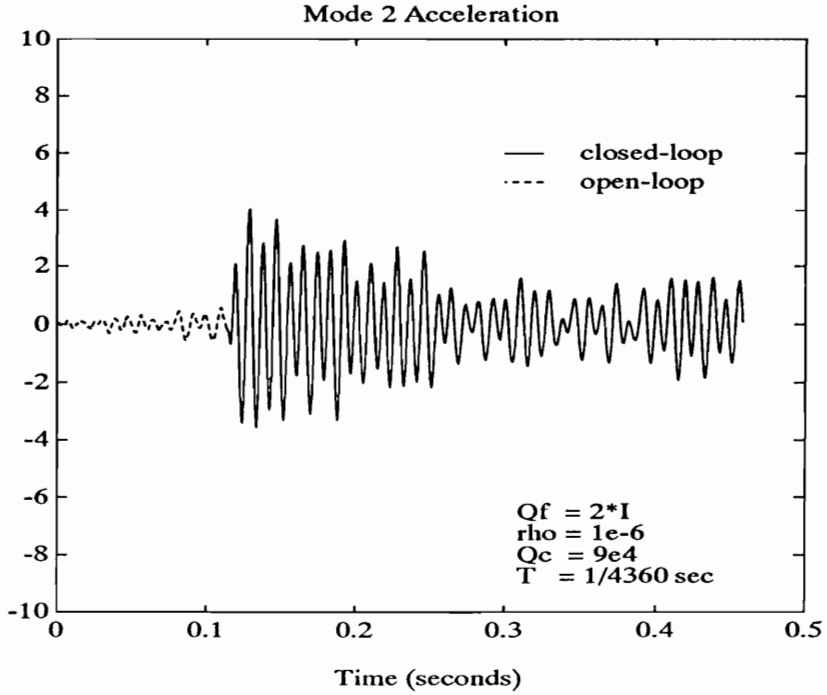
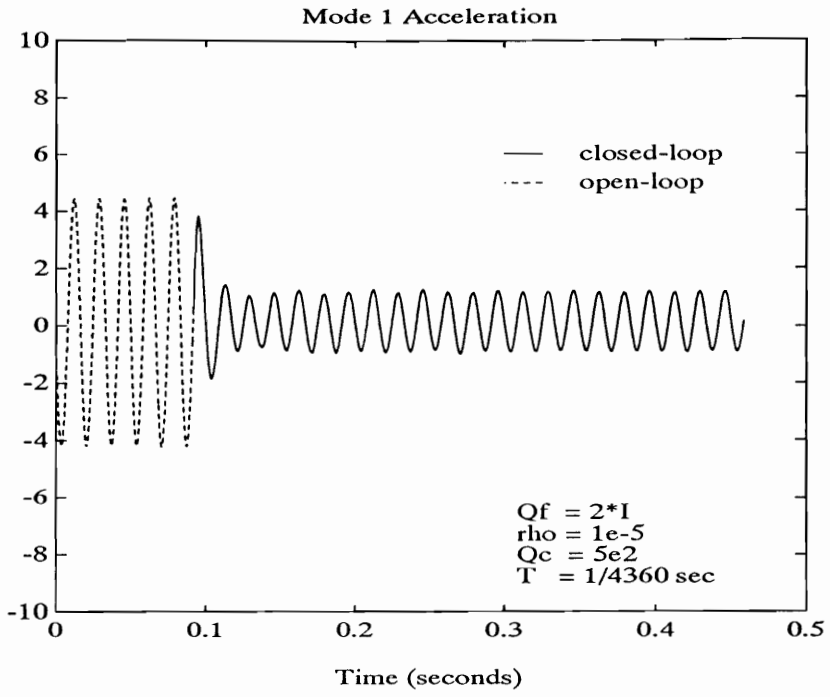


Figure 2.8: LQG Disturbance Rejection (Simulated, Narrowband, $\rho = 10^{-6}$)



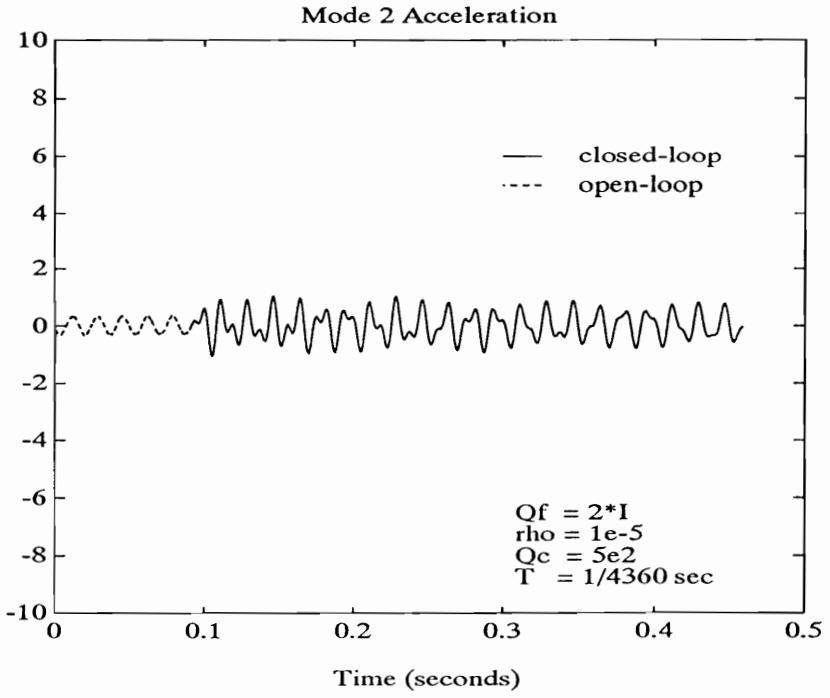
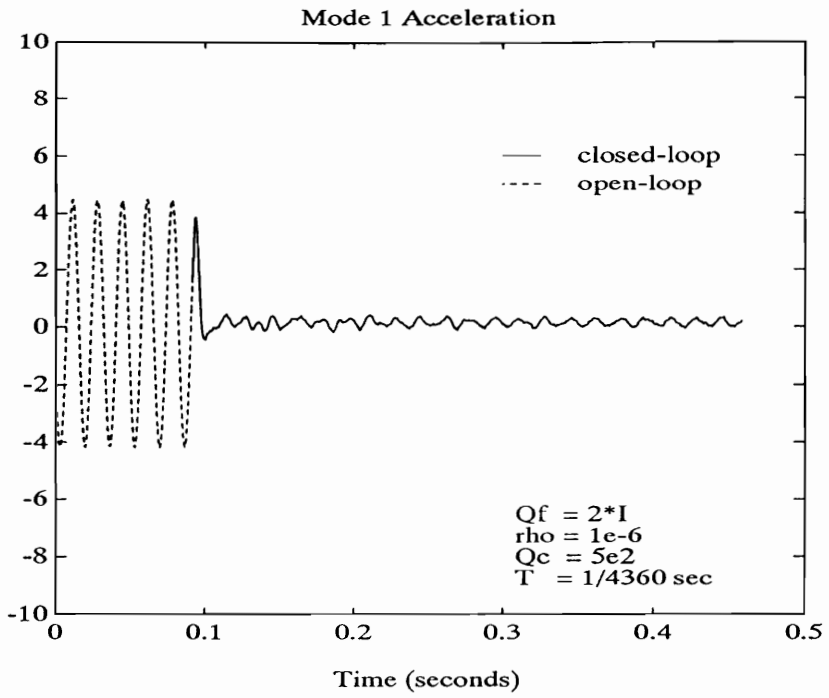


Figure 2.9: LQG Disturbance Rejection (Experimental, 60 Hz, $\rho = 10^{-5}$)



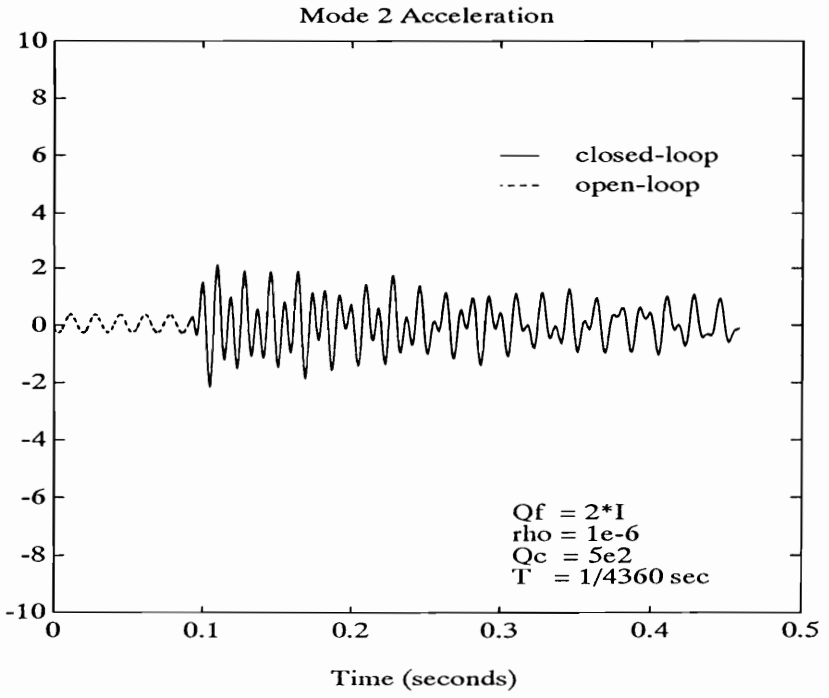
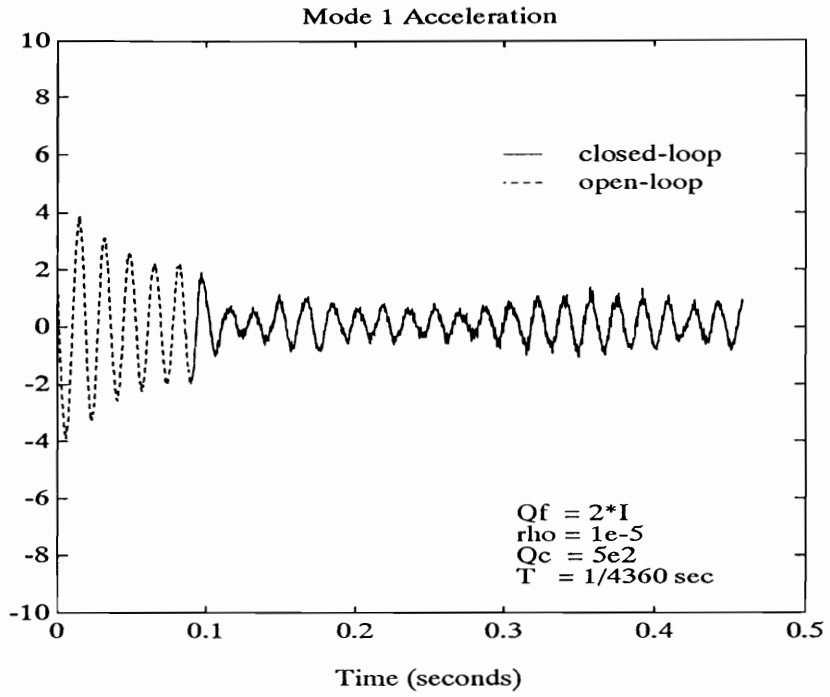


Figure 2.10: LQG Disturbance Rejection (Experimental, 60 Hz, $\rho = 10^{-6}$)



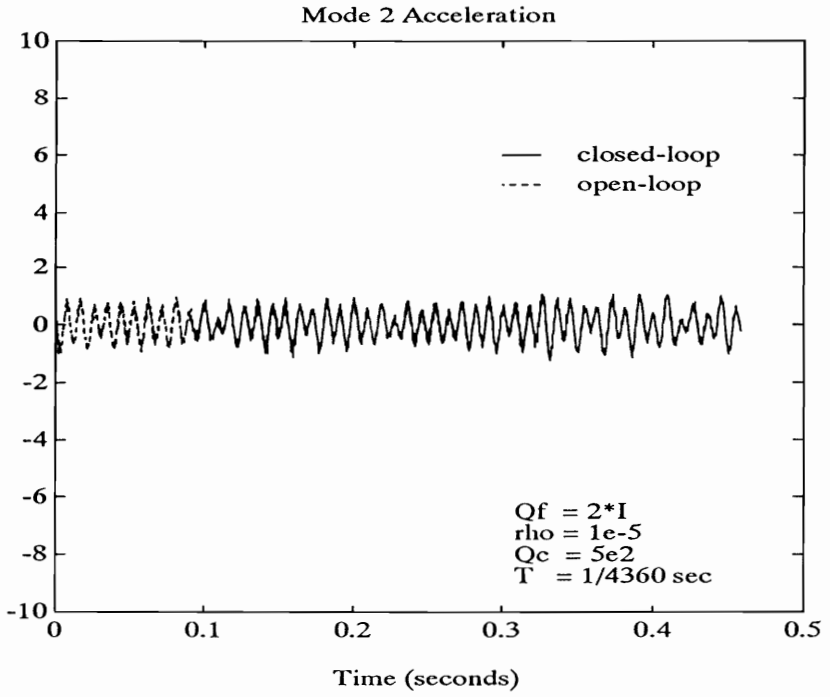
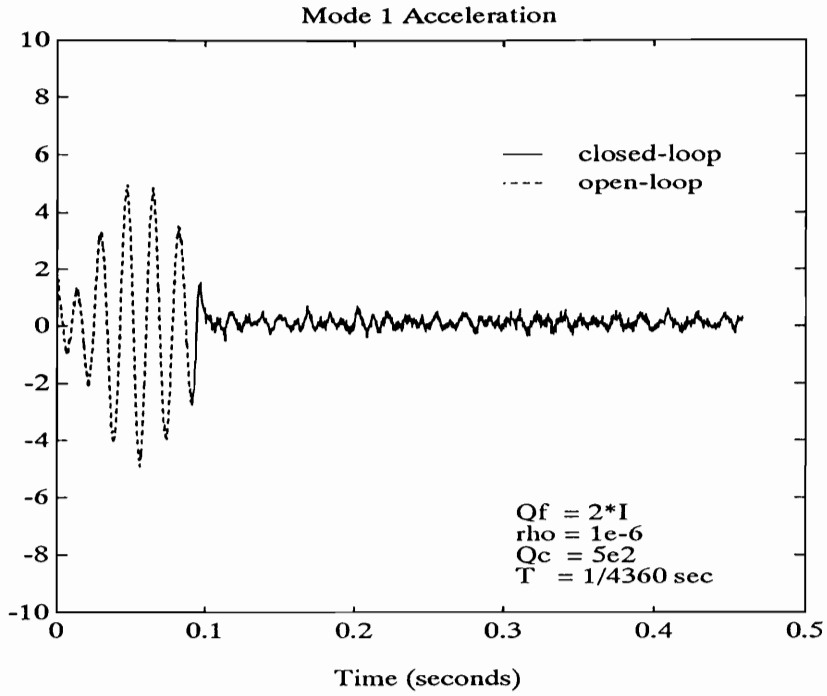


Figure 2.11: LQG Disturbance Rejection (Experimental, Narrowband, $\rho = 10^{-5}$)



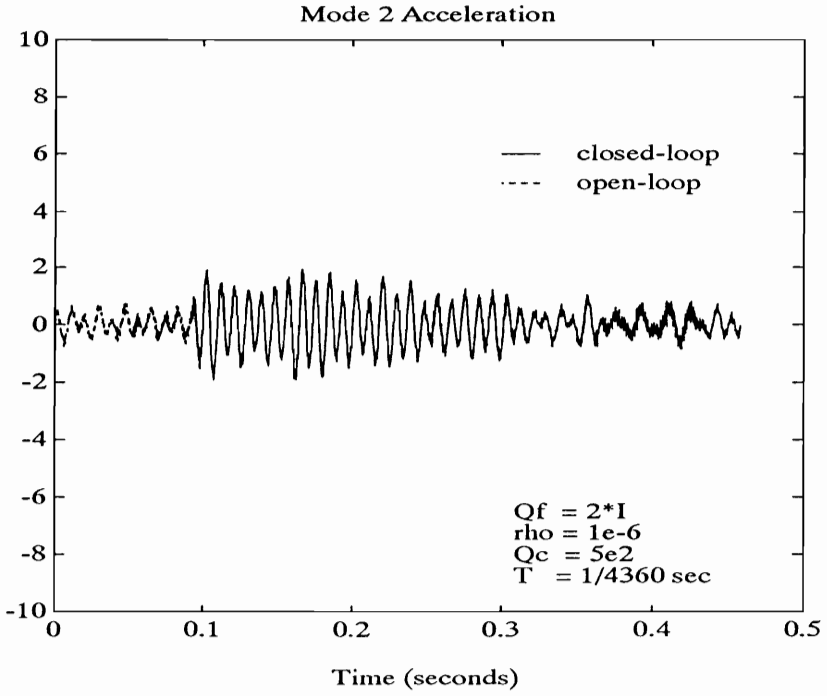


Figure 2.12: LQG Disturbance Rejection (Experimental, Narrowband, $\rho = 10^{-6}$)

As described in Section 2.1, given the state-space model

$$\dot{x}_a = F_a\, x_a + G_a\, u\ + L_a\, v \tag{2.4.1}$$

$$y\ = C_a\, x_a + \ \theta\ , \tag{2.4.2}$$

the Kalman-Bucy filter can be derived as

$$\dot{\hat{x}}_a = F_a\, \hat{x}_a + G_a\, u + K_f(y - \hat{y})\ . \tag{2.4.3}$$

Noting that $\hat{y} = C_a\ \hat{x}_a$, and assuming that the actual dynamics of the augmented system are described by

$$\dot{x}_a = \ \bar{F}_a\, x_a + \bar{G}_a\, u\ + \bar{L}_a\, v \tag{2.4.4}$$

$$y\ =\ \bar{C}_a\, x_a\ +\ \theta\ , \tag{2.4.5}$$

the estimator equation is written as

$$\dot{\hat{x}}_a = F_a\, \hat{x}_a + G_a\, u + K_f(\bar{C}_a\, x_a + \theta - C_a\, \hat{x}_a\ )\ . \tag{2.4.6}$$

Combining Eq. (2.4.4) and Eq. (2.4.5) with the estimated state feedback results in the state equation of the closed-loop system

$$\begin{bmatrix} \dot{x}_a \\ \dot{\hat{x}}_a \end{bmatrix} = \begin{bmatrix} \bar{F}_a & -\bar{G}_a K_{lq} \\ K_f \bar{C}_a & F_a - K_f C_a - G_a K_{lq} \end{bmatrix} \begin{bmatrix} x_a \\ \hat{x}_a \end{bmatrix} \ + \begin{bmatrix} \bar{L}_a & 0 \\ 0 & K_f \end{bmatrix} \begin{bmatrix} v \\ \theta \end{bmatrix}. \tag{2.4.7}$$

Since $v \rightarrow N(0, Q_c)$, Eq. (2.4.7) is equivalent to

$$\begin{bmatrix} \dot{x}_a \\ \dot{\hat{x}}_a \end{bmatrix} = \begin{bmatrix} \bar{F}_a & -\bar{G}_a K_{lq} \\ K_f \bar{C}_a & F_a - K_f C_a - G_a K_{lq} \end{bmatrix} \begin{bmatrix} x_a \\ \hat{x}_a \end{bmatrix} \ + \begin{bmatrix} \bar{L}_a Q_c^{1/2} & 0 \\ 0 & K_f \end{bmatrix} \begin{bmatrix} v_1 \\ \theta \end{bmatrix}, \tag{2.4.8}$$

where $v_1 \rightarrow N(0,1)$. Assuming that the measurement is contaminated by unit intensity white noise, we define the noise vector

$$w = \begin{bmatrix} v_1 \\ \theta \end{bmatrix},$$

where $\mathbb{E}[ww^{\mathrm{T}}] = I$.

The cost function to be minimized in LQG control is actually a 2-norm defined in time domain, which is explained in the following definition.

Definition . The 2-norm of a transfer function matrix $H \in C^{m \times n}$ is given by

$$\begin{aligned} \|H\|_2^2 \quad &\triangleq \langle H, H \rangle = \frac{1}{2\pi} \int_{-\infty}^{\infty} Tr[H(j\omega)^* H(j\omega)]\, d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \|H(j\omega)\|_F^2\, d\omega \\ &= \langle h, h \rangle = \int_{-\infty}^{\infty} Tr[h^{\mathrm{T}}(t)\, h(t)]\, dt \\ &= \int_{-\infty}^{\infty} \|h(t)\|_F^2\, dt \end{aligned}$$

where $h(t) \in R^{m \times n}$ is its impulse response matrix.

Now, if we define a controlled-variable vector z by

$$z = \begin{bmatrix} Q^{1/2} \cdot x_a \\ R^{1/2} \cdot u \end{bmatrix},$$

and let the closed-loop transfer function matrix from w to z equal H, then it can be shown[8] that the cost function J is actually $\|H\|_2^2$.

The next step is to calculate the value of the frequency domain 2-norm by using the time domain formulation. If the quadruple of matrices $\{A, B, C, D\}$

represents a realization of the transfer function matrix $H(s)$, then

$$H(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array}\right] \triangleq C(sI - A)^{-1}B \ ,$$

where $D = 0$ in order for $\| H \|_2 < \infty$. Let L_c and L_o denote the controllability Grammian and observability Grammian respectively:

$$L_c = \int_0^\infty e^{At} B \ B^{\mathrm{T}} \ e^{A^{\mathrm{T}} t} \ dt$$

$$L_o = \int_0^\infty e^{A^{\mathrm{T}} t} C^{\mathrm{T}} C \ e^{At} \ dt$$

then the transfer function 2-norm can be shown to be related to L_c and L_o as follows:

$$\begin{aligned} \| H \|_2^2 &= \int_0^\infty Tr[h(t) \ h^{\mathrm{T}}(t)]dt \\ &= Tr[\int_0^\infty (Ce^{At}B)(B^{\mathrm{T}}e^{A^{\mathrm{T}}t})dt \ ] = Tr(CL_cC^{\mathrm{T}}) \\ &= \int_0^\infty Tr[h^{\mathrm{T}}(t) \ h(t)]dt \\ &= Tr[\int_0^\infty (B^{\mathrm{T}}e^{A^{\mathrm{T}}t}C^{\mathrm{T}})(C \ e^{At}B)dt \ ] = Tr(B^{\mathrm{T}}L_oB) \ . \end{aligned}$$

The controllability gramian L_c can be shown[16] to satisfy the matrix Lyapunov equation

$$AL_c \ + \ L_cA^{\mathrm{T}} \ + \ BB^{\mathrm{T}} \ = \ 0 \ ,$$

and similarly the observability gramian L_o satisfies

$$A^{\mathrm{T}} L_o + L_o A + C^{\mathrm{T}} C = 0 .$$

Since the matrix Lyapunov equations are linear and can be solved by computer with reliable and efficient algorithms[17], the 2-norm of the transfer function and the cost of the LQG controlled system can be easily calculated.

If the model of the plant is not correctly identified, then the LQG controller will not be optimal and the cost is expected to go up. The worst case is that the system becomes unstable and theoretically the cost of an unstable system is infinity. Thus, the sensitivity analysis of the controller can be done by calculating the cost against the variation of the model parameters. As an example, the two mode controller of the simply-supported plate example given in Section 2.3 is considered. The cost is found to be most sensitive to the natural frequency of the 2nd mode and is plotted against deviation of the 2nd mode natural frequency in Figure 2.13 . This figure shows that a $\pm 6\%$ variation will make the system unstable.

2.5 Design Tradeoff

It has been shown[18] that addition of a fictitious noise term to the process noise can cause the LQG controlled system to asymptotically achieve the robust characteristic of full-state feedback. This is called the loop transfer recovery technique, also known as LQG/LTR .

Intuitively, by assuming a higher process and/or measurement noise intensity, equivalent to adding extra fictitious noise, the robustness of the controller might be enhanced. While there is no guarantee that robustness for an LQG controller will be enhanced by arbitrarily assuming higher intensities for process and/or measurement noise, it sometimes does so[8].

Thus, assuming higher noise intensity sometimes stabilizes a system with a controller designed using an incorrect model. However, performance must be

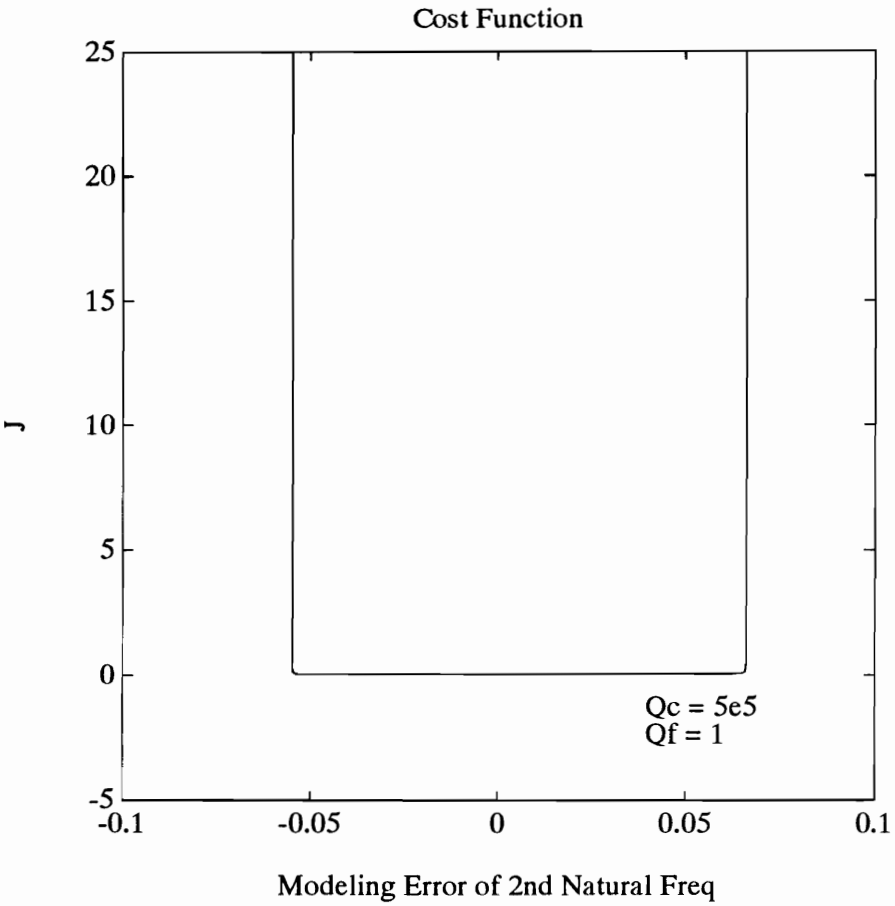


Figure 2.13: Sensitivity of LQG Control

sacrificed in order to maintain stability as can be seen by an increased cost. Taking the two mode controller of the simply-supported plate as an example, we show in Figure 2.14 how the cost and the robustness change for different measurement noise intensities. For a higher measurement noise intensity($Q_f = 2$), the system stays stable for more than $+10\%$ deviation of the parameter and the robustness is enhanced a little in the negative direction of deviation as well, however, the cost is seen to be higher.

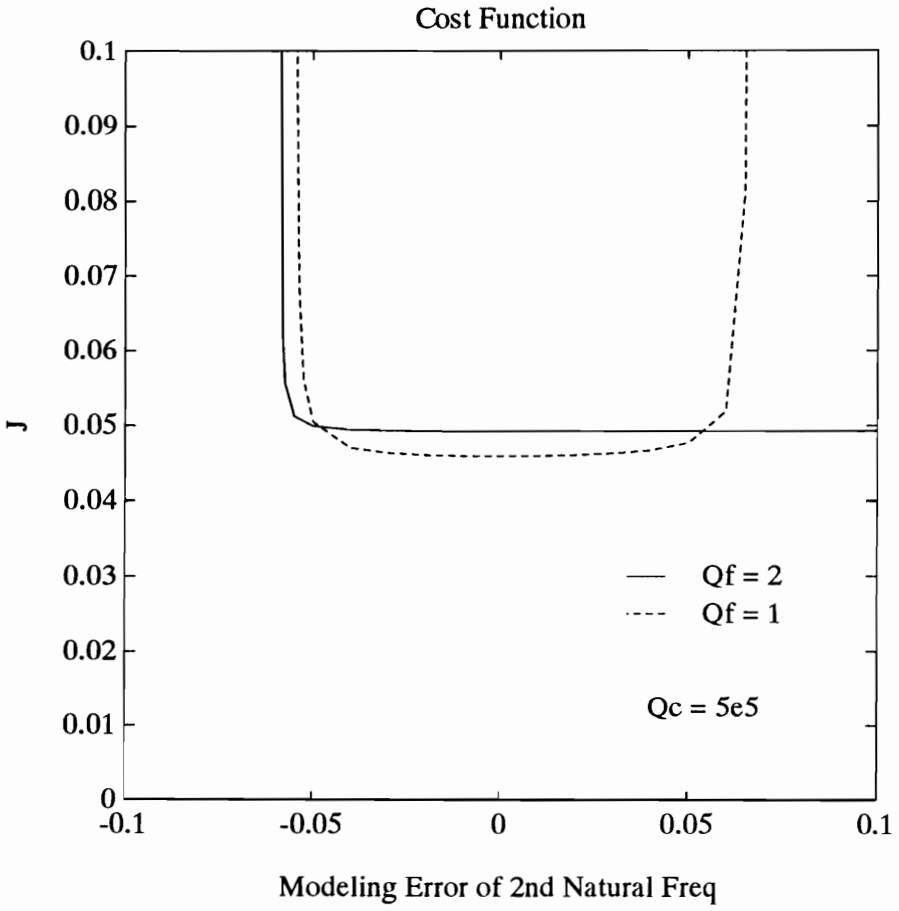


Figure 2.14: Performance of LQG Control

3. Recursive Parameter Estimation

Usually we have to start with a mathematical model if we are interested in studying, affecting, or controlling a dynamic system. Building mathematical models of dynamic systems based upon observed data from the systems is the subject of system identification. Quite often a set of candidate models is selected and the best model in the set is determined according to a rule by which candidate models can be assessed using the observed data. If the set of models is parametrized in terms of a parameter vector θ, the search for the best model in the set becomes a problem of determining or estimating the vector θ. In many cases, the model is needed to support on-line decisions and should be based upon measurement data up to the current time. The updating of the model at each time instant some new data becomes available is accomplished by a recursive algorithm. A recursive algorithm can be obtained by modifying an off-line counterpart or derived based upon other approaches. The requirement to satisfy is that one iteration in the numerical search is performed when a new observation is available. Recursive parameter estimation plays an important role in adaptive control since, pragmatically, adaptive control can be looked upon as a direct combination of a control methodology with some form of recursive system identification. In this chapter, several system and signal models useful in control systems design are introduced first and followed by a review of a few recursive parameter estimation methods from certain important categories of approaches. For more detailed discussion of these methods, see [19], [20], [21] and [22].

3.1 System and Signal Models

Consider a dynamic system with input signal $\{u_k\}$ and output signal $\{y_k\}$ and suppose that these signals are sampled at discrete time $k = 1, 2, 3, \cdots$. A linear difference equation can be written to describe the dynamic relationship between the input and output signals as

$$y_k + a_1 y_{k-1} + \cdots + a_n y_{k-n} = b_1 u_{k-1} + \cdots + b_m u_{k-m} + v_k \,, \tag{3.1.1}$$

where v_k is some disturbance of unspecified character.

If we let q^{-1} be the backward shift operator such that

$$q^{-1} y_k = y_{k-1} \tag{3.1.2}$$

then Eq. (3.1.1) can be written as

$$A(q^{-1}) y_k = B(q^{-1}) u_k + v_k \,, \tag{3.1.3}$$

where

$$A(q^{-1}) = 1 + a_1 q^{-1} + \cdots + a_n q^{-n} \tag{3.1.4}$$

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2} + \cdots + b_m q^{-m} \,. \tag{3.1.5}$$

With the *Auto*Regressive part $A(q^{-1})y_k$ and the control part $B(q^{-1})u_k$ in which the control signal is known as e*X*ogeneous variable, this model is recognized as the ARX model.

If we define the parameter vector as

$$\theta = [\ a_1 \cdots a_n \ \ b_1 \cdots b_m\ ]^{\mathrm{T}} \,, \tag{3.1.6}$$

and introduce the vector

$$\varphi_k = [\, -y_{k-1} \ \cdots \ -y_{k-n} \ \ u_{k-1} \ \cdots \ u_{k-m}]^{\mathrm{T}} \,, \tag{3.1.7}$$

then Eq. (3.1.1) can be written as

$$y_k = \theta^{\mathrm{T}} \varphi_k + v_k \,. \tag{3.1.8}$$

Thus, the observed variable y_k is expressed as an unknown linear

combination of the components of the observed vector φ_k plus noise and the model is called a linear regression in statistics where the components of φ_k are called regressors. $\theta^T\varphi_k$ can be thought of as the prediction of what y_k is going to be and recognized as

$$\hat{y}_k(\theta) \triangleq \theta^T\varphi_k \,. \tag{3.1.9}$$

Flexibility can be added to the ARX model by also modeling the disturbance term v_k. One approach is to describe the disturbance as a *M*oving *A*verage of a white noise sequence $\{e_k\}$ such that

$$v_k = C(q^{-1})e_k \tag{3.1.10}$$

$$C(q^{-1}) = 1 + c_1q^{-1} + \quad \cdots \quad + c_rq^{-r} \,, \tag{3.1.11}$$

and the resulting model is written as

$$A(q^{-1})y_k = B(q^{-1})u_k + \; C(q^{-1})e_k \,. \tag{3.1.12}$$

This model is called the ARMAX model and it has become a standard tool in control systems description and design.

In this model the parameter vector is defined as

$$\theta = [\;\; a_1 \cdots a_n \;\; b_1 \cdots b_m \;\; c_1 \;\; \cdots \;\; c_r \;\;]^T \,, \tag{3.1.13}$$

and the model can be written as

$$y_k = \theta^T\varphi_k + e_k \,, \tag{3.1.14}$$

where

$$\varphi_k = [\;\; -y_{k-1} \;\cdots\; -y_{k-n} \;\;\; u_{k-1} \;\cdots\; u_{k-m} \;\; e_{k-1} \;\cdots\; e_{k-r} \;\;]^T \,. \tag{3.1.15}$$

Since φ_k is not completely known, the prediction of y_k, $\hat{y}_k(\theta)$, is not immediately available as that of the ARX model. If Eq.(3.1.12) is written as

$$y_k = \left[1 - \frac{A(q^{-1})}{C(q^{-1})} \right] y_k + \frac{B(q^{-1})}{C(q^{-1})} u_k + e_k \,, \tag{3.1.16}$$

then the prediction of y_k can be defined as

$$\hat{y}_k(\theta) = \left[1 - \frac{A(q^{-1})}{C(q^{-1})} \right] y_k + \frac{B(q^{-1})}{C(q^{-1})} u_k \,, \tag{3.1.17}$$

and the prediction error defined by $\varepsilon_k(\theta) = y_k - \hat{y}_k(\theta)$ can be written as

$$\varepsilon_k(\theta) = \frac{1}{C(q^{-1})} [\, A(q^{-1}) y_k - B(q^{-1}) u_k \,] \,. \tag{3.1.18}$$

Now, if we define

$$\varphi_k(\theta) \triangleq [\, -y_{k-1} \;\cdots\; -y_{k-n} \quad u_{k-1} \;\cdots\; u_{k-m} \; \varepsilon_{k-1}(\theta) \;\cdots\; \varepsilon_{k-r}(\theta) \,]^{\mathrm{T}}, \tag{3.1.19}$$

then the prediction error can be written as

$$\varepsilon_k(\theta) = y_k - \theta^{\mathrm{T}} \varphi_k(\theta) \,, \tag{3.1.20}$$

and we have

$$\hat{y}_k(\theta) = \theta^{\mathrm{T}} \varphi_k(\theta) \,, \tag{3.1.21}$$

which can be called a pseudo-linear regression .

In some applications, it is better to describe a linear stochastic dynamic system by a state-space model

$$x_{k+1} = F\ x_k + G\ u_k + w_k \tag{3.1.22}$$

$$y_k = H\ x_k + \epsilon_k\ , \tag{3.1.23}$$

where $\{w_k\}$ and $\{\epsilon_k\}$, which are called process noise and measurement noise respectively, are sequences of independent random vectors with certain covariance matrices,

$$\mathbb{E}[w_k\ w_k^{\mathrm{T}}] = R_1 \tag{3.1.24}$$

$$\mathbb{E}[\ \epsilon_k\ \epsilon_k^{\mathrm{T}}] = R_2 \tag{3.1.25}$$

$$\mathbb{E}[w_k\ \epsilon_k^{\mathrm{T}}] = R_{12}\ . \tag{3.1.26}$$

If the matrices, F, G, H, R_1, R_2, and R_{12} are all known, predictions and state estimate can be obtained using the Kalman filter. If these matrices are not fully known and are functions of an unknown vector θ, $F(\theta)$, $G(\theta)$, $H(\theta)$, etc, the state-space model will be very useful for parameter estimation when we have knowledge about the basic mechanism in the system but certain parameters are uncertain.

3.2 The Least Squares Method

After a set of models is selected, the best model can be determined by assessing the size of the prediction error of each model in the set. The size of prediction error can be measured using any norm of it, quadratic or non-quadratic. The best model is the one that has a least norm. Approaches that are based on this criterion are called prediction-error methods (PEM).

The least squares method is the simplest method available when the set of ARX models is selected. The norm of the prediction error to be minimized is

$$V_N(\theta) = \frac{1}{N} \sum_{k=1}^{N} \alpha_k [y_k - \theta^{\mathrm{T}} \varphi_k]^2\ , \tag{3.2.1}$$

where N is the number of observations and $\{\alpha_k\}$ is a set of positive numbers and allows us to give different weights to different observations.

The quadratic criterion function V_N can be minimized with respect to θ to give the least squares estimate

$$\hat{\theta}_N = \left[\sum_{k=1}^{N} \alpha_k \varphi_k \varphi_k^{\mathrm{T}} \right]^{-1} \sum_{k=1}^{N} \alpha_k \varphi_k y_k , \tag{3.2.2}$$

if the inverse exists.

Cast in matrix form, the result can be written as

$$\hat{\theta}_N = (\Phi_N \Lambda_N \Phi_N^{\mathrm{T}})^{-1} \Phi_N \Lambda_N Y_N^{\mathrm{T}} , \tag{3.2.3}$$

where

$$\Phi_N = [\, \varphi_1 \;\; \varphi_2 \;\; \cdots \;\; \varphi_N \,] \tag{3.2.4}$$

$$\Lambda_N = \begin{bmatrix} \alpha_1 & & & \\ & \alpha_2 & & \\ & & \ddots & \\ & & & \alpha_N \end{bmatrix} \tag{3.2.5}$$

$$Y_N = [\, y_1 \;\; y_2 \;\; \cdots \;\; y_N \,] . \tag{3.2.6}$$

3.3 Recursive Least Squares

With the least square estimate obtained based upon the observations for $k = 1, 2, \cdots, N$, a recursive algorithm is desired when the observation at $k = N + 1$ becomes available.

The non-recursive least squares estimate based upon the observations for $k = 1,2,\cdots,N,N+1$ is

$$\widehat{\theta}_{N+1} = (\Phi_{N+1}\Lambda_{N+1}\Phi_{N+1}^{\mathrm{T}})^{-1}\Phi_{N+1}\Lambda_{N+1}Y_{N+1}^{\mathrm{T}} \ , \tag{3.3.1}$$

where

$$\Phi_{N+1} = [\Phi_N \quad \varphi_{N+1}] \tag{3.3.2}$$

$$\Lambda_{N+1} = \begin{bmatrix} \Lambda_N & 0 \\ 0 & \alpha_{N+1} \end{bmatrix} \tag{3.3.3}$$

$$Y_{N+1} = [\, Y_N \quad y_{N+1}] \, . \tag{3.3.4}$$

Since

$$\Phi_{N+1}\Lambda_{N+1}\Phi_{N+1}^{\mathrm{T}} = [\Phi_N \quad \varphi_{N+1}] \begin{bmatrix} \Lambda_N & 0 \\ 0 & \alpha_{N+1} \end{bmatrix} \begin{bmatrix} \Phi_N^{\mathrm{T}} \\ \varphi_{N+1}^{\mathrm{T}} \end{bmatrix} \tag{3.3.5}$$

$$= \Phi_N\Lambda_N\Phi_N^{\mathrm{T}} + \varphi_{N+1}\,\alpha_{N+1}\,\varphi_{N+1}^{\mathrm{T}} \ , \tag{3.3.6}$$

and

$$\Phi_{N+1}\Lambda_{N+1}Y_{N+1}^{\mathrm{T}} = [\Phi_N \quad \varphi_{N+1}] \begin{bmatrix} \Lambda_N & 0 \\ 0 & \alpha_{N+1} \end{bmatrix} \begin{bmatrix} Y_N^{\mathrm{T}} \\ y_{N+1} \end{bmatrix} \tag{3.3.7}$$

$$= \Phi_N\Lambda_N Y_N^{\mathrm{T}} + \varphi_{N+1}\,\alpha_{N+1}\,y_{N+1} \ , \tag{3.3.8}$$

if we define

$$\Phi_N\Lambda_N\Phi_N^{\mathrm{T}} = P_N^{-1} , \tag{3.3.9}$$

then

$$P_{N+1}^{-1} = P_N^{-1} + \varphi_{N+1}\,\alpha_{N+1}\,\varphi_{N+1}^{\mathrm{T}} \tag{3.3.10}$$

$$\Phi_{N+1}\Lambda_{N+1}Y_{N+1}^{\mathrm{T}} = P_N^{-1}\,\widehat{\theta}_N + \varphi_{N+1}\,\alpha_{N+1}\,y_{N+1}\,. \tag{3.3.11}$$

Thus,

$$\widehat{\theta}_{N+1} = P_{N+1}\,[\,P_N^{-1}\,\widehat{\theta}_N + \varphi_{N+1}\,\alpha_{N+1}\,y_{N+1}\,] \tag{3.3.12}$$

$$= P_{N+1}\,[(P_{N+1}^{-1} - \varphi_{N+1}\,\alpha_{N+1}\,\varphi_{N+1}^{\mathrm{T}})\,\widehat{\theta}_N + \varphi_{N+1}\,\alpha_{N+1}\,y_{N+1}] \tag{3.3.13}$$

$$= \widehat{\theta}_N + P_{N+1}\,\varphi_{N+1}\,\alpha_{N+1}\,[y_{N+1} - \varphi_{N+1}^{\mathrm{T}}\,\widehat{\theta}_N]\,, \tag{3.3.14}$$

and we have the expression of the estimate at time $N+1$, $\widehat{\theta}_{N+1}$, in terms of the estimate at time N, $\widehat{\theta}_N$, the new observation y_{k+1}, and an auxiliary matrix P_{N+1} of given fixed dimensions, into which the information of all the observed data is condensed.

Further simplification can be done using the matrix inversion lemma that

$$[A + BCD]^{-1} = A^{-1} - A^{-1}B[DA^{-1}B + \mathrm{C}^{-1}]^{-1}DA^{-1}\,, \tag{3.3.15}$$

where A, B, C and D matrices are of compatible dimensions so that the product BCD and the sum $A + BCD$ exist.

Eq.(3.3.10) can be written as

$$P_{N+1} = [\,P_N^{-1} + \varphi_{N+1}\,\alpha_{N+1}\,\varphi_{N+1}^{\mathrm{T}}\,]^{-1}\,, \tag{3.3.16}$$

and if P_N^{-1}, φ_{N+1}, α_{N+1} and $\varphi_{N+1}^{\mathrm{T}}$ are identified as A, B, C and D matrices respectively in the matrix inversion lemma, the update equation for P becomes

$$P_{N+1} = P_N - \frac{P_N\,\varphi(N+1)\,\varphi^{\mathrm{T}}(N+1)\,P_N}{\frac{1}{\alpha_{N+1}} + \varphi^{\mathrm{T}}(N+1)P_N\varphi(N+1)}\,, \tag{3.3.17}$$

and estimate update equation becomes

$$\widehat{\theta}_{N+1} = \widehat{\theta}_N + \frac{P_N \varphi_{N+1}}{\frac{1}{\alpha_{N+1}} + \varphi_{N+1}^{\mathrm{T}} P_N \varphi_{N+1}} [\, y_{N+1} - \widehat{\theta}_N^{\mathrm{T}} \varphi_{N+1}] \,. \tag{3.3.18}$$

3.4 Multivariable Recursive Least Squares

For a stochastic linear system with s inputs and p outputs, the difference equation which relates the inputs and outputs can be written as

$$y_k + A_1 y_{k-1} + \cdots + A_n y_{k-n} = B_1 u_{k-1} + \cdots + B_m u_{k-m} + v_k \,, \tag{3.4.1}$$

where y and v are p-dimensional vectors, u is a s-dimensional vector, A_i are $p \times p$ matrices and B_i are $p \times s$ matrices and if we introduce the polynomials

$$A(q^{-1}) = I + A_1 q^{-1} + \cdots + A_n q^{-n} \tag{3.4.2}$$

$$B(q^{-1}) = \mathrm{B}_1 q^{-1} + \cdots + B_m q^{-m} \,, \tag{3.4.3}$$

Eq.(3.4.1) can be written as

$$A(q^{-1}) y_k = B(q^{-1}) u_k + v_k \,.$$

One way to parametrize this equation is analogous to that of SISO case. First we introduce the vector

$$\varphi_k = \begin{bmatrix} -Y_{k-1} \\ -Y_{k-2} \\ \cdots \\ -Y_{k-n} \\ U_{k-1} \\ U_{k-2} \\ \cdots \\ U_{k-m} \end{bmatrix}, \tag{3.4.4}$$

which is an $(n \cdot p^2 + m \cdot s) \times p$ matrix, where

$$Y_{k-i} = \begin{bmatrix} y_{k-i} & & & \\ & y_{k-i} & & \\ & & \ddots & \\ & & & y_{k-1} \end{bmatrix} \tag{3.4.5}$$

$$U_{k-i} = \begin{bmatrix} u_{k-i} & & & \\ & u_{k-i} & & \\ & & \ddots & \\ & & & u_{k-i} \end{bmatrix} \tag{3.4.6}$$

and let

$$\theta = [A_{11}A_{12}\cdots A_{1p} \ \cdots \ A_{n1}A_{n2}\cdots A_{np} \ B_{11}B_{12}\cdots B_{1p} \ \cdots \ B_{m1}B_{m2}\cdots B_{mp}]^{\mathrm{T}} \tag{3.4.7}$$

where A_{ij} and B_{ij} are the jth rows of A_i and B_i respectively. Then Eq.(3.4.1) can be represented as

$$y_k = \varphi_k^{\mathrm{T}} \theta + v_k \, . \tag{3.4.8}$$

Parallel to the derivation of SISO case, the norm or the cost function to be minimized can be chosen as

$$V_N(\theta) = \frac{1}{N} \sum_{k=1}^{N} [\, y_k - \varphi_k^{\mathrm{T}} \theta \,]^{\mathrm{T}} \Lambda_k^{-1} [\, y_k - \varphi_k^{\mathrm{T}} \theta \,] , \tag{3.4.9}$$

where Λ_k is a $p \times p$ matrix that weighs the relative importance of the components of $\varepsilon_k (= y_k - \varphi_k^{\mathrm{T}} \theta)$, and the least square estimate is

$$\hat{\theta}_N = \left[\sum_{k=1}^{N} \varphi_k \Lambda_k^{-1} \varphi_k^{\mathrm{T}} \right]^{-1} \sum_{k=1}^{N} \varphi_k \Lambda_k^{-1} y_k . \tag{3.4.10}$$

The recursive algorithm becomes

$$\begin{aligned} \hat{\theta}_{N+1} = \hat{\theta}_N + & P_N \varphi_{N+1} \cdot \\ & [\Lambda_{N+1} + \varphi_{N+1}^{\mathrm{T}} P_N \varphi_{N+1}]^{-1} [y_{N+1} - \varphi_{N+1}^{\mathrm{T}} \hat{\theta}_N] \end{aligned} \tag{3.4.11}$$

$$P_{N+1} = P_N - P_N \varphi_{N+1} [\Lambda_{N+1} + \varphi_{N+1}^{\mathrm{T}} P_N \varphi_{N+1}]^{-1} \varphi_{N+1}^{\mathrm{T}} P_N . \tag{3.4.12}$$

The recursive least squares algorithm for MIMO systems is very similar to that of SISO systems, except that matrix inversion is involved.

3.5 Recursive Instrumental Variable Method (RIV)

Consider the linear regression model

$$y_k = \theta^{\mathrm{T}} \varphi_k + v_k . \tag{3.5.1}$$

and assume that the system output is generated according to

$$y_k = \theta_o^{\mathrm{T}} \varphi_k + v_k . \tag{3.5.2}$$

The least squares estimate can be written as

$$\begin{aligned}\widehat{\theta}_N &= \left[\sum_{k=1}^{N} \alpha_k \varphi_k \varphi_k^{\mathrm{T}}\right]^{-1} \sum_{k=1}^{N} \alpha_k \varphi_k (\theta_o^{\mathrm{T}} \varphi_k + v_k) \\ &= \left[\sum_{k=1}^{N} \alpha_k \varphi_k \varphi_k^{\mathrm{T}}\right]^{-1} \left[\sum_{k=1}^{N} \alpha_k \varphi_k \varphi_k^{\mathrm{T}} \theta_o + \alpha_k \varphi_k v_k\right] \\ &= \theta_o + \left[\sum_{k=1}^{N} \alpha_k \varphi_k \varphi_k^{\mathrm{T}}\right]^{-1} \sum_{k=1}^{N} \alpha_k \varphi_k v_k \\ &= \theta_o + \left[\frac{1}{N} \sum_{k=1}^{N} \alpha_k \varphi_k \varphi_k^{\mathrm{T}}\right]^{-1} \left[\frac{1}{N} \sum_{k=1}^{N} \alpha_k \varphi_k v_k\right], \end{aligned} \tag{3.5.3}$$

which shows that $\widehat{\theta}_N$ is likely to tend to θ_o only if

$$\frac{1}{N} \sum_{k=1}^{N} \alpha_k \varphi_k v_k$$

vanishes.

If $\{v_k\}$ is not a white noise sequence and the above condition does not hold because φ_k and v_k are not uncorrelated, the least squares estimate is biased and won't approach θ_o as N gets large.

The instrumental variable method is aimed at overcoming this problem. Since the problem originates from the correlation between φ_k and v_k, if ζ_k, which is chosen to be uncorrelated with v_k and is called the instrumental variable, is used to replace the regressor, φ_k, in the least squares estimate formula such that

$$\widehat{\theta}_N = \left[\sum_{k=1}^{N} \alpha_k \zeta_k \varphi_k^{\mathrm{T}}\right]^{-1} \sum_{k=1}^{N} \alpha_k \zeta_k y_k \,, \tag{3.5.4}$$

then the convergence result becomes

$$\widehat{\theta}_N = \theta_o + \left[\frac{1}{N} \sum_{k=1}^{N} \alpha_k \zeta_k \varphi_k^{\mathrm{T}}\right]^{-1} \left[\frac{1}{N} \sum_{k=1}^{N} \alpha_k \zeta_k v_k\right] . \tag{3.5.5}$$

Because ζ_k and v_k are uncorrelated,

$$\frac{1}{N}\sum_{k=1}^{N}\alpha_k\,\zeta_k\,v_k$$

approaches the sample mean of $\zeta_N v_N$, and $\widehat{\theta}_N$ tend to θ_o as long as v_k is zero mean and

$$\lim_{N\to\infty}\frac{1}{N}\sum_{k=1}^{N}\alpha_k\,\zeta_k\,\varphi_k^{\mathrm{T}}$$

is invertible.

A choice for the instrumental variable ζ_k should be one that is sufficiently correlated to the regressor φ_k but uncorrelated with the system noise terms v_k. A well-known choice is

$$\zeta_k = \quad [\,-y_{m,k-1}\ \cdots\ -y_{m,k-n}\ \ u_{k-1}\ \cdots\ u_{k-m}\,]^{\mathrm{T}}, \tag{3.5.6}$$

where y_m is the output of a reference model

$$y_{m,k}+\widehat{a}_1 y_{m,k-1}+\ \cdots\ +\widehat{a}_n y_{m,k-n}=\widehat{b}_1 u_{k-1}+\ \cdots\ +\widehat{b}_m u_{k-m}\,. \tag{3.5.7}$$

The model is based on the current estimate such that

$$y_{m,k}=\ \widehat{\theta}_k^{\mathrm{T}}\,\zeta_k \tag{3.5.8}$$

$$\widehat{\theta}_k=[\,\widehat{a}_{1,k}\cdots\widehat{a}_{n,k}\ \ \widehat{b}_{1,k}\cdots\widehat{b}_{n,k}]\,, \tag{3.5.9}$$

and is driven by actual input u_k assuming that no noise exists.

A recursive instrumental variable algorithm can be derived analogous to that of the least squares method as

$$\widehat{\theta}_{N+1}=\widehat{\theta}_N+\frac{P_N\,\zeta_{N+1}}{\frac{1}{\alpha_{N+1}}+\varphi_{N+1}^{\mathrm{T}}P_N\zeta_{N+1}}\,[\,y_{N+1}\ -\widehat{\theta}_N^{\mathrm{T}}\varphi_{N+1}\,] \tag{3.5.10}$$

$$P_{N+1} = P_N - \frac{P_N \zeta_{N+1} \varphi_{N+1}^{\mathrm{T}} P_N}{\frac{1}{\alpha_{N+1}} + \varphi_{N+1}^{\mathrm{T}} P_N \zeta_{N+1}} \ . \tag{3.5.11}$$

3.6 Recursive Maximum Likelihood Method (RML)

Recursive maximum likelihood method applied to the ARMAX model is one of the recursive prediction error identification methods. With the prediction error defined by $\varepsilon_k(\theta) = y_k - \widehat{y}_k(\theta)$, a cost function to measure how well the model θ performs can be chosen as

$$V_N(\theta) = \frac{1}{2} \sum_{k=1}^{N} \varepsilon_k^2(\theta) \ , \tag{3.6.1}$$

which is the sum of squared prediction errors.

To minimize the cost function, we need to take the partial derivative of $V_N(\theta)$ with respect to θ and the result is

$$\begin{aligned} \frac{\partial V_N}{\partial \theta} &= \frac{1}{2} \sum_{k=1}^{N} \frac{\partial}{\partial \theta} \varepsilon_k^2(\theta) \\ &= \sum_{k=1}^{N} \varepsilon_k(\theta) \frac{\partial}{\partial \theta} \varepsilon_k(\theta) \\ &= - \sum_{k=1}^{N} \varepsilon_k(\theta) \frac{\partial}{\partial \theta} \widehat{y}_k(\theta) \ . \end{aligned} \tag{3.6.2}$$

Since $\widehat{y}_k(\theta)$ is a nonlinear function of θ, no analytical result of $\widehat{\theta}_N$ can be obtained to minimize $V_N(\theta)$. Some numerical minimization algorithm is needed to determine $\widehat{\theta}_N$.

In order to obtain a $\widehat{\theta}_k$ that approximately minimizes $V_k(\theta)$, we assume that the next estimate $\widehat{\theta}_k$ to be found is in a small neighborhood of $\widehat{\theta}_{k-1}$ and by means of a Taylor expansion of $V_k(\theta)$ around $\widehat{\theta}_{k-1}$, we have

$$V_k(\theta) \simeq V_k(\widehat{\theta}_{k-1}) + V_k'(\widehat{\theta}_{k-1}) \, [\, \theta - \widehat{\theta}_{k-1}]$$

$$+\frac{1}{2}[\,\theta-\hat{\theta}_{k-1}]^{\mathrm{T}}V_k''(\hat{\theta}_{k-1})[\,\theta-\hat{\theta}_{k-1}]\,, \tag{3.6.3}$$

where the prime denotes differentiation with respect to θ . Minimization of this equation with respect to θ gives

$$\hat{\theta}_k \simeq \hat{\theta}_{k-1}-[\,V_k''(\hat{\theta}_{k-1})\,]^{-1}V_k'(\hat{\theta}_{k-1})\,. \tag{3.6.4}$$

Now, if we define

$$\psi_k(\theta) = -\frac{d}{d\theta}\,\varepsilon_k(\theta)\,,$$

then we have

$$V_k'(\theta) = -\sum_{k'=1}^{k}\psi_{k'}(\theta)\,\varepsilon_{k'}(\theta) = V_{k-1}'(\theta)-\psi_k(\theta)\,\varepsilon_k(\theta) \tag{3.6.5}$$

$$V_k''(\theta) = V_{k-1}''(\theta)+\psi_k(\theta)\,\psi_k^{\mathrm{T}}(\theta)+\varepsilon_k''(\theta)\,\varepsilon_k(\theta)\,. \tag{3.6.6}$$

Since $\hat{\theta}_k$ is in the neighborhood of $\hat{\theta}_{k-1}$, we claim that

$$V_k''(\hat{\theta}_k) = V_k''(\hat{\theta}_{k-1})\,, \tag{3.6.7}$$

and assuming that $\hat{\theta}_{k-1}$ is indeed the optimal estimate at the time $k-1$, we have

$$V_{k-1}'(\hat{\theta}_{k-1}) = 0\,. \tag{3.6.8}$$

Finally, because $\{\varepsilon_k(\theta)\}$ is almost white when the estimate is close to the real value, it can be considered to be zero mean and independent of $\varepsilon_k''(\theta)$. Thus we are able to set

$$\varepsilon_k''(\hat{\theta}_{k-1})\,\varepsilon_k(\hat{\theta}_{k-1}) = 0\,. \tag{3.6.9}$$

If we denote $V_k''(\hat{\theta}_{k-1})$ by P_k^{-1}, then from Eq.(3.6.6) we have

$$P_k^{-1} = P_{k-1}^{-1} + \psi_k(\hat{\theta}_{k-1})\,\psi_k^{\mathrm{T}}(\hat{\theta}_{k-1})\,. \tag{3.6.10}$$

and from Eq.(3.6.5) we have

$$\begin{aligned} V_k'(\hat{\theta}_{k-1}) &= V_{k-1}'(\hat{\theta}_{k-1}) - \psi_k(\hat{\theta}_{k-1})\,\varepsilon_k(\hat{\theta}_{k-1}) \\ &= -\psi_k(\hat{\theta}_{k-1})\,\varepsilon_k(\hat{\theta}_{k-1})\,. \end{aligned} \tag{3.6.11}$$

Thus, Eq.(3.6.4) can be written by

$$\hat{\theta}_k \simeq \hat{\theta}_{k-1} + P_k\,\psi_k(\hat{\theta}_{k-1})\,\varepsilon_k(\hat{\theta}_{k-1})\,, \tag{3.6.12}$$

and by using the matrix inversion lemma, Eq.(3.6.10) can be transformed to

$$P_k = P_{k-1} - \frac{P_{k-1}\,\psi_k(\hat{\theta}_{k-1})\,\psi_k^{\mathrm{T}}(\hat{\theta}_{k-1})\,P_{k-1}}{1+\psi_k^{\mathrm{T}}(\hat{\theta}_{k-1})\,P_{k-1}\,\psi_k(\hat{\theta}_{k-1})}\,. \tag{3.6.13}$$

In order to apply Eqs.(3.6.12) and (3.6.13), $\psi_k(\hat{\theta}_{k-1})$ and $\varepsilon_k(\hat{\theta}_{k-1})$ must be determined first. From the expression for the gradient of the prediction error

$$\psi_k(\theta) = -\frac{d}{d\theta}\,\varepsilon_k(\theta) = \frac{d}{d\theta}\,\hat{y}_k(\theta)\,, \tag{3.6.14}$$

we have

$$\frac{\partial}{\partial a_i}\,\hat{y}_k(\theta) = \frac{1}{C(q^{-1})}\,y_{k-i} \tag{3.6.15}$$

$$\frac{\partial}{\partial b_i}\,\hat{y}_k(\theta) = \frac{1}{C(q^{-1})}\,u_{k-i} \tag{3.6.16}$$

$$\frac{\partial}{\partial c_i}\,\hat{y}_k(\theta) = \frac{1}{C(q^{-1})}\,\varepsilon_{k-i}(\theta)\,, \tag{3.6.17}$$

and it can be seen that

$$\psi_k(\theta) = \frac{1}{C(q^{-1})}\,\varphi_k(\theta)\,. \tag{3.6.18}$$

If we make the approximations $\varepsilon_k = \varepsilon_k(\widehat{\theta}_{k-1})$ and $\psi_k = \psi_k(\widehat{\theta}_{k-1})$, a recursive algorithm can be obtained as

$$\varepsilon_k = y_k - \widehat{\theta}_{k-1}^{\mathrm{T}}\,\varphi_k \tag{3.6.19}$$

$$P_k = P_{k-1} - \frac{P_{k-1}\,\psi_k\,\psi_k^{\mathrm{T}}\,P_{k-1}}{1 + \psi_k^{\mathrm{T}}\,P_{k-1}\,\psi_k} \tag{3.6.20}$$

$$\widehat{\theta}_k = \widehat{\theta}_{k-1} + P_k\,\psi_k\,\varepsilon_k \tag{3.6.21}$$

$$\psi_k = -\widehat{c}_1\,\psi_{k-1} - \widehat{c}_2\,\psi_{k-2} - \cdots - \widehat{c}_r\,\psi_{k-r} + \varphi_k\,, \tag{3.6.22}$$

where

$$\varphi_k \triangleq [-y_{k-1}\ \cdots\ -y_{k-n}\quad u_{k-1}\ \cdots\ u_{k-m}\ \varepsilon_{k-1}\ \cdots\ \varepsilon_{k-r}]^{\mathrm{T}}. \tag{3.6.23}$$

3.7 Extended Least Squares (ELS)

The extended least squares method belongs to another category of parameter estimation methods called pseudo-linear regression methods when ARMAX models are considered. It is obtained by neglecting the implicit θ-dependence in $\varphi_k(\theta)$ when calculating the gradient of the prediction such that

$$\frac{d}{d\theta}\,\widehat{y}_k(\theta) \simeq \varphi_k(\theta)\,. \tag{3.7.1}$$

By using the expression in Eq.(3.7.1) for gradient, we get the following algorithm

$$\varepsilon_k = y_k - \widehat{\theta}_{k-1}^{\mathrm{T}}\varphi_k \tag{3.7.2}$$

$$P_k = P_{k-1} - \frac{P_{k-1}\,\varphi_k\,\varphi_k^{\mathrm{T}}\,P_{k-1}}{1 + \varphi_k^{\mathrm{T}}\,P_{k-1}\,\varphi_k} \tag{3.7.3}$$

$$\widehat{\theta}_k = \widehat{\theta}_{k-1} + P_k \ \varphi_k \ \varepsilon_k \tag{3.7.4}$$

3.8 Stochastic Approximation Approach

Given a sequence of random variables $\{e_k\}$ and a function $Q(x, e_k)$ of e_k and x, where the distribution of e_k is unknown but the values of $Q(x, e_k)$ are observed or can be constructed for any chosen x. The problem of stochastic approximation is to find the solution x to the equation

$$\mathbb{E}\{Q(x, e_k\} = f(x) = 0 \ , \tag{3.8.1}$$

where $\mathbb{E}$ denotes expectation over e_k.

Robbins and Monro suggested the following recursive algorithm

$$\widehat{x}_{k+1} = \widehat{x}_k + \gamma_{k+1} \ Q(\widehat{x}_k, e_{k+1}) \ , \tag{3.8.2}$$

where γ is a sequence of positive scalars tending to zero.

3.8.1 Least Mean Squares (LMS) Method

Consider the linear regression model

$$y_k = \theta^{\mathrm{T}} \varphi_k + v_k \ , \tag{3.8.3}$$

and θ is to be found by minimizing the variance of the equation error v_k

$$V(\theta) = \frac{1}{2} \mathbb{E}[\ y_k - \varphi_k^{\mathrm{T}} \ \theta \ ]^2 \ . \tag{3.8.4}$$

The necessary condition for $V(\theta)$ to be minimized is

$$\left[-\frac{d}{d\theta} V(\theta) \right] = \mathbb{E} \ \varphi_k \ [\ y_k - \varphi_k^{\mathrm{T}} \ \theta] = 0 \ . \tag{3.8.5}$$

Here the stochastic approximation comes into play since the probability

distribution of (y_k, φ_k) is not known and the expectation in Eq. (3.8.4) and Eq. (3.8.5) can not be evaluated.

If the Robbins-Monro stochastic approximation scheme is applied to the linear regression model, a recursive algorithm is obtained as

$$\widehat{\theta}_k = \widehat{\theta}_{k-1} + \gamma_k \varphi_k [\, y_k - \varphi_k^{\mathrm{T}} \widehat{\theta}_{k-1}] \,, \tag{3.8.6}$$

when we set

$$x = \theta$$

$$e_k = \begin{bmatrix} y_k \\ \varphi_k \end{bmatrix}$$

$$Q(x, e_k) = \varphi_k [\, y_k - \varphi_k^{\mathrm{T}} \theta \,] \,. \tag{3.8.7}$$

Eq.(3.8.6) is the well-known LMS algorithm where $\{\gamma_k\}$ is the gain sequence. Several choices for $\{\gamma_k\}$ are

$$\gamma_k = \gamma_o \ (constant)$$

$$\gamma_k = \gamma_o \,/\, |\, \varphi_k \,|^{\,2} \ (normalized)$$

$$\gamma_k = \left[\sum_{k'=1}^{k} |\, \varphi_k \,|^{\,2} \right]^{-1} \ (normalized \ and \ decreasing)$$

3.8.2 A Stochastic Newton Method

The Robbins-Monro scheme can be looked upon as a stochastic gradient method if we consider the minimization problem

$$\min_x V(x), \tag{3.8.8}$$

$$V(x) = \mathbb{E}\,[H(x, e_k)] \,. \tag{3.8.9}$$

Let

$$-\frac{d}{dx}V(x) = f(x)\ , \tag{3.8.10}$$

and assume that the gradient

$$-\frac{d}{dx}\ H(x, e_k) = Q(x, e_k) \tag{3.8.11}$$

can be obtained for any chosen x . Thus, the problem becomes solving the equation

$$0 = \ -\frac{d}{dx}V(x) = f(x)\ \ = \mathbb{E}\{Q(x, e_k)\}\ , \tag{3.8.12}$$

and the Robbins-Monro scheme Eq.(3.8.2) is regarded as an algorithm to minimize $V(x)$ in Eq.(3.8.8) . The adjustment of x is made in the direction that is the negative gradient of $H(x, e_k)$.

A more efficient adjustment direction is the well-known Newton direction

$$-\left[\frac{d^2}{dx^2}\ V(x)\right]^{-1}\ \left[\frac{d}{dx}V(x)\right], \tag{3.8.13}$$

based upon which the Robbins-Monro scheme can be modified as

$$\widehat{x}_{k+1} = \widehat{x}_k + \gamma_{k+1}\ [\bar{V}''(\widehat{x}_k, e^{k+1})]^{-1}\ Q(\widehat{x}_k, e_{k+1})\ , \tag{3.8.14}$$

where $\bar{V}''(\widehat{x}_k, e^{k+1})$ is the approximated Hessian which depends on all the previous noise values $e^{k+1} = \{e_{k+1} \cdots e_1\}$.

When the linear regression problem is considered, we have

$$\frac{d^2}{d\theta^2}\ V(\theta)\ \ = \ \mathbb{E}\{\varphi_k\ \varphi_k^{\mathrm{T}}\}\ , \tag{3.8.15}$$

and since the Hessian is independent of θ, it can be determined as R of the equation

$$\mathbb{E}\{\varphi_k\ \varphi_k^{\mathrm{T}}\ -\ R\} = 0\ . \tag{3.8.16}$$

Applying Robbins-Monro scheme to the above equation results in

$$R_k = R_{k-1} + \gamma_k \left[\varphi_k \varphi_k^{\mathrm{T}} - R_{k-1} \right] , \tag{3.8.17}$$

and the stochastic Newton algorithm is obtained as

$$\widehat{\theta}_k = \widehat{\theta}_{k-1} + \gamma_k R_k^{-1} \varphi_k \left[y_k - \varphi_k^{\mathrm{T}} \widehat{\theta}_{k-1} \right] . \tag{3.8.18}$$

The algorithm is closely related to the recursive least squares algorithm since if $\gamma_k = 1/k$, Eq.(3.8.18) is the same as Eq.(3.3.18) for $\alpha_k = 1$ and Eq.(3.8.17) can be shown to be the same as Eq.(3.3.17) by applying the matrix inversion lemma.

3.9 Bayesian Approach

The parameter to be estimated in the Bayesian approach is taken to be a random variable. The information about its value is inferred based upon observations of other random variables that are correlated with the parameter.

Suppose that a dynamic system is parametrized by a vector θ, we consider θ to be a random vector with a certain prior distribution. With the observations $\{y^k, u^k\}$, where

$$y^k = \left[y_k \; y_{k-1} \; \cdots \; y_1 \right]$$

$$u^k = \left[u_k \; u_{k-1} \; \cdots \; u_1 \right] ,$$

we are interested in obtaining the posterior probability density function for θ, which is $p(\theta \mid y^k, u^k)$ and also the estimate $\widehat{\theta}_k$ which can be the conditional expectation:

$$\widehat{\theta}_k = \mathbb{E}(\theta \mid y^k, u^k) , \tag{3.9.1}$$

and $\widehat{\theta}_k$ can be considered to be the estimate that minimizes $\mathbb{E}[\,|\,\theta - \widehat{\theta}_k\,|^2]$.

The well-known Kalman filter is developed in such a Bayesian framework. In the following, we show that the Bayesian solution to linear regression problem is a special case of Kalman filter.

Given a state space model of a dynamic system

$$x_{k+1} = F_k x_k + w_k \tag{3.9.2}$$

$$y_k = H_k x_k + e_k \tag{3.9.3}$$

where $\{w_k\}$ and $\{e_k\}$ are mutually independent sequences of independent random vectors with zero mean and covariance matrices R and r . The solution to the problem of determination of

$$\widehat{x}_{k+1} = \mathbb{E}[\,x_{k+1}\,|\,y^k] \tag{3.9.4}$$

is the Kalman filter

$$\widehat{x}_{k+1} = F_k\,\widehat{x}_k \;+\; K_k[\,y_k - H_k\widehat{x}_k] \tag{3.9.5}$$

$$K_k = \frac{F_k\,P_k\;H_k^{\mathrm{T}}}{r_k \;+\; H_k\,P_k\,H_k^{\mathrm{T}}} \tag{3.9.6}$$

$$\begin{aligned} P_{k+1} = {} & F_k\,P_k\;F_k^{\mathrm{T}} + R_k \\ & - F_k\,P_k\;H_k^{\mathrm{T}}\,[\,r_k + H_k\,P_k\;H_k^{\mathrm{T}}\,]^{-1} H_k\,P_k\;F_k^{\mathrm{T}}\,. \end{aligned} \tag{3.9.7}$$

Now, if we set $F_k = I$, $R_k = 0$, and $H_k = \varphi_k^{\mathrm{T}}$, we obtain the linear regression model

$$y_k = \varphi_k^{\mathrm{T}}\,\theta + e_k\,, \tag{3.9.8}$$

where x is recognized as θ and the recursive algorithm for the estimate of θ becomes

$$\widehat{\theta}_k = \widehat{\theta}_{k-1} + K_k[\, y_k - \widehat{\theta}_{k-1}\varphi_k] \tag{3.9.9}$$

$$K_k = \frac{P_{k-1}\ \varphi_k}{r_k\ +\ \varphi_k^{\mathrm{T}}\ P_{k-1}\ \varphi_k} \tag{3.9.10}$$

$$P_k = P_{k-1} - \frac{P_{k-1}\ \varphi_k\ \varphi_k^{\mathrm{T}}\ P_{k-1}}{r_k\ +\ \varphi_k^{\mathrm{T}}\ P_{k-1}\ \varphi_k} \tag{3.9.11}$$

$$\widehat{\theta}_0 = \theta_0\ . \tag{3.9.12}$$

3.10 Joint Estimation Methods

Based upon a linear model, the Kalman filter is known to be the optimal state estimator when a linear filtering problem is considered. In the cases that states of the system are not linearly related, those linear algorithms of the Kalman filter can be adapted to the non-linear environment and a suboptimal solution can be obtained for the non-linear systems. The problem of joint estimation is concerned with taking parameters of the system as extra states and estimating the parameters and the states together. Since the parameters and the states of the system are not linearly related, it can be formulated as a non-linear filtering problem. The extended Kalman filter has been the most famous solution to non-linear filtering problems. In this section, the extended Kalman filter will be derived as a standard and exact Kalman filter of a linearized model of a non-linear system.

In [23], Salut *et.al.* gave a presentation of a canonical representation of MIMO linear stochastic systems, in which, for the joint state and parameter estimation problem, all unknown parameters appear linearly when an input-output record is available. This extended system was also derived independently in [24] by Van Landingham and an algorithm, which now is known as Pseudo Linear Identification (PLID), was developed extensively in [25]. In this section,

the extended state representation of SISO systems is briefly summarized.

3.10.1 Extended Kalman Filter (EKF)

The basic premise of the use of the extended Kalman filter is that the non-linear system can be well linearized by employing the best estimates of the state vector as the reference values used at each stage for the linearization. Thus, given the non-linear model

$$x_{k+1} = f_k(x_k) + g_k(x_k)\, w_k \tag{3.10.1}$$

$$z_k = h_k(x_k) + v_k\ , \tag{3.10.2}$$

we denote

$$F_k = \left.\frac{\partial f_k(x)}{\partial x}\right|_{x\,=\,\widehat{x}_{k/k}} \tag{3.10.3}$$

$$H_k^{\mathrm{T}} = \left.\frac{\partial h_k(x)}{\partial x}\right|_{x\,=\,\widehat{x}_{k/k-1}} \tag{3.10.4}$$

$$G_k = g_k(\widehat{x}_{k/k})\ . \tag{3.10.5}$$

The non-linear functions $f_k(x_k)$, $g_k(x_k)$, and $h_k(x_k)$, if sufficiently smooth, can be expanded in Taylor series about the conditional means $\widehat{x}_{k/k}$ and $\widehat{x}_{k/k-1}$ as

$$f_k(x_k) \;=\; f_k(\widehat{x}_{k/k}) \;+\; F_k(x_k - \widehat{x}_{k/k}) \;+\; \cdots \tag{3.10.6}$$

$$g_k(x_k) \;=\; g_k(\widehat{x}_{k/k}) \;+\; \cdots \;=\; G_k \;+\; \cdots \tag{3.10.7}$$

$$h_k(x_k) \;=\; h_k(\widehat{x}_{k/k-1}) \;+\; H_k^{\mathrm{T}}(x_k - \widehat{x}_{k/k-1}) \;+\; \cdots\ . \tag{3.10.8}$$

Neglecting higher order terms and assuming knowledge of $\widehat{x}_{k/k}$ and $\widehat{x}_{k/k-1}$, Eq.(3.10.1) and (3.10.2) can be approximated as

$$x_{k+1} = F_k\, x_k + G_k\, w_k + f_k(\widehat{x}_{k/k}) - F_k \widehat{x}_{k/k} \tag{3.10.9}$$

$$z_k = H_k^{\mathrm{T}}\, x_k \, + v_k \, + h_k(\widehat{x}_{k/k-1}) \, - \, H_k^{\mathrm{T}}\, \widehat{x}_{k/k-1} \, , \tag{3.10.10}$$

and the Kalman filter can be derived for the above linear model.

3.10.2 Pseudo-Linear Identification (PLID)

Consider an observable canonical discrete state model

$$x_{k+1} = A\, x_k \, + \, B\, u_k \tag{3.10.11}$$

$$y_k = C\, x_k \, + \, D\, u_k \, , \tag{3.10.12}$$

where

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & a_0 \\ 1 & 0 & \cdots & 0 & 0 & a_1 \\ & & \vdots & & & \\ 0 & 0 & \cdots & 0 & 1 & a_{n-1} \end{bmatrix} , \qquad B = \begin{bmatrix} b_0 + a_o b_n \\ b_1 + a_1 b_n \\ \vdots \\ b_{n-1} + a_{n-1} b_n \end{bmatrix}$$

$$C = [\, 0 \quad 0 \quad \cdots \quad 0 \quad 0 \quad 1 \,] \quad \text{and} \quad D = [\, b_n \,] \, .$$

We define the vector p that contains the unknown parameters of the system as

$$p = [\, a_0 \;\; a_1 \;\; a_2 \;\; \cdots \;\; a_{n-1} \;\; b_0 \;\; b_1 \;\; b_2 \;\; \cdots \;\; b_n \,]^{\mathrm{T}}, \tag{3.10.13}$$

and Eq.(3.10.11) can be written as

$$x_{k+1} = J\, x_k + a\, y_k + b\, u_k \, , \tag{3.10.14}$$

where

$$a = [a_0 \;\; a_1 \;\; a_2 \;\; \cdots \;\; a_{n-1} \,]^{\mathrm{T}} \, , \qquad b = [b_0 \;\; b_1 \;\; b_2 \;\; \cdots \;\; b_n \,]^{\mathrm{T}}$$

and

$$J = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 & 0 \\ & & \vdots & & & \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}.$$

If the observation vector is defined as

$$z_k = [\ u_k \ \ y_k\ ]^{\mathrm{T}}\ , \tag{3.10.15}$$

then with some noise n and v included, the extended state model is obtained as

$$s_{k+1} = F_k\ s_k \tag{3.10.16}$$

$$y_k = H_k\ s_k + v_k\ , \tag{3.10.17}$$

where $s_k = [\ x_k^{\mathrm{T}} \ \ p^{\mathrm{T}}\ ]$ and

$$F[z_k] = \begin{bmatrix} J & I_n y_k & I_n u_k & 0 \\ 0 & & I_{2n+1} & \end{bmatrix}$$

$$H[z_k] = [\quad C \qquad 0 \qquad 0 \qquad u_k \quad]\ .$$

Thus, for a completely unknown system, a linear estimator can be used to obtain estimates of the unknown parameters together with system states as long as the autocovariance of the state noise and the system input and output are known.

3.11 Implications for Real-Time Adaptive Control

Consider a linear MIMO system which can be described by an ARX model

$$y_k + A_1 y_{k-1} + \cdots + A_n y_{k-n} = B_1 u_{k-1} + \cdots + B_m u_{k-m} + v_k\ .$$

Based upon the formulation in Section 3.4, the number of unknown parameters to be determined in the model is $np^2 + msp$ if the number of inputs and outputs are s and p respectively. Most recursive estimation algorithms in the previous review involve a vector update equation for the parameters and a matrix update equation for the generalized covariance. Since both equations involve a $(p \times p)$ matrix inversion operation and many matrix multiplications, the computation load will be determined by p and $q(= np^2 + msp)$. For the recursive least squares algorithm, the approximate numbers of operations are listed in Table 3.1 . If the recursive least squares algorithm is used in adaptive control, these operations represent the extra computation load added to the controller just for estimating the unknown model parameters. For a time-varying Kalman filter, the error covariance propagation enormously increases the computation involved in a time-invariant Kalman filter. Similarly, the generalized covariance update, Eq.(3.4.12), in the recursive least squares algorithm also requires enormous computation.

The LMS algorithm derived as a stochastic approximation method for parameter estimation is different from the other structurally similar algorithms and is computationally attractive since it does not involve the general covariance matrix update and saves a lot of computations.

The joint estimation methods are able to estimate the parameters along with the state variables, which makes the number of unknowns increase by the the number of the state variables and requires much more intensive computation. Thus, joint estimation methods are attractive only when the state estimate is also required.

Table 3.1 Approximate Number of Operations in RLS Algorithm

Operation	# of Operations	Equations Involved
Multiplication of a $q \times q$ matrix and a $q \times p$ matrix	1	(3.4.11), (3.4.12)
Multiplication of a $p \times q$ matrix and a $q \times p$ matrix	1	(3.4.11), (3.4.12)
Multiplication of a $q \times p$ matrix and a $p \times p$ matrix	1	(3.4.11), (3.4.12)
Multiplication of a $q \times p$ matrix and a $p \times q$ matrix	1	(3.4.12)
Multiplication of a $p \times q$ matrix and a $q \times q$ matrix	1	(3.4.12)
Multiplication of a $q \times p$ matrix and a $p \times 1$ matrix	1	(3.4.11)
Multiplication of a $p \times q$ matrix and a $q \times 1$ matrix	1	(3.4.11)
Inversion of a $p \times p$ matrix	1	(3.4.11), (3.4.12)
Addition of two $p \times p$ matrices	1	(3.4.11)
Addition of two $q \times q$ matrices	1	(3.4.11)
Addition of two $p \times 1$ matrices	1	(3.4.12)
Addition of two $q \times 1$ matrices	1	(3.4.12)

4. Adaptive Control

The control of a dynamic system is aimed at manipulating the inputs to the system in such a way that the outputs of the system satisfy certain specified goals. Closed-loop feedback control, compared with open-loop control, has advantages of increased accuracy, reduced sensitivity to the controlled plant, reduced effect of disturbance, increased speed of response and bandwidth, etc. Thus, feedback control is oriented toward eliminating the effect of state perturbations. However, the performance of the control system will degrade if variation of the dynamic parameters exists in the controlled plant. Adaptive control is oriented toward eliminating the effect of structural perturbations[26], which are caused by the variation of dynamic parameters, upon the performance of the control system. In this chapter, several categories of adaptive controllers are briefly reviewed.

4.1 Definition

Adaptive control is an important field in modern control. During the past four decades, a lot of adaptive schemes have been developed to deal with structural perturbations and environmental variations in control systems and several definitions have been proposed for adaptive control; however, agreement over definitions for adaptive control remains a very controversial issue. It is quite difficult to give a universal definition that can be applied to every situation because of the many forms of uncertainties and different methodologies adopted to tackle the problems. However, in order to control uncertain systems, an adaptive controller usually has a structure that includes a subsystem for the on-line estimation of unknown parameters or system structure and a subsystem for the generation of control inputs based upon the estimated parameters. Thus, adaptive control can be seen as a direct aggregation of a non-adaptive control methodology with some form of recursive system identification.[27] Based upon the basic aspects of adaptive control, the following description of adaptive systems is

considered[28].

> An adaptive system measures a certain index of performance (IP) using the inputs, the states, and the outputs of the adjustable system. From the comparison of the measured IP values and a set of given ones, the adaptation mechanism modifies the parameters of the adjustable system or generates an auxiliary input in order to maintain the IP values close to the set of given ones.

This description is illustrated in Figure 4.1. The performance of the adjustable system in the figure can be adjusted by modifying its parameters or by modifying its inputs. The IP measurement can be done indirectly by identifying dynamic parameters of the system. There are a lot of methods that can be used, several of them are discussed in Chapter 3. The measured IP is compared with the given IP in the comparison and decision block and if the measured IP is decided to be not acceptable, the adaptation mechanism will act to modify the system performance either by modifying the parameter of the adjustable system or by modifying the input signal. Depending upon how the system performance is modified, adaptive control systems can be divided into parameter-adaptive control systems and signal-synthesis adaptive control systems.

Many adaptive control schemes were developed based upon a common framework of conventional feedback control. They all started with an ordinary feedback control loop and a regulator with adjustable parameters. The key problem is to find a convenient way of changing the regulator parameters in response to changes in process and disturbance dynamics[29]. Depending upon how the parameters of the regulator are adjusted, several adaptive schemes can be distinguished. Three most well-known ones are gain scheduling, model reference control and self-tuning control.

In addition to the above adaptive schemes, there are many adaptive strategies proposed based upon different frameworks and different points of view. The least mean squares (LMS) adaptive algorithm, based upon adaptive signal processing concepts, is a feedforward adaptive control approach that has been the subject of extensive research recently.

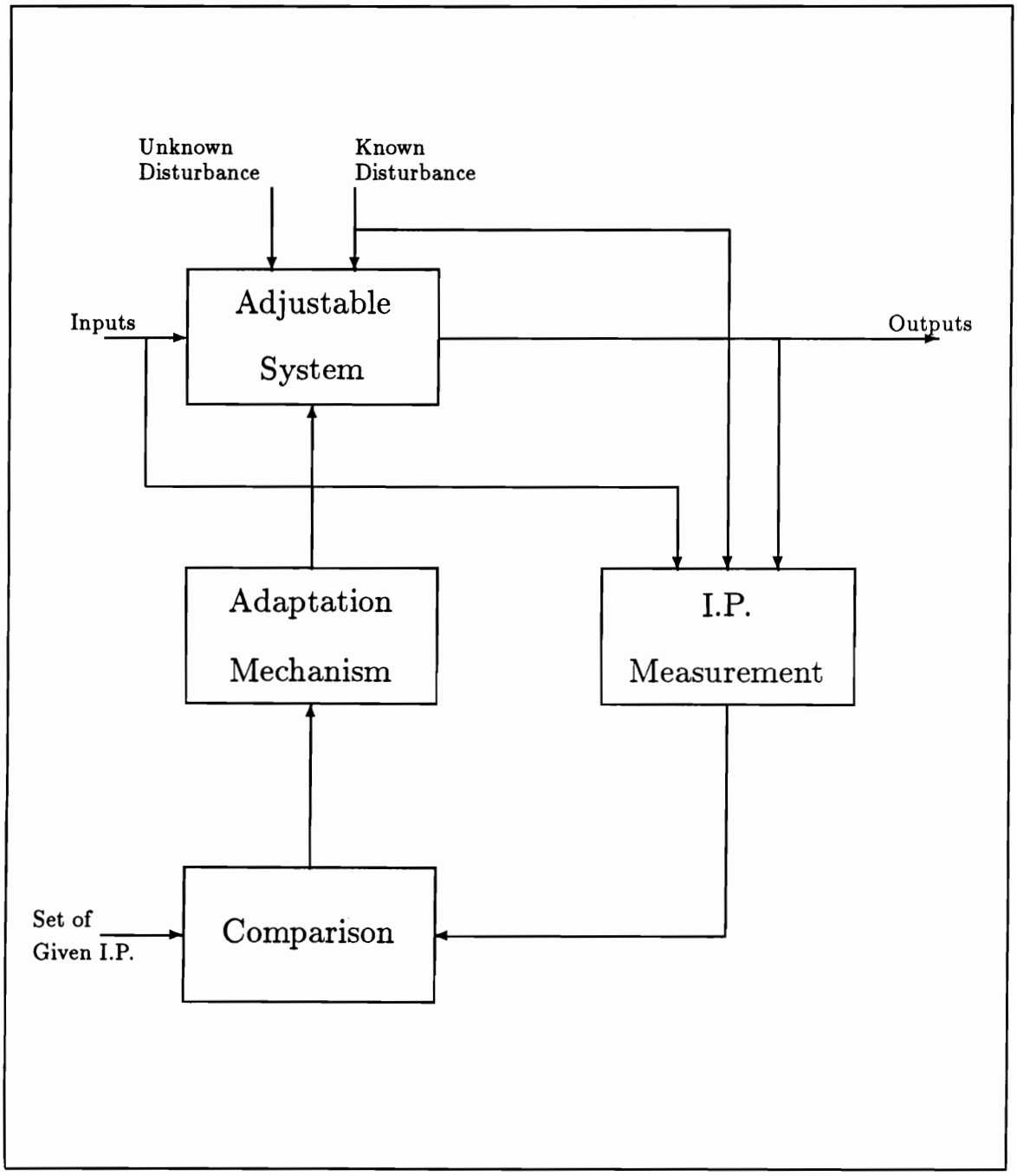


Figure 4.1 Essential Components of an Adaptive Control System

In the following, we will review the three most applied feedback adaptive schemes and a feedforward LMS adaptive scheme.

4.2 Gain Scheduling

Gain scheduling was introduced in the development of flight control systems in the 1950's and 1960's and has also found applications in process control systems. The basic idea is to find auxiliary variables that correlate well with the change in plant dynamics and then the performance of the system can be modified by changing parameters of the controller as functions of the auxiliary variables. This can be illustrated in Figure 4.2. This scheme is quite useful if the plant dynamics depends in a well-known fashion on relatively few easily measurable auxiliary variables. A number of operating conditions are distinguished according to the auxiliary variables and parameters of the controller are determined for these operating conditions by some design methods. The number of operating conditions can be increased or interpolation between controllers can be used to improve adaptivity. Thus, parameters of the controller can be changed very quickly in response to changes in the plant dynamics. However, one drawback of gain scheduling is that it is actually an open-loop adaptation scheme without real learning or intelligence and because of this there is a controversy over whether gain scheduling should be looked upon as an adaptive control method. Furthermore, since the parameters of the controller must be determined for many operating conditions and performance must be checked by computer simulation, the design is quite time-consuming.

4.3 Model Reference Adaptive Control

Model reference adaptive control (MRAC) was first designed for implementation using analog hardware and suggested for aerospace problems in the 1950's[30]. The distinctive feature of MRAC is the construction of a reference model that operates together with the plant and the adaptation takes the form of adjustment of one or more of the controller's parameters or input signals so as to

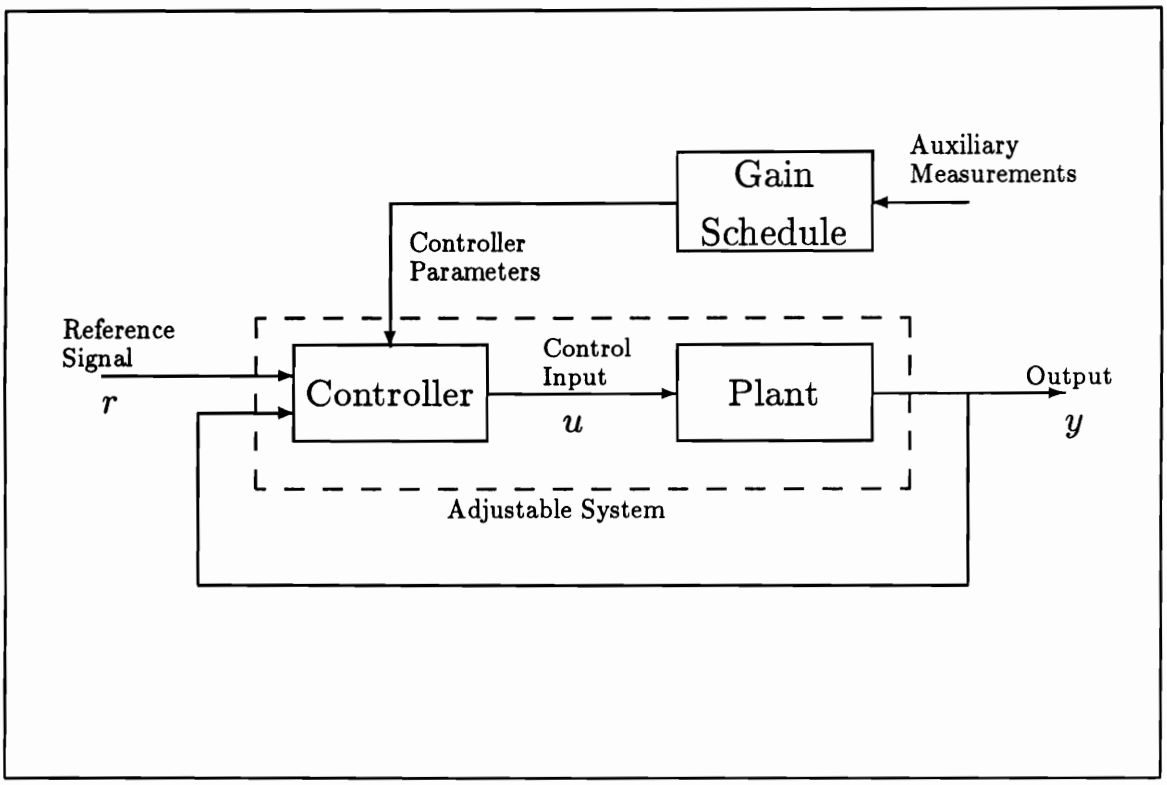


Figure 4.2 Gain Scheduling Control System

force the response of the resulting closed-loop control system towards that of the reference model. The way parameters of the controller are adjusted can be illustrated in Figure 4.3 for a parameter-adaptive model reference control system. Illustrated in Figure 4.4 is a signal-synthesis model reference control system in which the adaptation mechanism generates an auxiliary input to minimize the difference between the output of the reference model and that of the adjustable system. The key problem of MRAC is to determine the adaptation mechanism such that the resulting system is stable and also the error between plant output and model output is brought to zero. The adaptive mechanism used in the original MRAC, now known as the MIT rule, is a modification of the gradient search method.

If $e(\theta)$, a function of the adjustable parameter vector θ of the controller, is the error between model output y_m and plant output y, then in order to make the square of error small, θ is adjusted along the direction of steepest decent, the direction of negative gradient of $e^2(\theta)$ as

$$\begin{aligned}\frac{d\theta}{dt} &= -k\,\frac{\partial}{\partial\theta}\,(e^2(\theta)) \\ &= -2\,k\,e(\theta)\,\frac{\partial}{\partial\theta}\,(e(\theta)) \\ &= -2\,k\,e(\theta)\,\frac{\partial}{\partial\theta}\,(y(\theta)).\end{aligned} \tag{4.3.1}$$

Since the sensitivity function $\frac{\partial}{\partial\theta}(y(\theta))$ usually depends on unknown controller parameters, the MIT rule is obtained by replacing the unknown parameters by their estimate at time t. The MIT rule will work well if the k is kept small; however, it is not possible to guarantee stability of the closed-loop system or convergence of the error to zero. Modified adjustment rule can be designed based upon Lyapunov theory[31] to get provably stable and convergent model reference adaptive schemes and the only difference from the MIT rule is that the sensitivity function is replaced by other functions.

Even though MRAC was originally designed in continuous-time domain and implemented with analog hardware, the use of digital hardware has brought about more flexibility for MRAC designed in the discrete-time domain.

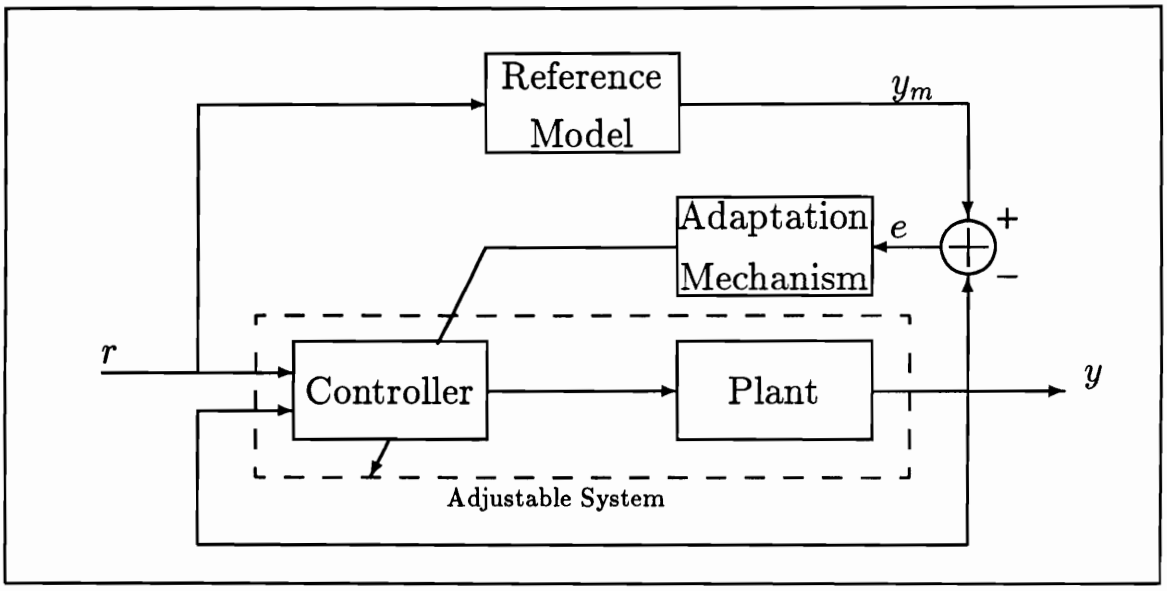


Figure 4.3 Parameter-Adaptive Model Reference Control System

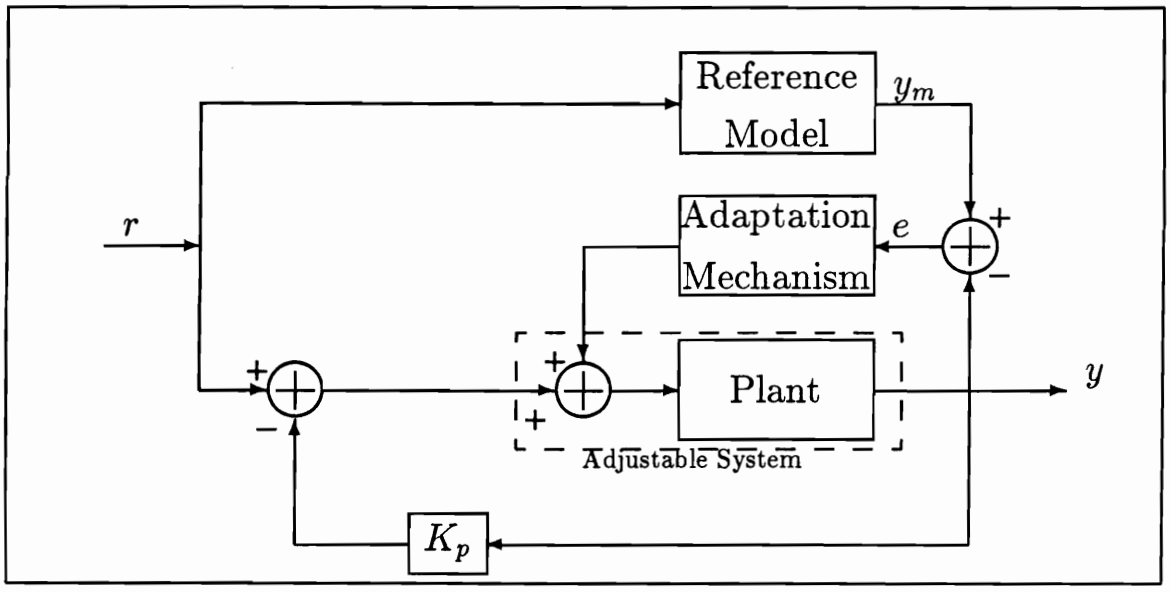


Figure 4.4 Signal-Synthesis Model Reference Control System

Adaptation rules need to be developed directly for discrete-time systems instead of being obtained by discretizing continuous-time adaptation rules, however, time-varying adaptive gains can be easily implemented for discrete-time MRAC systems.

4.4 Self-Tuning Regulator

Self-tuning approach was originally proposed by Kalman[32] and clarified by Astrom and Wittenmark[33]. The regulator can be thought of as composed of two loops. As illustrated in Figure 4.5, the inner loop consists of a conventional feedback controller but with varying parameters and the outer loop consists of a recursive parameter estimator and a design calculation. Thus, after a suitable model structure is determined for the plant, the self-tuning regulator is capable of automating modeling and design by estimating model parameters and using the estimated parameters as if they were the true parameters to obtain the control law. This is usually called *the certainty equivalent principle.* The self-tuning regulator is very flexible with respect to the design method and virtually any design technique can be accommodated. However, the analysis of self-tuning adaptive systems is more complex than the analysis of model reference systems, due primarily to the usually nonlinear transformation from identified parameters to controller parameters.[27]

Among the many design methods that can be adopted in self-tuning control, the LQG design method can be easily applied, even to open-loop unstable or non-minimum phase plants, and is immediately applicable to multivariable systems. As described in Chapter 2, a state-space LQG controller consists of a state estimator with fixed Kalman gains and a set of fixed feedback gains based upon an accurate model. These gains are optimal only if the model is accurate and any variation or uncertainty in the model will result in degraded performance or instability. Self-tuning of an LQG controller is aimed at identifying the model variation or uncertainty and incorporating it into the estimator and the feedback gains calculation such that they are based upon an increasingly accurate model. Intuitively, for an LQG regulator to self-tune its parameters, it is required to estimate all the parameters of the plant using a recursive method and to use these

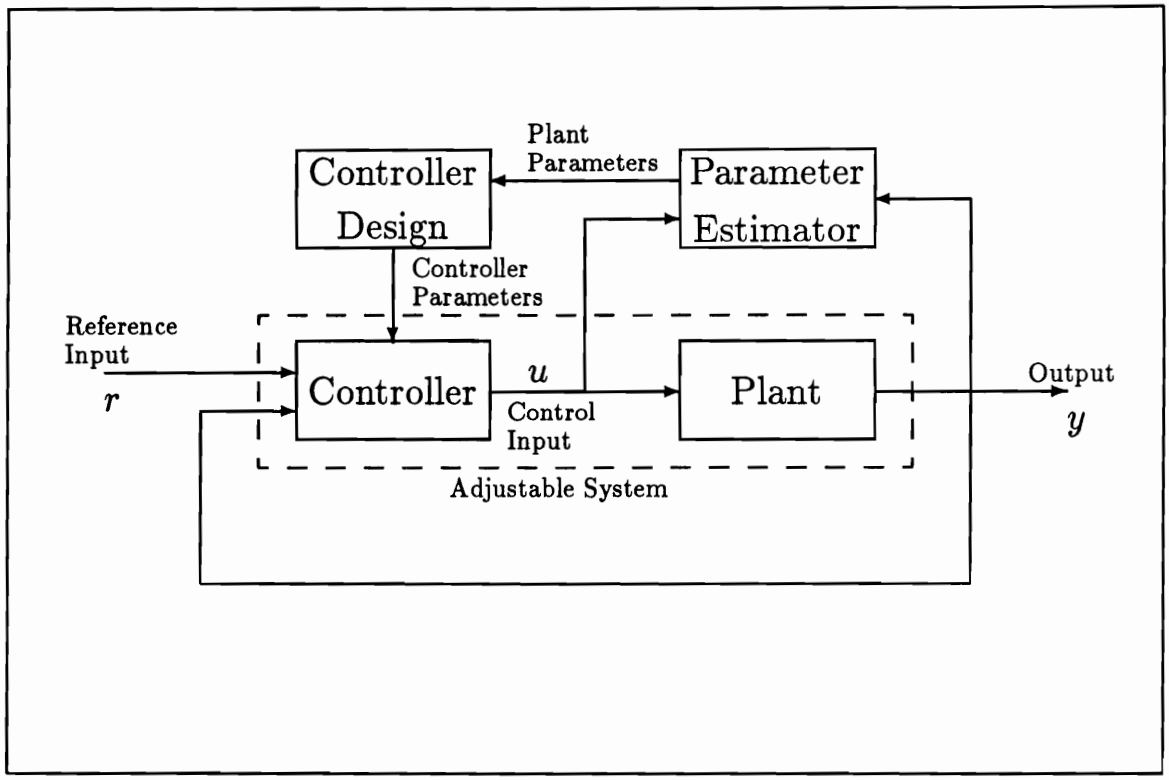


Figure 4.5 Self-Tuning Control System

estimated parameters in the state-space model to reconstruct the states and evaluate optimal feedback gains in a single sampling period. Although this concept is not new, the intensive computation involved in the estimation of the plant parameters, the propagation of the error covariance matrix to get the Kalman gains and the solving the Riccati equation iteratively to calculate the feedback gains prohibits actual implementation. A simplification was made by Lam[34], who proposed the use of an adaptive filter instead of a Kalman filter for state estimation. The computation is reduced by conducting a transmittance matrix calculation instead of propagating the error covariance matrix. However, only a SISO result was shown, the model of the system has to be in a certain canonical form and all the parameters need to be identified

The stochastic optimal control problem can also be considered in the polynomial domain. Kucera[35] first proposed a optimal controller derived by the polynomial approach. The major difficulty when self-tuners are designed based upon this approach is the computation burden associated with conducting a spectral factorization and solving two diophantine equations. Although stable iterative algorithms for the operation of spectral factorization are available, the computation time for each iteration is not fixed. Hence there is no guarantee that the algorithm will converge within a specified time given a particular error bound.

4.5 Feed-Forward LMS Control

An adaptive processor in a closed-loop adaptation configuration is the major portion of many adaptive signal processing systems. The adaptive processor has an input signal x and an output signal y. A signal d is defined as the desired output of the adaptive system and an error signal ε is obtained as the difference between the desired output and the output signal of the adaptive processor y. An adaptive algorithm uses the error signal to adjust the adaptive processor by minimizing some measure of the error signal. This is illustrated in Figure 4.6[36].

Although the output of the processor is indeed affected by the error signal in a closed-loop fashion, the input signal of the processor is not modified by the

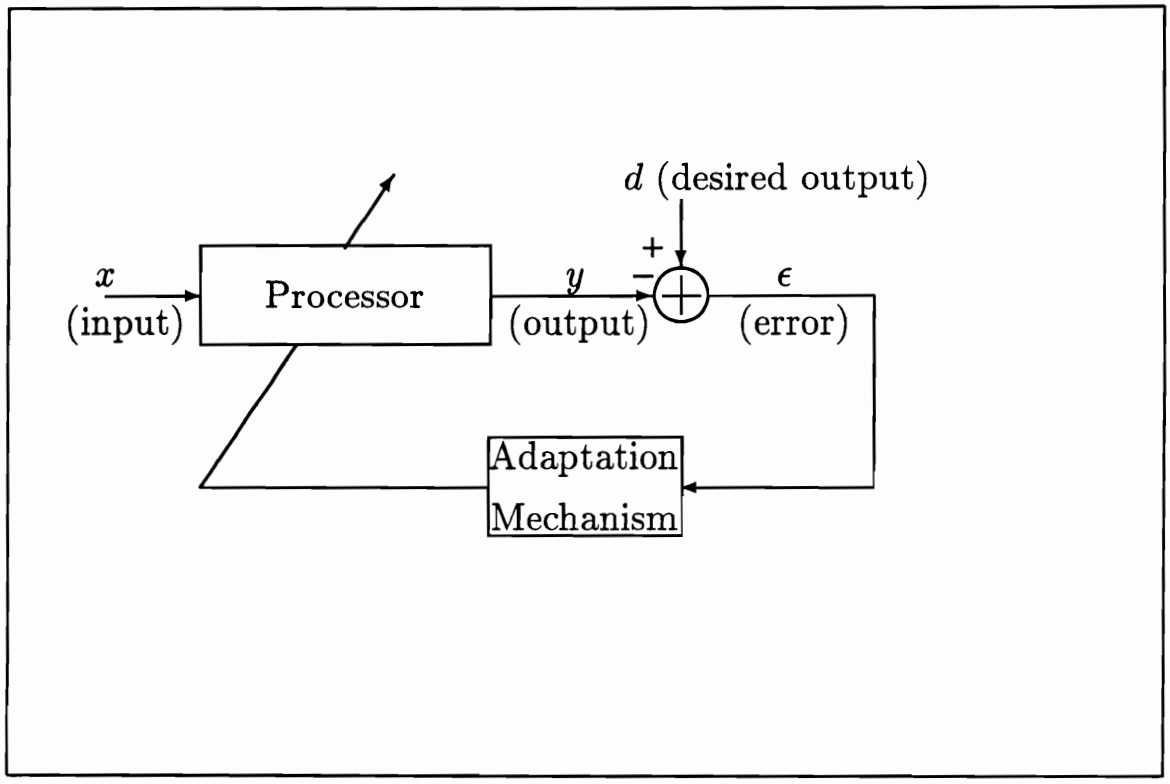


Figure 4.6: Adaptive Processor in Closed-Loop Configuration

error signal. This is different from conventional feedback and can be called *performance feedback.*

In this section, we derive the LMS algorithm for the adaptive linear combiner which is the most fundamental component in adaptive signal processing. Also, application of the *performance feedback* scheme in control systems is reviewed.

4.5.1 The LMS Algorithm for Adaptive Linear Combiner

The adaptive linear combiner is a multiple-input filter of which the output signal is a linear combination of the input signals with adjustable weights. However, if a single input and its delayed versions are combined linearly with adjustable weights, we have a special form of adaptive linear combiner — the single-input transversal filter as shown in Figure 4.7. It is equivalent to a FIR filter with adjustable coefficients. The input-output relationship can be described by

$$y_k = \sum_{l=0}^{L} w_{lk}\, x_{k-l}\,, \tag{4.5.1}$$

and in matrix form by

$$y_k = X_k^{\mathrm{T}}\, W_k = W_k^{\mathrm{T}}\, X_k\,, \tag{4.5.2}$$

where $W_k = [w_{0k} \;\; w_{1k} \;\; \cdots \;\; w_{Lk}]^{\mathrm{T}}$.

Assuming that in Figure 4.6 the adaptive processor is the adaptive transversal filter, the error signal at the time instant k becomes

$$\varepsilon_k = d_k - y_k = d_k - X_k^{\mathrm{T}}\, W_k\,. \tag{4.5.3}$$

The LMS algorithm is derived based upon minimization of the square of the error signal ε_k^2, which is taken to be an estimate of the expected value $\mathbb{E}[\varepsilon_k^2]$. The estimated gradient will be

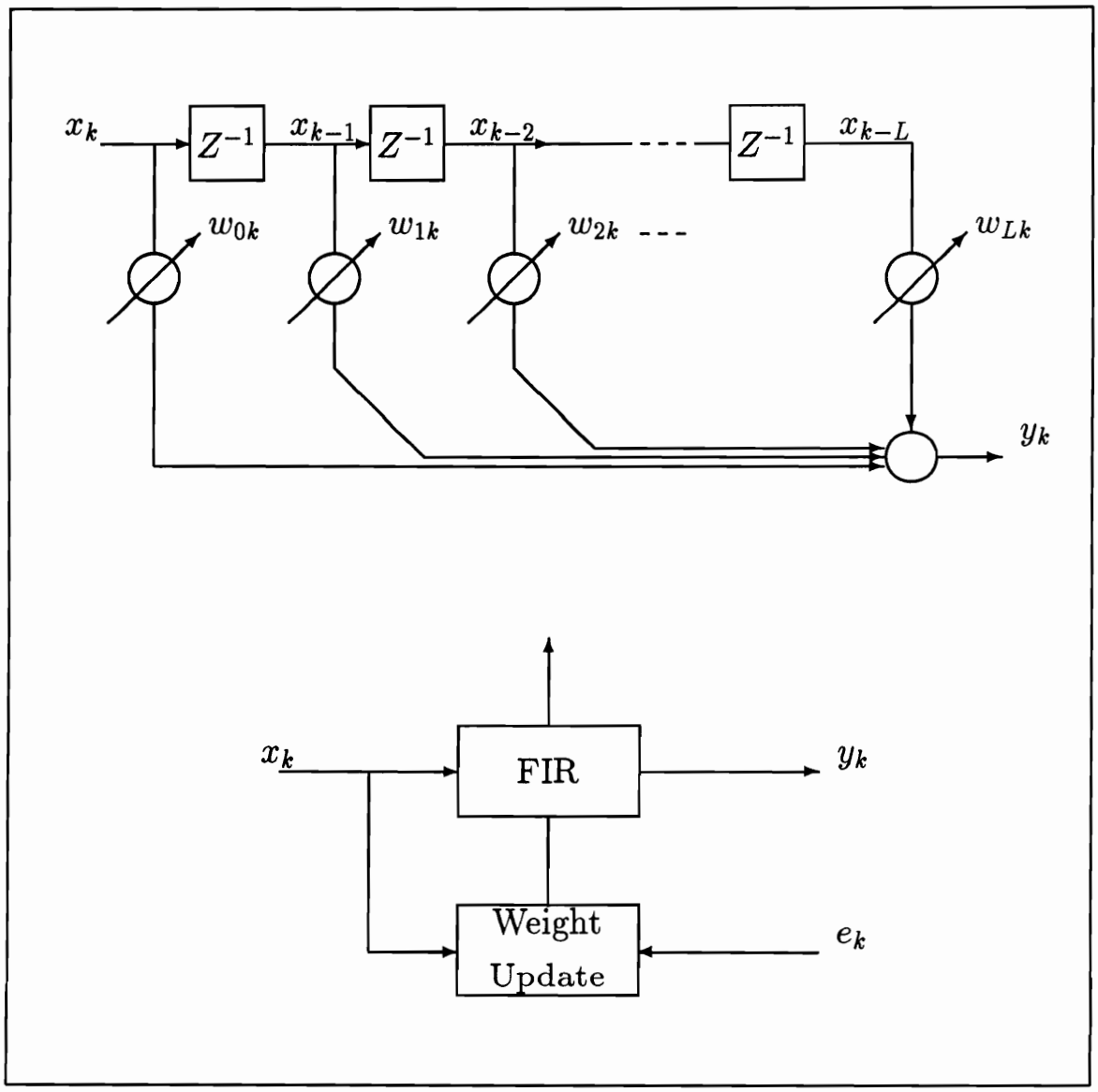


Figure 4.7: Adaptive Linear Combiner

$$\hat{\nabla}_k = \frac{\partial \varepsilon_k^2}{\partial W_k} = \begin{bmatrix} \frac{\partial \varepsilon_k^2}{\partial w_0} \\ \vdots \\ \frac{\partial \varepsilon_k^2}{\partial w_L} \end{bmatrix} = 2\varepsilon_k \begin{bmatrix} \frac{\partial \varepsilon_k}{\partial w_0} \\ \vdots \\ \frac{\partial \varepsilon_k}{\partial w_L} \end{bmatrix} = -2\ \varepsilon_k X_k\ , \tag{4.5.4}$$

and the weights can be adjusted in the negative estimated gradient direction as

$$W_{k+1} = W_k - \mu\ \hat{\nabla}_k = W_k\ +\ 2\ \mu\ \varepsilon_k\ X_k\ , \tag{4.5.5}$$

where μ is the gain constant that regulates the speed and stability of adaptation.

The LMS algorithm in Eq.(4.5.5) has the same form as that derived from stochastic approximation in Section 3.8 .

4.5.2 Adaptive Control Application

As described earlier, the LMS adaptive filter algorithm was developed for use in adaptive signal processing. It is the development of the filtered-X algorithm by Widrow[37] that starts applications on adaptive control. The filtered-X algorithm is useful when there are transfer functions in the auxiliary path following the adaptive filter. It was shown that when a transfer function follows the adaptive filter, it must also be placed in the input to the weight update process or error correlators[37]. Thus, the input signal X_k in Eq.(4.5.5) will be changed to a filtered version of X_k. This algorithm has been successfully applied in the field of adaptive active noise control[38] structural vibration control[39]. The noise or a signal that is coherent to the disturbance is passed through an adaptive filter and the control signal is obtained as the output of the filter whose coefficients are updated such that a quadratic cost function based upon a measurable variable, usually an error signal, of the system is minimized. The filtered-X algorithm is required since there is a transfer function along the auxiliary path between the output of the filter(control signal) and the error signal. While the adaptive filter in the filtered-X algorithm has an FIR filter structure, which implements an all-zero function, Eriksson[40] developed an algorithm called the filtered-U algorithm in which the adaptive filter has an infinite impulse filter structure that can represent both poles and zeros directly.

Using the LMS algorithm, the feedforward adaptive control method is very attractive because the update adaptive algorithm is quite simple and computationally economic. However, the noise or the coherent disturbance signal may not be available in many control problems and then the method is not applicable.

5. STR Design: Unknown System Parameters

With the advantages of the increased flexibility of control programs and the decision-making logic capability of digital systems, computer control of physical systems is becoming more and more common[41]. Instead of being designed using a continuous-time model with physically meaningful parameters, a digital controller is designed based upon a ZOH discrete-time model to which a continuous-time model is transformed. Usually, after transformation, a sparse continuous-time model matrix results in a non-sparse discrete-time model matrix, the parameters of which bear little physical meaning. Uncertainty in a small number of parameters in a continuous-time model gets smeared over the entire corresponding discrete-time model. While a controller designed by the method of linear optimal control theory with a Kalman filter state estimator is rather sensitive to the uncertainty and variation of parameters, the stability may be most sensitive only to the errors in a small number of parameters. This chapter focuses on the design of a self-tuning type digital controller where the most sensitive system parameters of the continuous-time model are identified recursively. Design of the self-tuning regulator based upon one most sensitive parameter, which can be extended to two or more parameters, in a continuous-time model is presented. The example of rejecting a disturbance in a simply-supported plate is given. Both simulated and experimental results are provided. Performance of the adaptive controller and the nonadaptive robust controller are compared. Also, the possibility of applying this technique to design controllers that self-tune the parameters in a disturbance model is explored.

5.1 Continuous-Time Model Uncertainty

For a flexible structure modeled by n of its vibration modes, the state equations for all the n modes are combined to form the model in state-space form

$$\dot{x} = F\,x + G\,u_p + L\,d\,, \tag{5.1.1}$$

where $x \in R^{2n}$ is the state of the system, $u_p \in R$ is the control input and $d \in R$ is the disturbance. One way to combine all the state equations results in a model in which

$$F = \begin{bmatrix} & 0_{n\times n} & & & I_{n\times n} & \\ -\omega_1^2 & & & -2\zeta_1\omega_1 & & \\ & -\omega_2^2 & & & -2\zeta_2\omega_2 & \\ & & \ddots & & & \ddots \\ & & & -\omega_n^2 & & & -2\zeta_n\omega_n \end{bmatrix} \tag{5.1.2}$$

$$G = \begin{bmatrix} 0_{1\times n} & g_1 & g_2 & \cdots & g_n \end{bmatrix}^{\mathrm{T}} \tag{5.1.3}$$

$$L = \begin{bmatrix} 0_{1\times n} & l_1 & l_2 & \cdots & l_n \end{bmatrix}^{\mathrm{T}}, \tag{5.1.4}$$

where ω_i and ζ_i are the natural frequency and damping coefficient of the ith mode and g_i and l_i are determined by the locations where the control signal and the disturbance enter the system.

Assuming that the controlled outputs are the modal accelerations, the output equation is given by

$$y = C\,x + D\,u_p + E\,d + \theta\,, \tag{5.1.5}$$

in which

$$C = \begin{bmatrix} -\omega_1^2 & & & -2\zeta_1\omega_1 & & \\ & -\omega_2^2 & & & -2\zeta_2\omega_2 & \\ & & \ddots & & & \ddots \\ & & & -\omega_n^2 & & & -2\zeta_n\omega_n \end{bmatrix} \tag{5.1.6}$$

$$D = \begin{bmatrix} g_1 & g_2 & \cdots & g_n \end{bmatrix}^{\mathrm{T}} \tag{5.1.7}$$

$$E = \begin{bmatrix} l_1 & l_2 & \cdots & l_n \end{bmatrix}^{\mathrm{T}}, \tag{5.1.8}$$

and θ is a Gaussian measurement noise vector.

The LQG controller may be very sensitive to only a small number of parameters in the model. It was shown in Chapter 2 that the stability of the LQG regulator is most sensitive to errors in the natural frequency of the second mode in the case of rejecting a disturbance in a simply-supported plate.

Here, we assume that the natural frequency of the jth mode is the most sensitive parameter in the model, call the nominal frequency ω_{jo} and the variation of the nominal frequency from the actual natural frequency of the mode is $\Delta\omega_j$. Thus, we have

$$\omega_j = w_{jo} + \Delta\omega_j, \tag{5.1.9}$$

where ω_j is the actual natural frequency of the jth mode. A self-tuning algorithm is developed based upon the assumuption that all the other parameters are correct. The self-tuning mechanism is to estimate the frequency variation and actual natural frequency such that the state estimator and the state feedback gains are based upon an increasingly accurate model.

5.2 Self-Tuning Formulation

5.2.1 State and Parameter Estimation

In the LQG control framework, the models of the structure, the smoothing filter and the shaping filter are combined to form an augmented model based upon which the controller is designed. Under the assumption that the stability is most sensitive to the natural frequency of the jth mode, ω_j, the augmented system matrix, F_a, is taken to be a function of ω_j, $F_a(\omega_j)$, and the discrete-time system matrix can be written by

$$\begin{aligned} dF_a(\omega_j) &= e^{F_a(\omega_j)T} \\ &= \sum_{n=0}^{\infty} \frac{(F_a(\omega_j)\,T)^n}{\mathrm{n}\,!}, \end{aligned} \tag{5.2.1}$$

where T is sampling period and the discrete-time augmented control influence matrix becomes

$$dG_a(\omega_j) = \int_0^T \sum_{n=0}^{\infty} \frac{(F_a(\omega_j)\, T)^n}{n\,!}\, G_a\, dt\ . \tag{5.2.2}$$

The jth mode natural frequency ω_j appears nonlinearly in the model because of the entry $-\omega_j^2$. Nonlinearity is also introduced when the continuous-time model is transformed into a discrete-time model. These make the nonlinear formulation of the problem obvious. The nonlinear relation between the natural frequency and the state to be estimated forces us to make use of an extended Kalman filter as will be shown later.

By approximating dF_a with the first two terms of the Taylor's series expansion of $e^{F_a(\omega_j)T}$ around ω_{jo} , we have

$$dF_a = e^{F_a(\omega_{jo})T} + \Delta\omega_j \frac{d}{d\omega_j}\left(e^{F_a(\omega_j)\, T}\right)\Big|_{\omega_j = \omega_{jo}} \tag{5.2.3}$$

$$dG_a = \int_0^T e^{F_a(\omega_{jo})t}\, dt\, G_a \;+\; \Delta\omega_j \int_0^T \frac{d}{d\omega_j}\left(e^{F_a(\omega_j)t}\right)\Big|_{\omega_j = \omega_{jo}} dt\, G_a\ . \tag{5.2.4}$$

Now, since

$$\frac{d}{d\omega_j}\left(e^{F_a(\omega_j)T}\right) = \sum_{m=0}^{\infty} \sum_{n=0}^{m} F_a^n\, \frac{dF_a(\omega_j)}{d\omega_j}\, F_a^{(m-n)}\, \frac{T^{(m+1)}}{(j+1)!} \tag{5.2.5}$$

can be evaluated at $\omega_j = \omega_{jo}$ using the fact that

$$\frac{dF_a(\omega_j)}{d\omega_j} = \begin{bmatrix} & 0_{n\times n} & & & 0_{n\times n} & \\ 0 & & & 0 & & \\ & \ddots & & & \ddots & 0 \\ & -2\omega_j & & & -2\zeta_j & \\ & & 0 & & & 0 \\ & & & \ddots & & & \ddots \\ & & & & 0 & & & 0 \end{bmatrix}, \quad (5.2.6)$$

the discrete-time model Eq.(2.2.1) is approximated by

$$x_{a,k+1} = dF_a(\omega_{jo})\, x_{a,k} + \Delta\omega_j\, \Delta F_a\, x_{a,k} + dG_a(\omega_{jo})\, u_k + \Delta\omega_j\, \Delta G_a\, u_k + v_k \quad (5.2.7)$$

where

$$\Delta F_a = \left.\frac{d}{d\omega_j}\left(e^{F_a(\omega_j)T}\right)\right|_{\omega_j = \omega_{jo}} \quad (5.2.8)$$

$$\Delta G_a = \int_0^T \left.\frac{d}{d\omega_j}\left(e^{F_a(\omega_j)t}\right)\right|_{\omega_j = \omega_{jo}} dt\, G_a \quad (5.2.9)$$

and

$$v_k = N(0, Q_d) \quad (5.2.10)$$

$$Q_d \simeq \int_0^T e^{F_a(\omega_{jo})t}\, L_a\, Q_c\, L_a^{\mathrm{T}} \left[e^{F_a(\omega_{jo})t}\right]^{\mathrm{T}} dt\,. \quad (5.2.11)$$

Treating $\Delta\omega_j$ as a state variable to be estimated, we have from Eq. (5.2.7) the equation

$$\begin{bmatrix} x_{a,k+1} \\ \Delta\omega_{j,k+1} \end{bmatrix} = \begin{bmatrix} dF_a(\omega_{jo}) & \Delta F_a\, x_{a,k} + \Delta G_a\, u_k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{a,k} \\ \Delta\omega_{j,k} \end{bmatrix}$$

$$+\begin{bmatrix} dG_a(\omega_{jo}) \\ 0 \end{bmatrix} u_k + \tilde{v}_k \ , \tag{5.2.12}$$

where

$$\tilde{v}_k = N(0, Q_a) \quad \text{and} \quad Q_a = \begin{bmatrix} Q_d & 0 \\ 0 & 0 \end{bmatrix}.$$

Since state variables appear in the state coefficient matrix, the formulation results in a nonlinear state equation. The extended Kalman filter can be used to obtain an approximate linear model.

Assuming that $\hat{x}_{a,\,k/k}$ and $\widehat{\Delta\omega}_{j,\,k}$ are the estimates of the state and the $\Delta\omega_j$ respectively at the time instant k based on the information up to time instant k, the extended Kalman Filter technique suggests that the equation be linearized at $\hat{x}_{a,\,k/k}$ and $\widehat{\Delta\omega}_{j,\,k/k}$. The state model becomes

$$\begin{bmatrix} x_{a,\,k+1} \\ \Delta\omega_{j,\,k+1} \end{bmatrix} = \begin{bmatrix} dF_a(\omega_{jo}) + \Delta F_a \widehat{\Delta\omega}_{j,\,k/k} & \Delta F_a\, \hat{x}_{a,\,k/k} + \Delta G_a\, u_k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{a,\,k} \\ \Delta\omega_{j,\,k} \end{bmatrix}$$

$$+\begin{bmatrix} dG_a(\omega_{jo}) \\ 0 \end{bmatrix} u_k + \tilde{v}_k + \begin{bmatrix} -\Delta F_a \hat{x}_{a,\,k/k} \widehat{\Delta\omega}_{j,\,k/k} \\ 0 \end{bmatrix}$$

$$\triangleq\ dF_{as} \begin{bmatrix} x_{a,\,k} \\ \Delta\omega_{j,\,k} \end{bmatrix} + dG_{as}\, u_k + \tilde{v}_k + dE_{as} \ . \tag{5.2.13}$$

The C_a matrix in the augmented output equation, a function of the unknown

$\Delta\omega_j$, can be written as $C_a(\Delta\omega_j)$. Linearized at $\hat{x}_{a,k/k-1}$ and $\widehat{\Delta\omega}_{j,k/k-1}$, the output equation becomes

$$y_{a,k} = \begin{bmatrix} & & 0 \\ & & \vdots \\ C_a(\widehat{\Delta\omega}_{j,k/k-1}) & -2(\omega_{jo}+\widehat{\Delta\omega}_{j,k/k-1})\hat{x}_{a,j} - 2\zeta_j\,\hat{x}_{a,j+n} & \\ & & 0 \\ & & \vdots \\ & & 0 \end{bmatrix} \begin{bmatrix} x_{a,k} \\ \Delta\omega_{j,k} \end{bmatrix} -$$

$$\begin{bmatrix} 0 \\ \vdots \\ -2(\omega_{jo}+\widehat{\Delta\omega}_{j,k/k-1})\,\hat{x}_{a,j} - 2\zeta_j\,\hat{x}_{a,j+n} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \widehat{\Delta\omega}_{j,k/k-1} + \theta_k$$

$$\triangleq\ dC_{as} \begin{bmatrix} x_{a,k} \\ \Delta\omega_{j,k} \end{bmatrix} + dD_{as} + \theta_k\,. \tag{5.2.14}$$

Based upon the above equations, estimates of the term $\Delta\omega_j$ together with the state of the system can be obtained by the Kalman filter which is also implemented as a one-step-ahead predictor. With the state error covariance propagation and filter gain update, the algorithm is listed as

$$P_{k/k-1} = dF_{as,k-1}\,P_{k-1/k-1}\,dF_{as,k-1}^{\mathrm{T}} + Q_a \tag{5.2.15}$$

$$K_k = P_{k/k-1}\,dC_{as,k}^{\mathrm{T}}(dC_{as,k}\,P_{k/k-1}\,dC_{as,k}^{\mathrm{T}} + R_a)^{-1} \tag{5.2.16}$$

$$\hat{x}_{s,k/k-1} = dF_{as,k-1}\,\hat{x}_{s,k-1/k-1} + dG_{as}\,u_{k-1} + dE_{as} \tag{5.2.17}$$

$$\hat{y}_{a,k} = dC_{as,k}\,\hat{x}_{s,k/k-1} + dD_{as} \tag{5.2.18}$$

$$\widehat{x}_{s,k/k} = \widehat{x}_{s,k/k-1} + K_k[y_a - \widehat{y}_{a,k}] \tag{5.2.19}$$

$$P_{k/k} = (I - K_k\, dC_{as,k})\, P_{k/k-1}\,, \tag{5.2.20}$$

where

$$Q_a = \mathbb{E}\,[\,\widetilde{v_k}\;\widetilde{v_k}^{\mathrm{T}}]$$

$$\widehat{x}_{s,k} = \begin{bmatrix} \widehat{x}_{a,k} \\ \widehat{\Delta\omega}_{j,k} \end{bmatrix},$$

P is the error covariance, and K is the Kalman filter gain.

5.2.2 Feedback Gains Calculation

Applying the certainty equivalent principle, the optimal feedback gains can be designed based upon the estimated parameters of the plant as if they were correct. The feedback control law has the form[42]

$$u_k = -K_{lq,k}\, x_{a,k} \tag{5.2.21}$$

where

$$K_{lq,k} = \frac{dG_{as1}^{\mathrm{T}}\, P_{lq,k}\, dF_{as11,k}}{R + dG_{as1}^{\mathrm{T}} P_{lq,k} dG_{as1}} \tag{5.2.22}$$

$$dF_{as} = \begin{bmatrix} dF_{as11} & dF_{as12} \\ 0 & 1 \end{bmatrix}$$

$$dG_{as1} = \begin{bmatrix} dG_{as1} \\ 0 \end{bmatrix},$$

and $P_{lq,k}$ is the solution of the discrete-time algebraic Riccati equation

$$P_{lq,k} = dF_{as11,k}^{\mathrm{T}}(P_{lq,k} - \frac{P_{lq,k}dG_{as1}dG_{as1}^{\mathrm{T}}P_{lq,k}}{R + dG_{as1}^{\mathrm{T}}P_{lq,k}dG_{as1}})dF_{as11,k} + Q \ . \tag{5.2.23}$$

Eq.(5.2.23) can be iteratively solved until a convergent $P_{lq,k}$ is obtained. However, only one iteration is done in one sample time in order to save computation time. Thus $P_{lq,k}$ is obtained approximately as

$$P_{lq,k} = dF_{as11,k}^{\mathrm{T}}(P_{lq,k-1} - \frac{P_{lq,k-1}dG_{as1}dG_{as1}^{\mathrm{T}}P_{lq,k-1}}{R + dG_{as1}^{\mathrm{T}}P_{lq,k-1}dG_{as1}})dF_{as11,k} + Q \ . \tag{5.2.24}$$

5.3 Self-Tuning Experiment

In this section, self-tuning control of a simply-supported plate is considered. The purpose of the controller is to reject a disturbance in the plate in the face of model uncertainty. The plate is modeled by the first two vibration modes in state space form. The natural frequency of the second mode ω_2 is assumed to be 6% off the identified value. It has been shown that the controller designed in Chapter 2 is most sensitive to this parameter variation and 6% variation is able to cause instability of the system.

The correct natural frequency of the 2nd mode ω_2 is 108.96 Hz and ω_{jo} is assumed to be 102.42 Hz, 6% lower than ω_2, so $\Delta\omega_2$ is equal to around 41.08 rad/sec. Since this variation of the parameter will cause instability, the self-tuning regulator must estimate the parameter while stabilizing the system.

A simulation of the adaptive control was performed assuming two vibration modes for the plate. Figure 5.1 is the simulated result for the estimation of $\Delta\omega_2$. The estimator was turned on in the beginning of the simulation at 0 second and

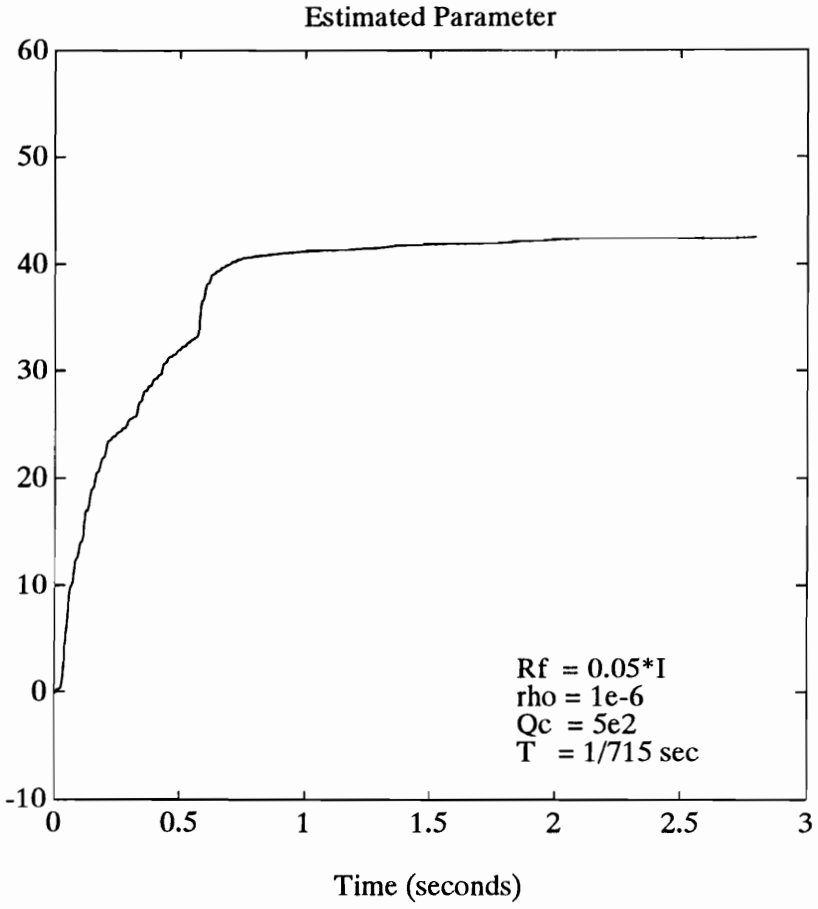


Figure 5.1: Estimation of $\Delta\omega_2$ (Simulated)

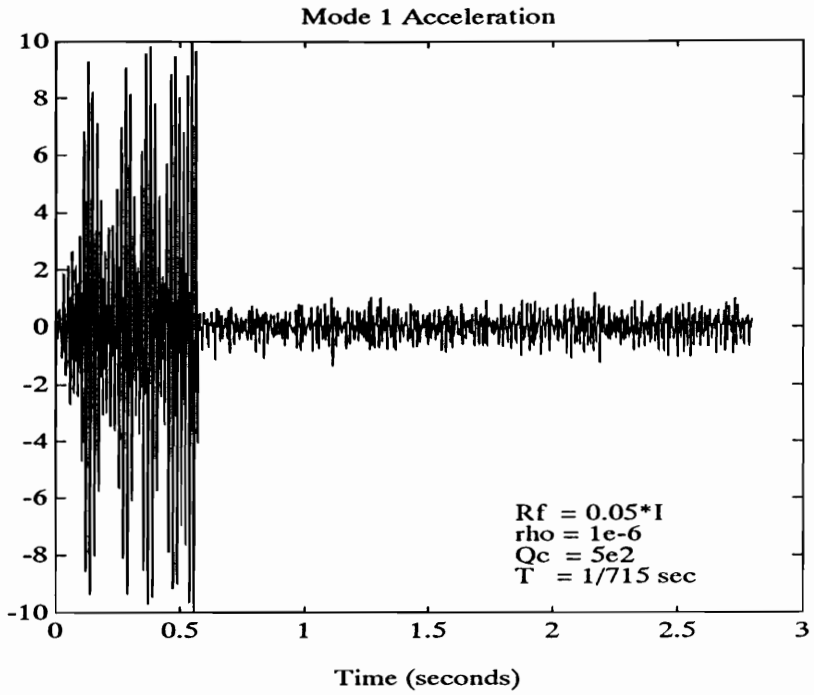
the controller was turned on at about 0.56 second. It shows that the estimate converges to the vicinity of $\Delta\omega_2$. The modal accelerations of the plate are plotted in Figure 5.2 which shows stable responses.

When the nonadaptive controller, designed based upon the incorrect model with 6% variation of 2nd mode natural frequency, is implemented on the real plate, the system becomes unstable. This is shown by the experimental results in Figure 5.3 in which after the controller is turned on at 0.56 second, the response of the 2nd mode goes out of bound. However, by implementing the self-tuning controller on the plate, the system is stabilized. The sampling rate can go as high as 715 Hz when the controller is implemented by the transputer system. The estimation of $\Delta\omega_2$, which converges to around the correct value, is plotted in Figure 5.4 and the stabilized responses of modal accelerations are plotted in Figure 5.5 .

It is important to know that in the experiment, we have no exact knowledge about the actual values of the most sensitive parameter and the other parameters of the plate, only a model. By assuming a 6% variaton on the most sensitive parameter in the model, the self-tuner is able to approximately estimate this variation.

In Figure 5.4, the control loop is closed at about 0.56 second and it is shown that the control signal contributes a lot to the convergence of the estimated parameter. This is because when the controller is turned on, 2nd mode tends to go unstable, resulting in an abrupt change of signal level in mode 2 acceleration before the self-tuning effect takes place to stabilize the system, and this richness of signal is critical to the fast convergence of the estimate. Similarly, convergence speed of open-loop estimation depends on the richness of both mode 1 and mode 2 accelerations. A comparison of the open-loop estimation in Figure 5.1 and Figure 5.4 shows that richness of signal of mode 1 acceleration in Figure 5.1 makes the convergence of the estimate faster.

Based upon the results of Chapter 2, the sampling rate of a non-adaptive fixed-gain 2-mode controller implemented on our transputer system is 4350 Hz. The maximum sampling rate of our 2-mode adaptive controller implemented on the same transputer system is 715 Hz. Roughly speaking, the non-adaptive controller can be sampled 6 times faster than the adaptive one. However, the state



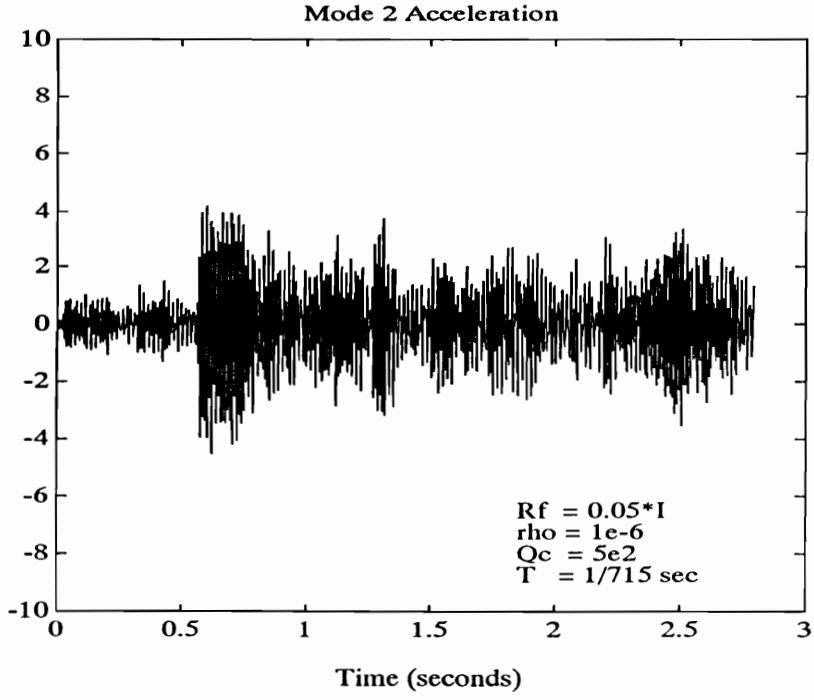
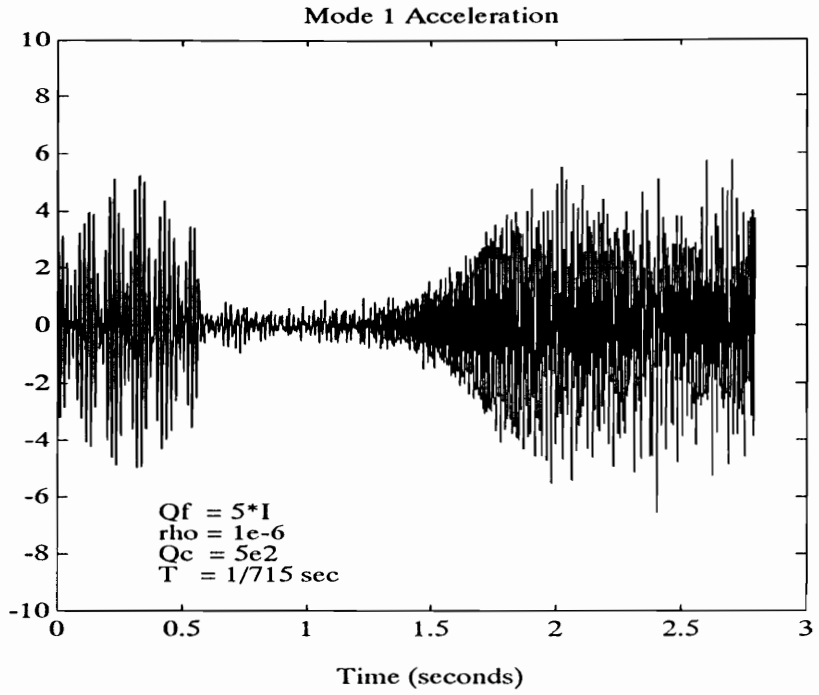


Figure 5.2: Self-Tuning LQG Narrowband Disturbance Rejection (Simulated)



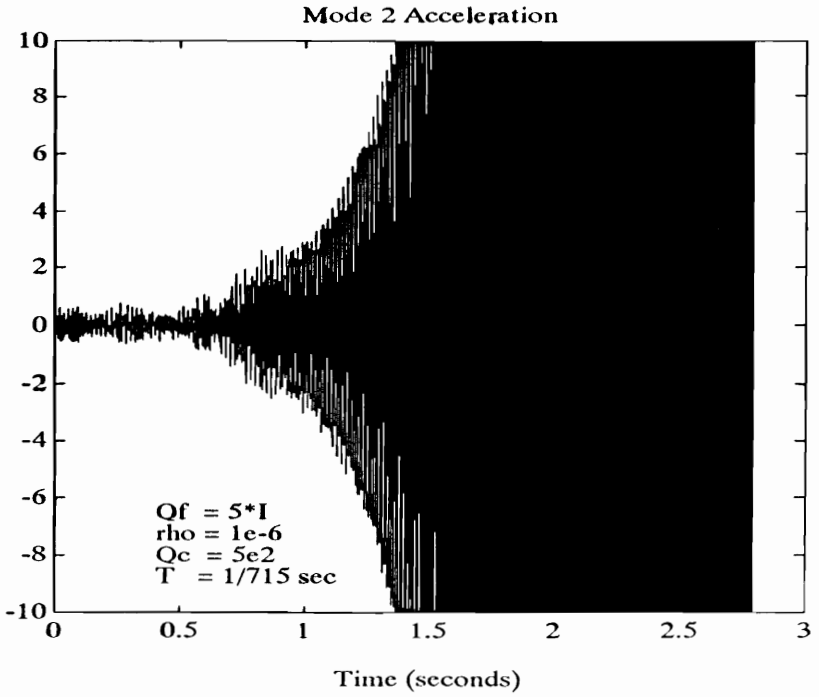


Figure 5.3: Unstable Response of LQG Control with Incorrect Model

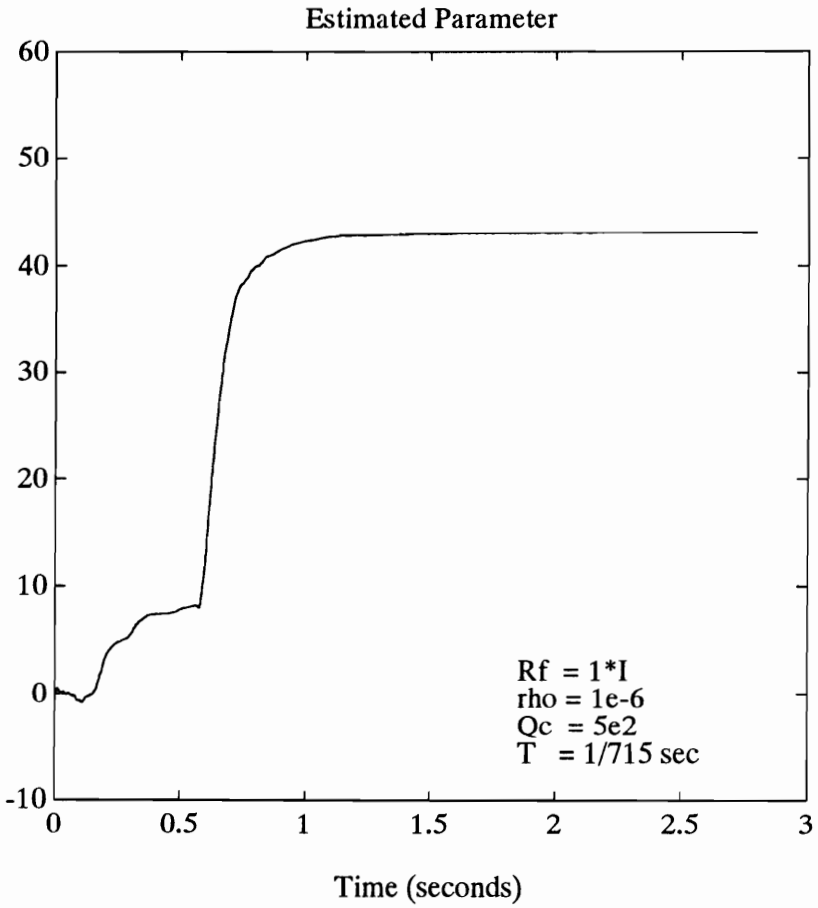
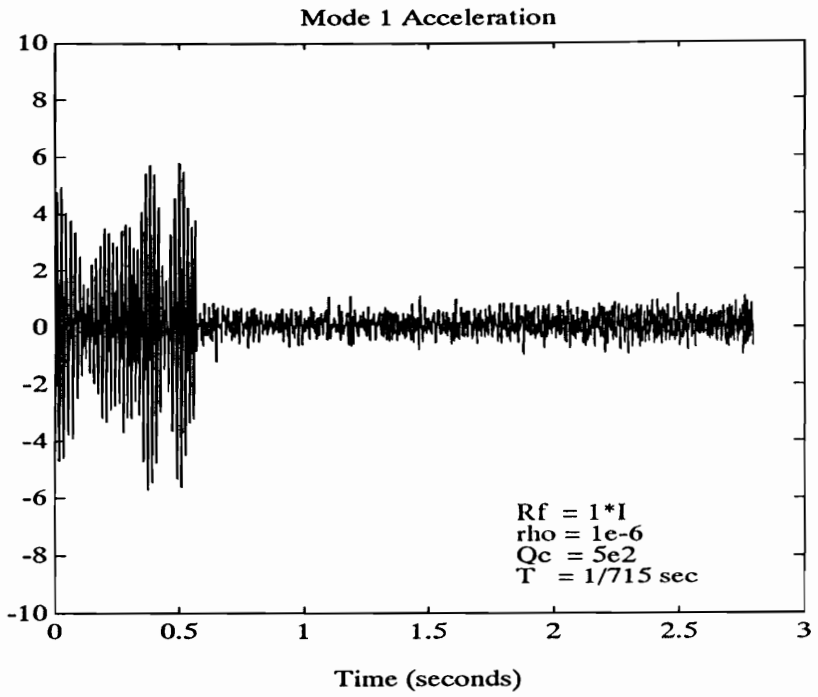


Figure 5.4: Estimation of $\Delta\omega_2$ (Experimental)



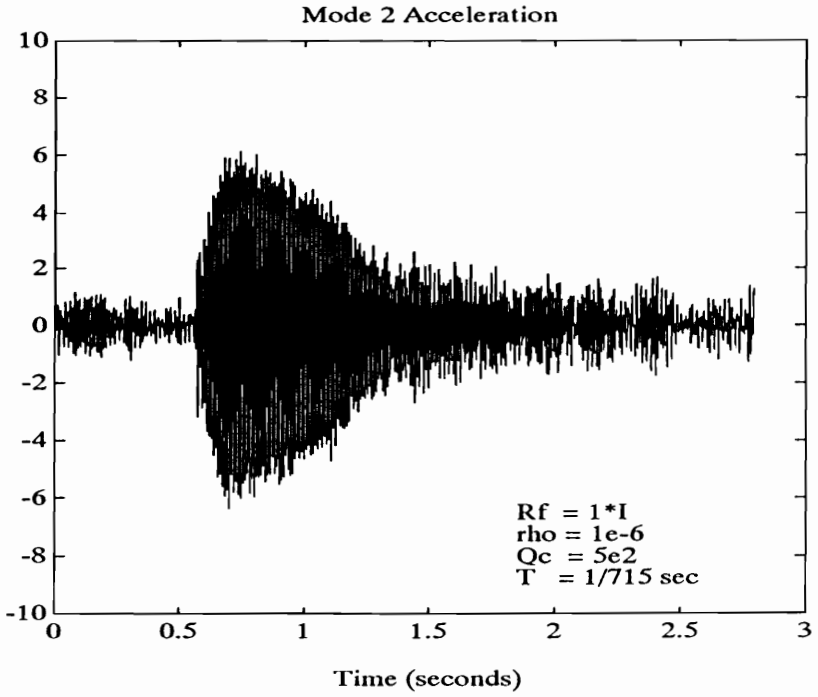


Figure 5.5: Self-Tuning LQG Narrowband Disturbance Rejection (Experimental)

feedback gains are not affected by the variation of the second mode natural frequency in our experiment and are not adaptively tuned. Were this not true, the penalty incurred by using adaptive control will be the reduction of the sampling rate to less than one tenth of the non-adaptive sampling rate.

5.4 Performance of the Self-Tuning Regulator

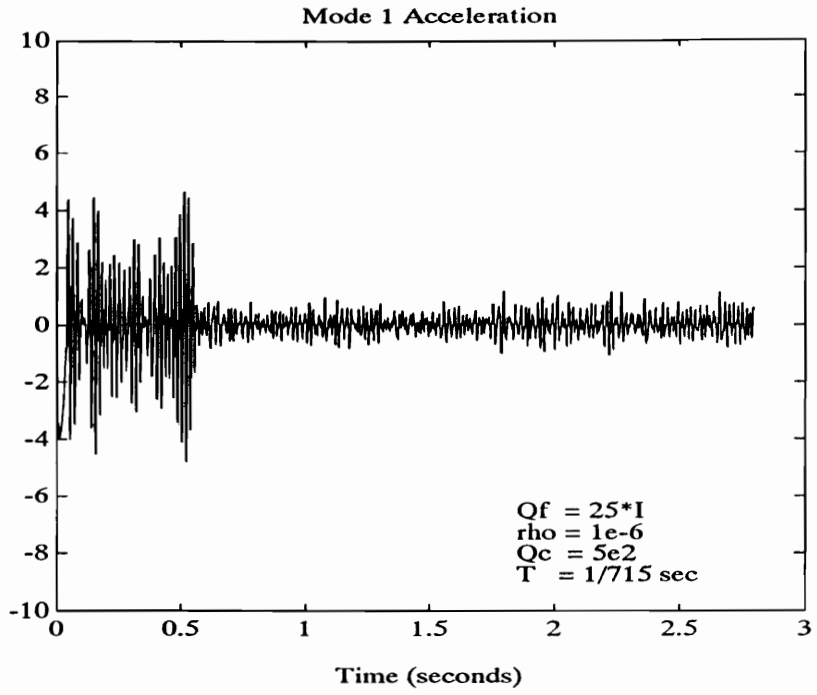
It has been shown that by assuming higher measurement noise intensity, the robustness of the system might be enhanced. The controller designed by assuming higher $Q_f(=25)$ indeed stabilizes the system as can be seen by the result shown in Figure 5.6 compared with Figure 5.3($Q_f = 5$) . The results presented in this section show that the self-tuning controller has better performance than the non-adaptive robust controller.

In Figure 5.5, when the controller is turned on at about 0.56 second and the estimated parameter is converging, the mode 1 acceleration is suppressed and most of the residual looks like high-frequency spillover into mode 1 due to imperfect modal filters. However, in Figure 5.6, most of the residual in mode 1 acceleration looks like true 60 Hz disturbance. Thus, although both the adaptive self-tuning controller and the non-adaptive robust controller have stable responses and suppress the vibration, the self-tuning controller rejects the 60 Hz narrow-band disturbance most effectively.

In the case of rejecting a pure 60 Hz disturbance by an adaptive controller, Figure 5.7 shows the result of parameter estimation and Figure 5.8 is the stabilized response of modal accelerations. Compared with the result of the robust controller($Q_f = 25$, incorrect model), shown in Figure 5.9, the self-tuning controller is shown to have better disturbance rejection performance in terms of amplitude of the response.

5.5 Unknown Disturbance Model

In LQG control, we usually assume that the disturbance can be modeled



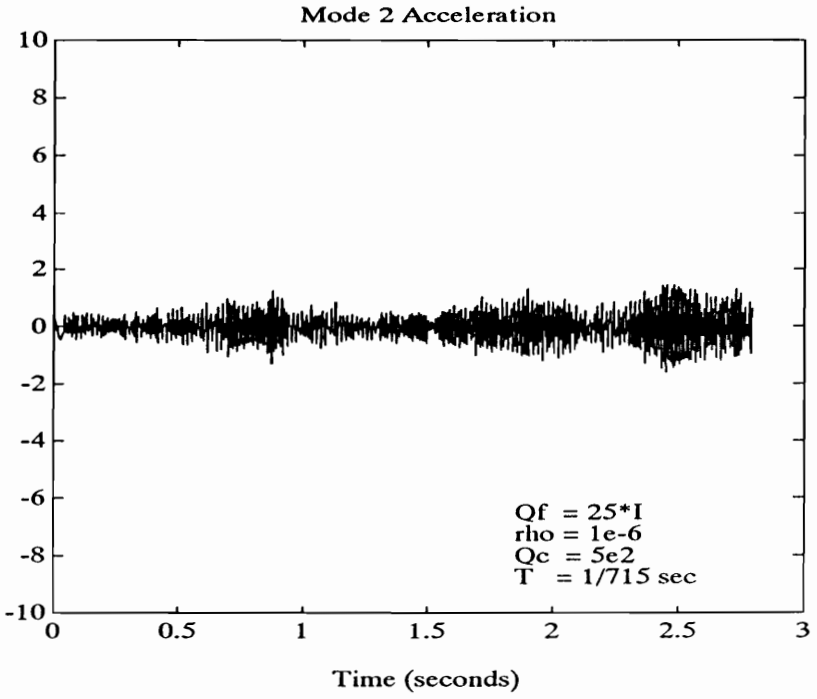


Figure 5.6: Robust LQG Narrowband Disturbance Rejection ($Q_f = 25$)

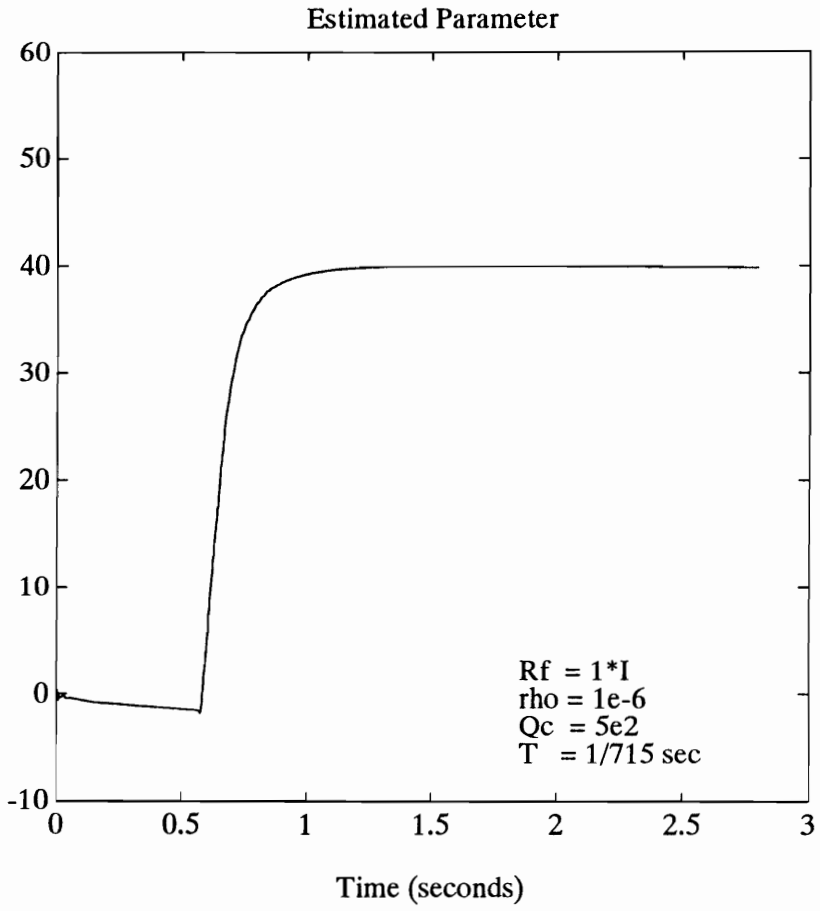
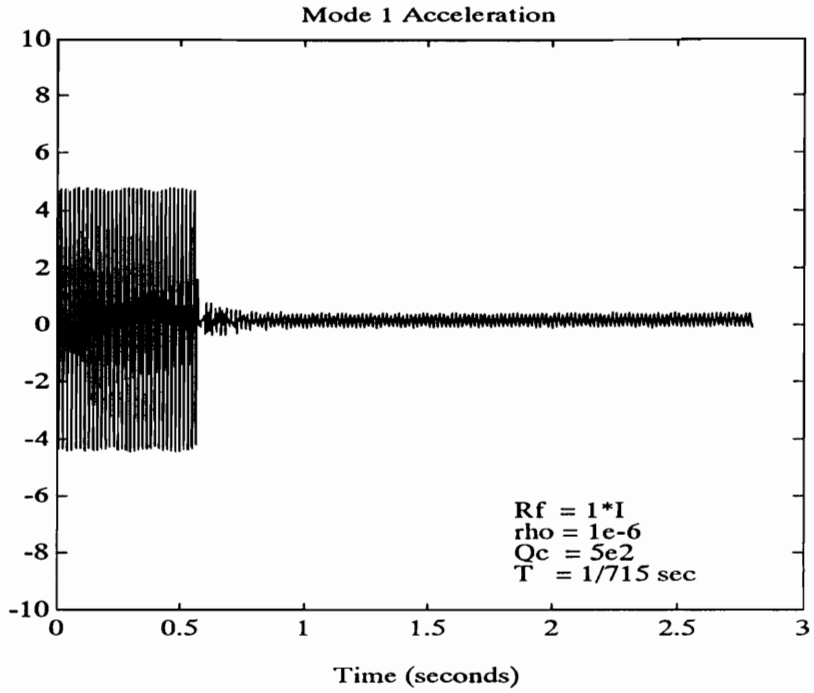


Figure 5.7: Estimation of $\Delta\omega_2$ (Experimental, 60 Hz)



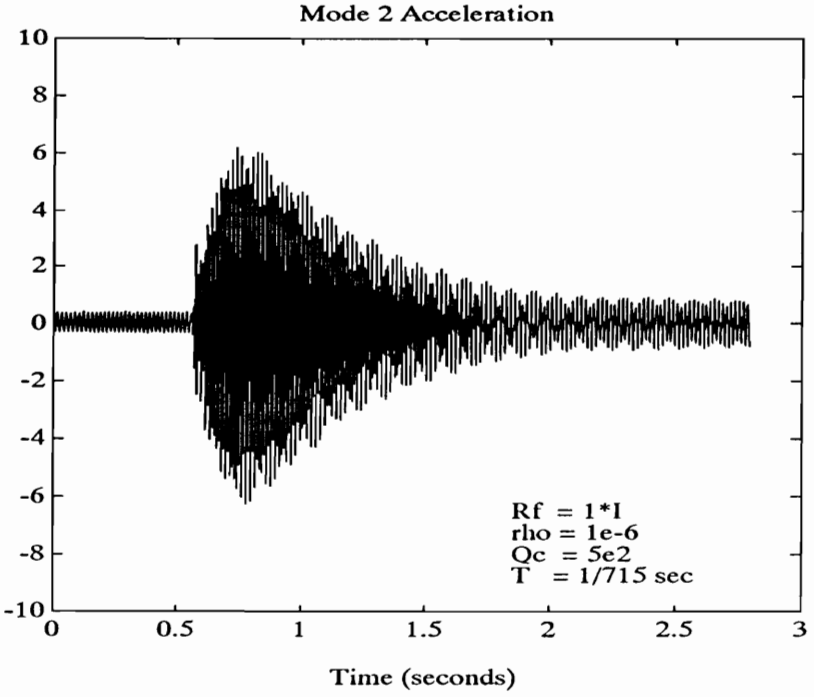
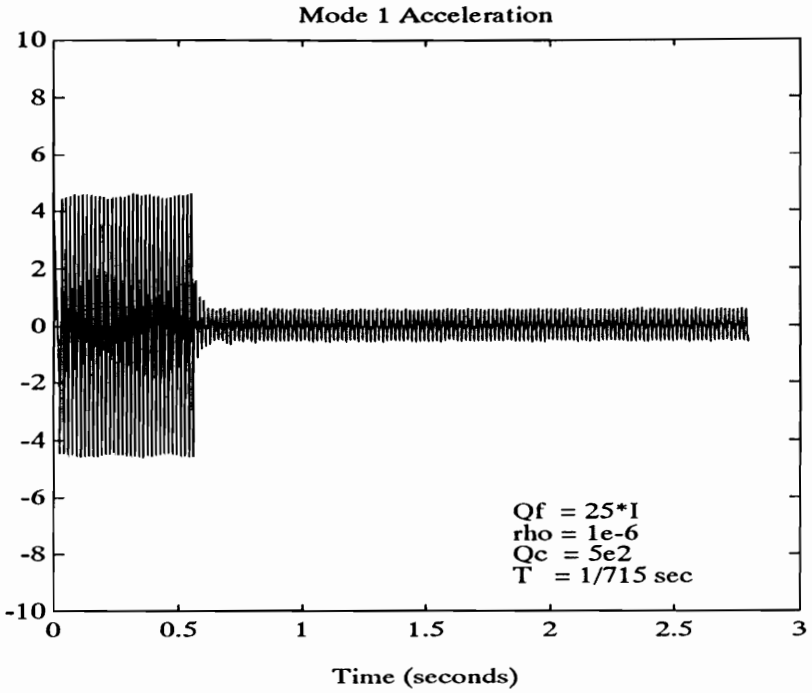


Figure 5.8: Self-Tuing LQG 60 Hz Disturbance Rejection (Experimental)



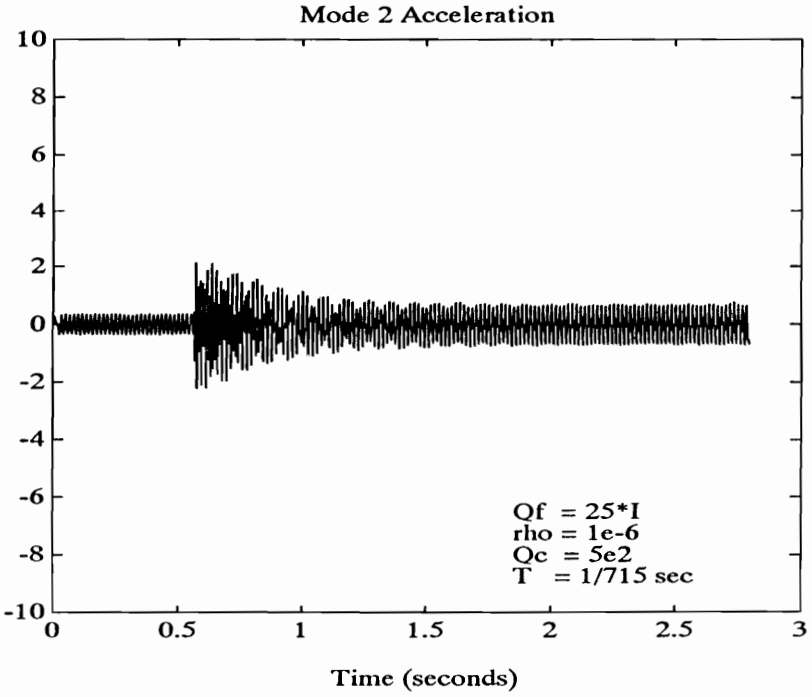


Figure 5.9: Robust LQG 60 Hz Disturbance Rejection ($Q_f = 25$)

correctly; however, this is not always the case. In this section, we explore the possibility of applying the technique described in this chapter to design controllers that self-tune the parameters in disturbance models.

5.5.1 Disturbance Model Uncertainty

The disturbance is modeled by the output of a shaping filter whose input is a zero mean Gaussian random signal. The shaping filter is described by

$$\dot{x} = F_w\, x \; + G_w\, v$$

$$d = C_w\, x \;,$$

where $v = N(0, Q_c)$ and d is the disturbance. For disturbance with narrow-band characteristics, we have

$$F_w = \begin{bmatrix} 0 & 1 \\ -\omega_w^2 & -2\zeta_w\omega_w \end{bmatrix}, \qquad G_w = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad C_w = \begin{bmatrix} 0 & 1 \end{bmatrix},$$

where ω_w is the center frequency of the pass-band and ζ_w is the damping coefficient. We assume that ω_w is incorrectly identified.

5.5.2 Self-Tuning Formulation

When the dynamics of the shaping filter are augmented to the combined model of the structure and the smoothing filter, we have the augmented model matrices F_a, G_a, L_a, C_a . Since ω_w can be treated as a variable in F_a in the same way as the natural frequencies of the structure, the whole design method in previous sections can be applied directly without any change. Although, as before, we combine the shaping filter model and the structure model in series, the whole system becomes uncontrollable but stabilizable if the shaping filter contains only stable modes, however, since the control signal does not affect the dynamics of the shaping filter, the control signal won't help to make the convergence of estimation any better.

5.5.3 Simulated Example

In this section we consider the simply-supported plate described in Chapter 2. The plate is subject to a narrow-band disturbance with pass-band center frequency equal to 60 Hz and bandwidth of 1 Hz. The center frequency is assumed to be incorrectly identified and 6% off the real value. Thus, $\omega_w = 376.8$ rad/sec and $\Delta\omega_w = 18.84$ rad/sec. Simulation is performed for two cases with disturbances of different intensities. It is shown that the estimate converges faster for higher intensity disturbance, which is richer than lower intensity disturbance, as can be seen in Figure 5.10 and Figure 5.11.

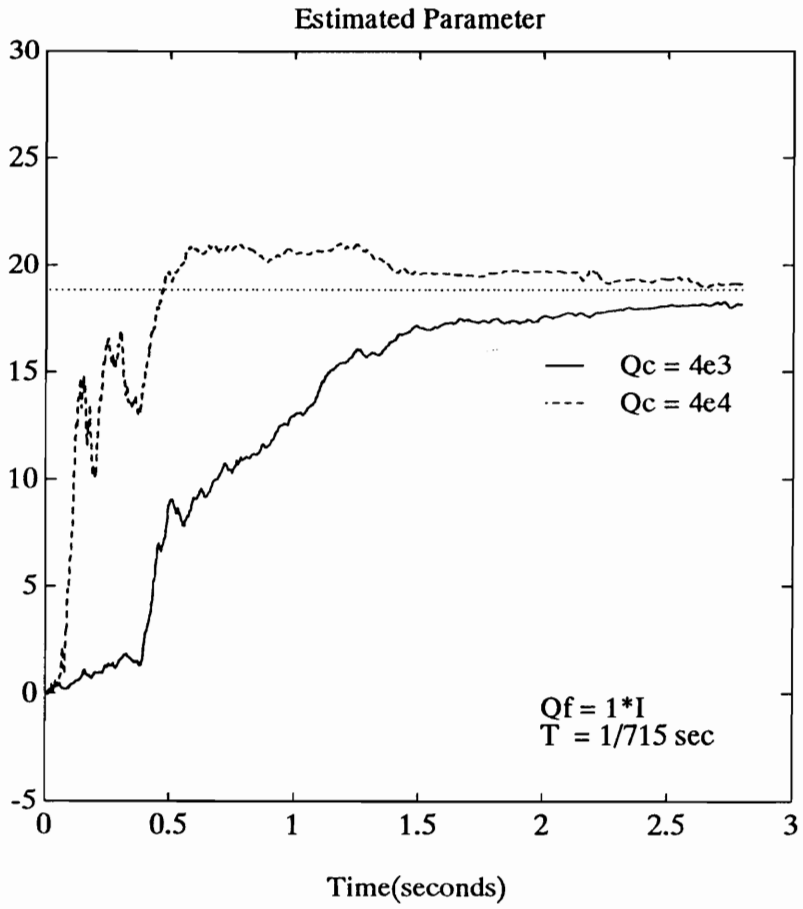
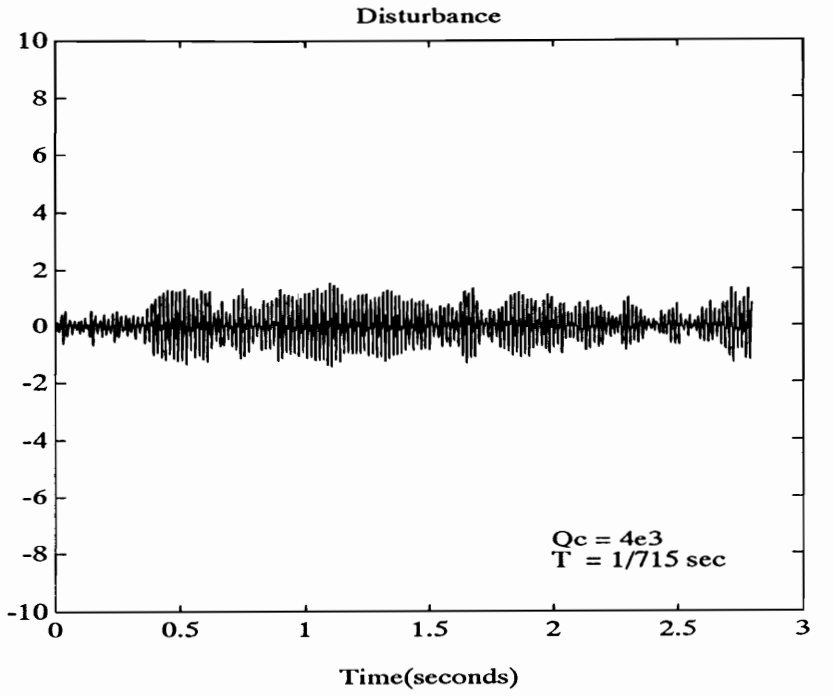


Figure 5.10: Estimation of $\Delta\omega_w$ in Disturbance Model



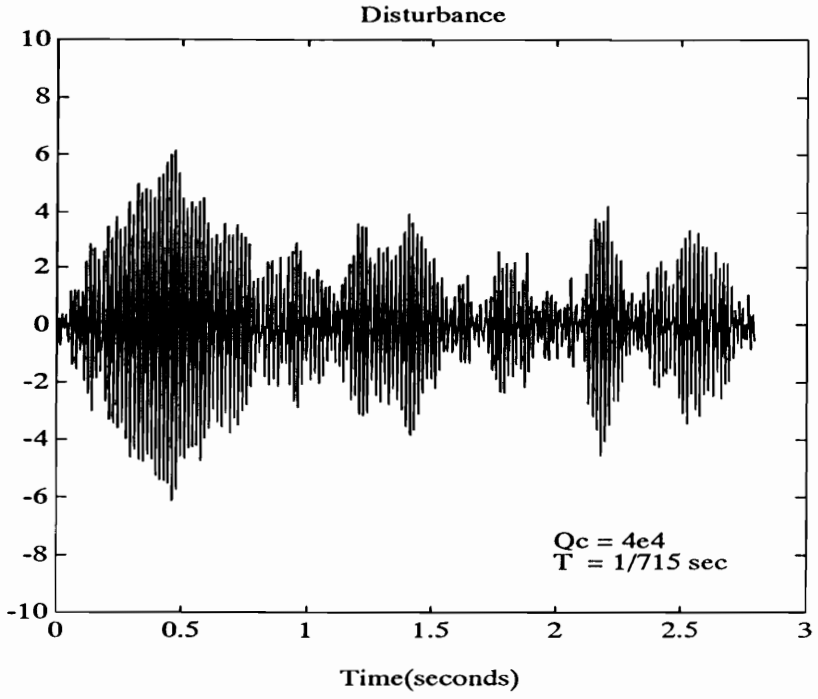


Figure 5.11 Disturbance of Different Intensities

6. STR Design: Simplified Computation Method

The need to propagate the state error covariance and to solve the control Riccati equation make LQG self-tuning regulators computationally intensive and are the major difficulties in implementing the controllers. Eq.(5.2.15) and Eq.(5.2.20), the state error covariance propagation and update equations, involve matrix multiplications, and Eq.(5.2.16), the Kalman gains update equation, needs a matrix inversion in addition to matrix multiplications. Compared to the fixed LQG controller described in Chapter 2, these equations represent a major portion of the extra calculations needed in one sampling period in order to obtain state and parameter estimates. The approximate numbers of operations involved in the extra calculations are listed in Table 6.1 for n modeled modes with only one parameter to be estimated and the order of the whole augmented system being equal to N. Multiplication of two $(N+1)\times(N+1)$ matrices, involved in Eq.(5.2.15) and Eq.(5.2.20), needs roughly $(N+1)^3$ floating-point number multiplications and is where the bulk of the computations are because the number of floating-point multiplications is proportional to the cube of the order of the system and the number of parameter together. We can not expect a linear increase in the number of computations when more modes or parameters are included.

This chapter is devoted to a method that will reduce the computing load of the self-tuning regulator proposed in Chapter 5 by approximating the state error covariance propagation steps.

6.1 Simplified Time-Varying Kalman Filter

In Chapter 2 the Kalman filter gains are obtained by solving the steady-state discrete algebraic Riccati equation for a steady-state error covariance matrix P. This can be done because the system is time-invariant and the error covariance matrix and the Kalman filter gain asymptotically reach their steady-

Table 6.1: Approximate Numbers of Operations of Extra Calculations

Operation	# of Operations	Equations Involved
Multiplication of two $(N+1)\times(N+1)$ matrices	3	(5.2.15) (5.2.20)
Multiplication of an $(N+1)\times(N+1)$ matrix and an $(N+1)\times n$ matrix	2	(5.2.16)
Multiplication of an $(N+1)\times n$ matrix and an $n\times n$ matrix	1	(5.2.16)
Multiplication of an $(N+1)\times n$ matrix and an $n\times(N+1)$ matrix	1	(5.2.20)
Addition of two $(N+1)\times(N+1)$ matrices	1	(5.2.15)
Addition of two $n\times n$ matrices	1	(5.2.16)
Inversion of an n $\times n$ matrix	1	(5.2.16)

state values. However, this is not true for the adaptive controller which is a time-variant system.

From Chapter 5, we define

$$dF_{as} = \left[\begin{array}{c|c} dF_a(\omega_{jo}) + \Delta F_a \widehat{\Delta\omega}_{j,k/k} & \Delta F_a \widehat{x}_{a,k/k} + \Delta G_a u_k \\ \hline 0 & 1 \end{array}\right]$$

$$= \left[\begin{array}{c|c} df^{(1)} & df^{(2)} \\ \hline 0 & 1 \end{array}\right] \tag{6.1.1}$$

$$dC_{as} = \left[\begin{array}{c|c} dc^{(1)} & dc^{(2)} \end{array}\right]. \tag{6.1.2}$$

If we partition the error covariance and Kalman filter gain according to dF_{as} , we end up with

$$P = \left[\begin{array}{c|c} p^{(1)} & p^{(2)} \\ \hline p^{(3)} & p^{(4)} \end{array}\right] \tag{6.1.3}$$

$$K = \left[\begin{array}{c} k^{(1)} \\ \hline k^{(2)} \end{array}\right] . \tag{6.1.4}$$

In the beginning of our effort to simplify the self-tuning algorithm, we tried to steady state the whole state error covariance matrix, choosing a small fixed number for $p^{(4)}$ and setting $p^{(2)}$ to zero. We found that a stable response can be obtained only for $p^{(4)}$ in a very small region, outside of which the system blew up. So, we chose to steady state only part of the state error covariance matrix.

It is found that the entries in $p^{(1)}$ and $p^{(4)}$ reach their steady-state values

monotonically, however, entries in $p^{(2)}$ do not, neither do the entries in $p^{(3)}$, which is the transpose of $p^{(2)}$. This can be seen from the simulation results of the adaptive controller in Figure 6.1, which shows the propagation of each entry of $p^{(2)}$, and in Figure 6.2, which shows the propagation of $p^{(4)}$, when the self-tuning controller succeeds in rejecting a narrow-band disturbance in a simply-supported plate. In those figures, the estimator is turned on in the beginning of the simulation at 0 second and the controller is turned on at about 0.56 second. $p^{(4)}$ converges monotonically to it's steady state while the entries of $p^{(2)}$ oscillate about their steady state values and converge asymptotically but not monotonically. It is the oscillations of those entries in $p^{(2)}$ that contribute to the successful estimation of the parameter and this is why a constant state error covariance matrix fails to give either good parameter estimation or stable response.

Based upon the assumption that the parameters have only small variations, we claim that the steady-state value of $p^{(1)}$ is approximately equal to the steady-state error covariance obtained from the incorrect model of the nonadaptive system. The only thing we know about the steady state value of $p^{(4)}$ is that it is a small number and there doesn't seem to be a systematic way of obtaining an approximation of it, so we chose to let it adapt itself. Thus, only $p^{(2)}$ and $p^{(4)}$ need to propagate and the Eqs.(2.2.16) and (2.2.21) can be written as

$$p^{(1)}_{k/k-1} = p^{(1)} + q^{(1)} \tag{6.1.5}$$

$$p^{(2)}_{k/k-1} = df^{(1)}_{k-1} p^{(2)}_{k-1/k-1} + df^{(2)}_{k-1} p^{(4)}_{k-1/k-1} + q^{(2)} \tag{6.1.6}$$

$$p^{(3)}_{k/k-1} = p^{(2)^{\mathrm{T}}}_{k-1/k-1} df^{(1)^{\mathrm{T}}}_{k-1} + p^{(4)^{\mathrm{T}}}_{k-1/k-1} df^{(2)^{\mathrm{T}}}_{k-1} + q^{(3)} \tag{6.1.7}$$

$$p^{(4)}_{k/k-1} = p^{(4)}_{k-1/k-1} + q^{(4)} \tag{6.1.8}$$

$$p^{(1)}_{k/k} = p^{(1)} \tag{6.1.9}$$

$$p^{(2)}_{k/k} = p^{(2)}_{k/k-1} k^{(1)}_k dc^{(1)} p^{(2)}_{k/k-1} - k^{(1)}_k dc^{(2)} p^{(4)}_{k/k-1} \tag{6.1.10}$$

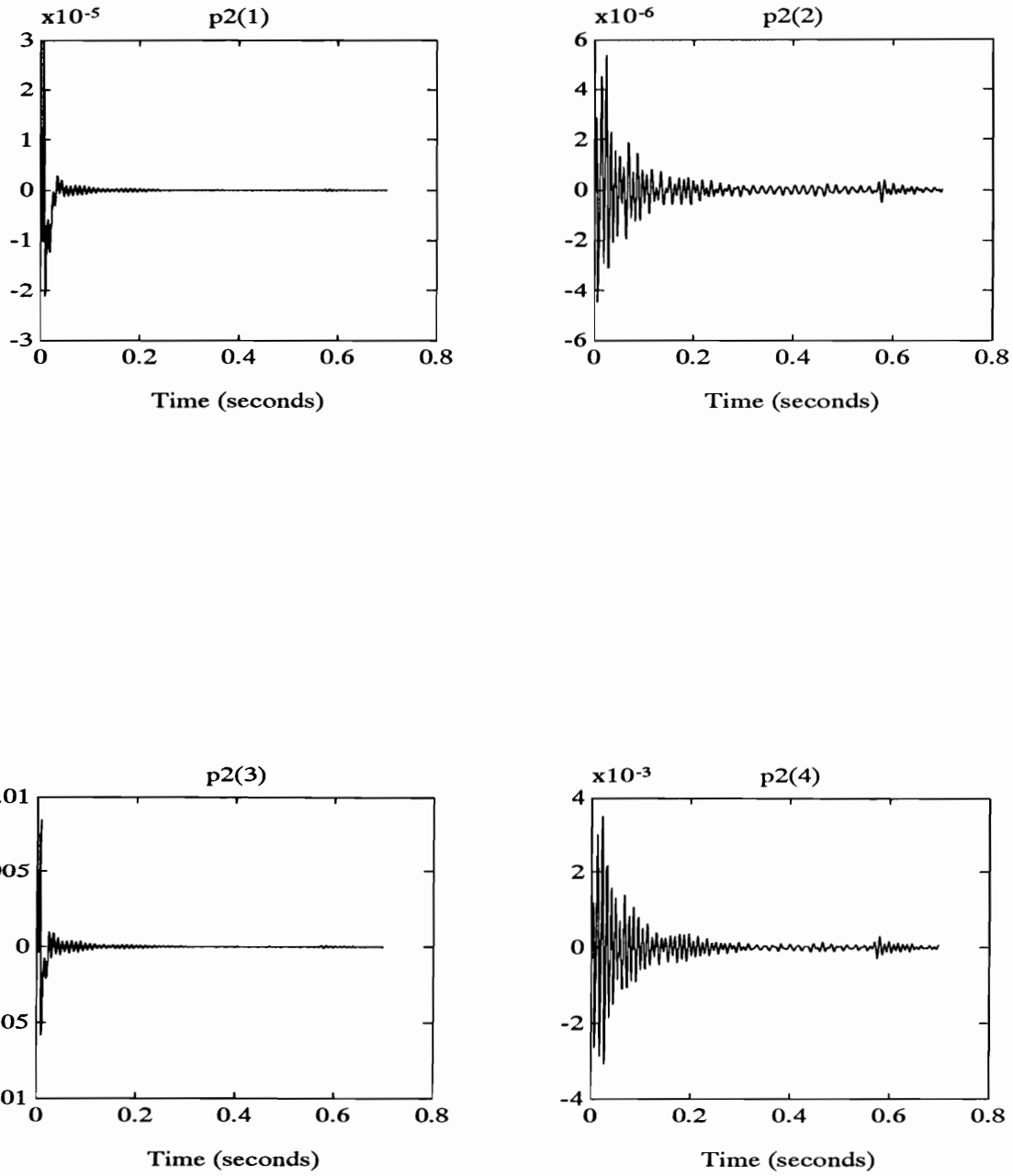


Figure 6.1: Propagation of State Error Covariance Matrix Entries $p^{(2)}$

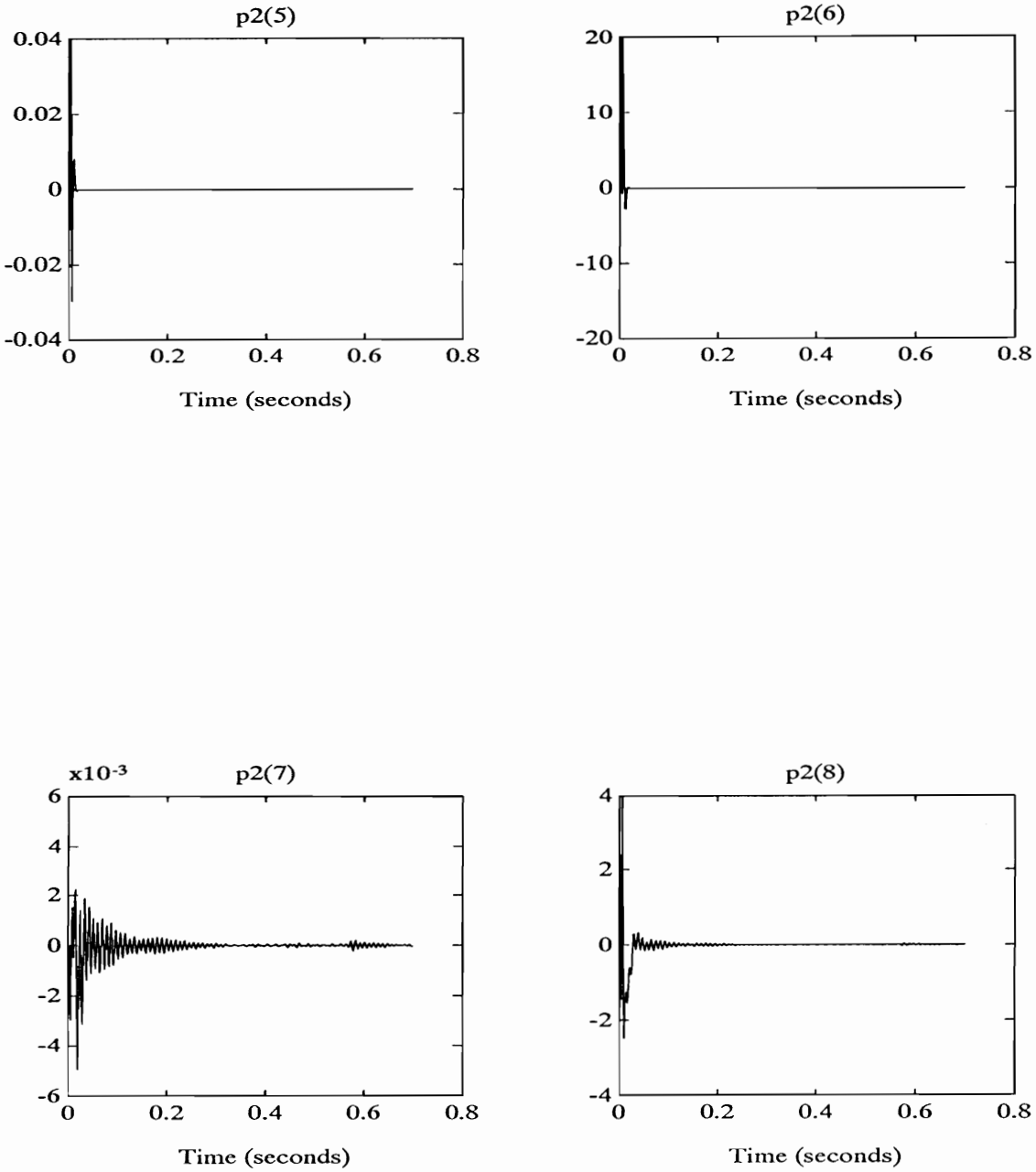


Figure 6.1: Propagation of State Error Covariance Matrix Entries $p^{(2)}$ (Continued)

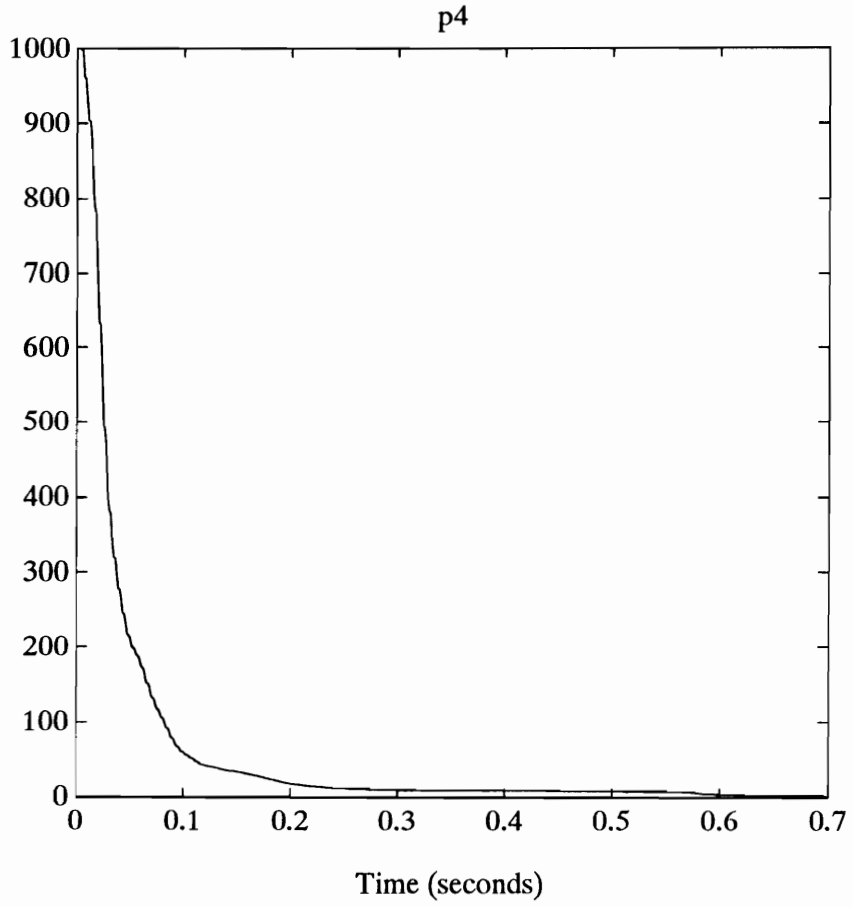


Figure 6.2: Propagation of State Error Covariance Matrix Entry $p^{(4)}$

$$p^{(3)}_{k/k} = p^{(2)^{\mathrm{T}}}_{k/k} \tag{6.1.11}$$

$$p^{(4)}_{k/k} = p^{(4)}_{k/k-1} - k^{(2)}_k dc^{(1)}\, p^{(2)}_{k/k-1} - k^{(2)}_k dc^{(2)}\, p^{(4)}_{k/k-1} \tag{6.1.12}$$

where

$$Q_a = \begin{bmatrix} q^{(1)} & q^{(2)} \\ q^{(3)} & q^{(4)} \end{bmatrix}. \tag{6.1.13}$$

With the simplification described above, the number of calculations is reduced considerably since part of the state error covariance matrix is fixed and no multiplication of two $(N+1)\times(N+1)$ matrices is needed. Take a two-mode controller for example, N equals 8 for the simply-supported plate in Chapter 2. In the simplified algorithm, the multiplication of two 9×9 matrices is reduced to the multiplication of a (8×8) matrix and a (8×1) vector and several operations that are not computationally intensive. The computing load is increased with the number of parameters to be estimated and the rate of speedup is expected to be proportional to the square of the number or less.

6.2 Experiment

As in Chapter 5, the problem of rejecting a disturbance in a simply-supported plate in the face of model uncertainty is considered. The simplified algorithm is used. A simulated result of parameter estimation is shown in Figure 6.3 and modal accelerations are plotted in Figure 6.4. The estimator is turned on in the beginning of the simulation at 0 second as before and the controller is turned on at 0.24 second.

When the self-tuning controller with simplified computation method was implemented on our transputer system, the sampling rate was 1636 Hz, which is more than twice the speed the transputer system can give for the controller without the simplified algorithm. Experimental results are shown in Figure 6.5

and Figure 6.6.

It can be seen from simulated and experimental results that by applying the simplified algorithm the self-tuning controller can still estimate the unknown parameter quite well without losing performance. The control signal pumps up the signal level of the 2nd mode acceleration, which is rich enough to make the convergence of the estimate very fast. For the open-loop estimation in the simulation, the richness of the 1st mode acceleration contributes a slower convergence of the estimate, however, in the experiment the modal accelerations don't seem to be rich enough to help the speed of convergence and the result shows very slow convergence.

In Eq.(5.2.16), the matrix to be inverted is $dC_{as}\ P\ dC_{as}^{\mathrm{T}} + R_a$, the dimension of which depends on the number of output variables and in our case is equal to the number of vibration modes. Directly calculating the inverse of a matrix is usually not recommended unless the dimension is smaller than 3. A method of obtaining the Kalman gains in Eq.(5.2.16) without actually performing the matrix inversion is to rearrange the equation as

$$AK^{\mathrm{T}} = B,$$

where $B = (dC_{as}\ P\ dC_{as}^{\mathrm{T}})^{\mathrm{T}}$ and $A = (P\ dC_{as}^{\mathrm{T}} + R_a)^{\mathrm{T}}$, and solve for K^{T} column by column. This method needs to transform the linear equation such that matrix A is triangular and perform back substitutions to get the result.

Assuming that the variation of the parameter is small and using the fact that the state error covariance matrix is partially fixed, we can write

$$(dC_{as}\ P\ dC_{as}^{\mathrm{T}} + R_a\ )^{-1} = (\Delta\ + \delta\ )^{-1},$$

such that $\|\delta\| \ll \|\Delta\|$, where Δ is fixed and δ is the variation. If this were true, we could approximate the matrix inversion by

$$(\Delta\ + \delta\ )^{-1} \simeq \Delta^{-1} - \delta\ \Delta^{-2},$$

where Δ^{-1} and Δ^{-2} can be calculated in advance and inversion of a matrix is

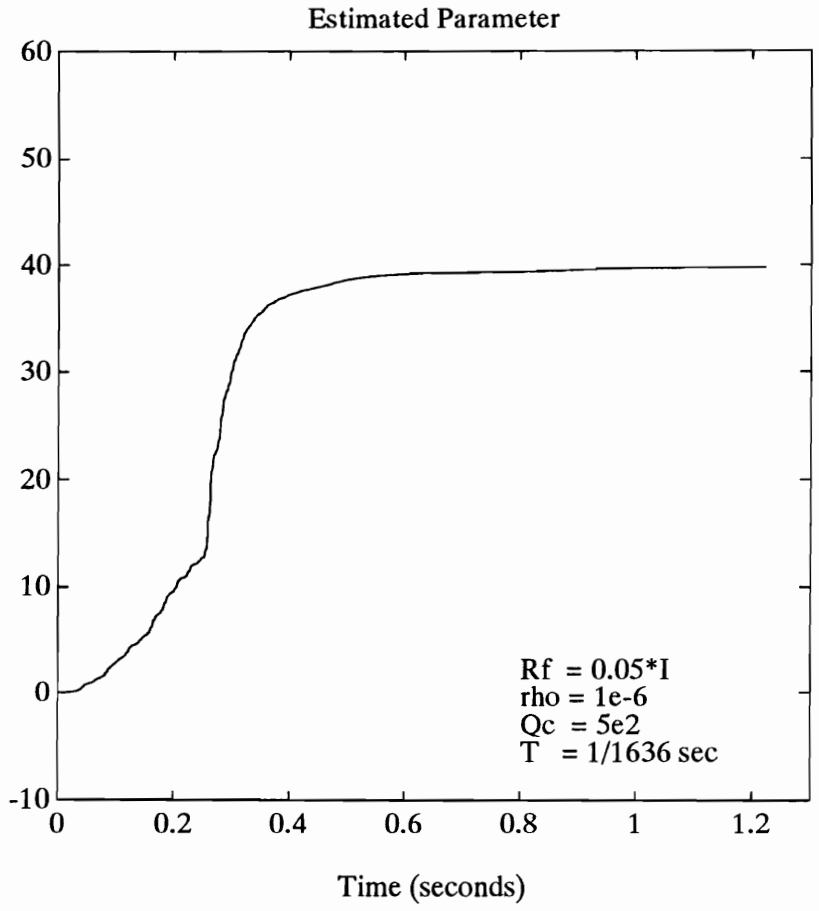
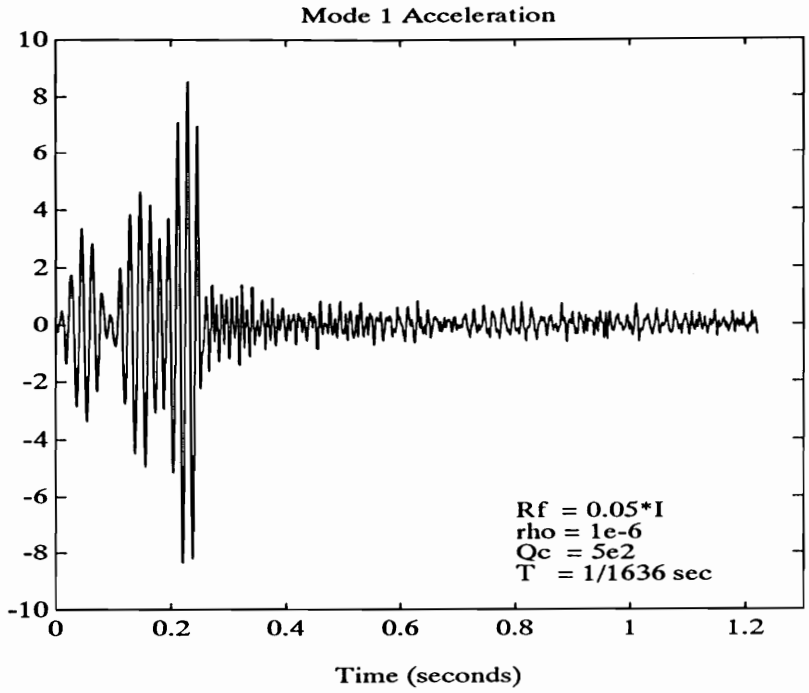


Figure 6.3: Estimation of $\Delta\omega_2$ (Simulated, Simplified Algorithm)



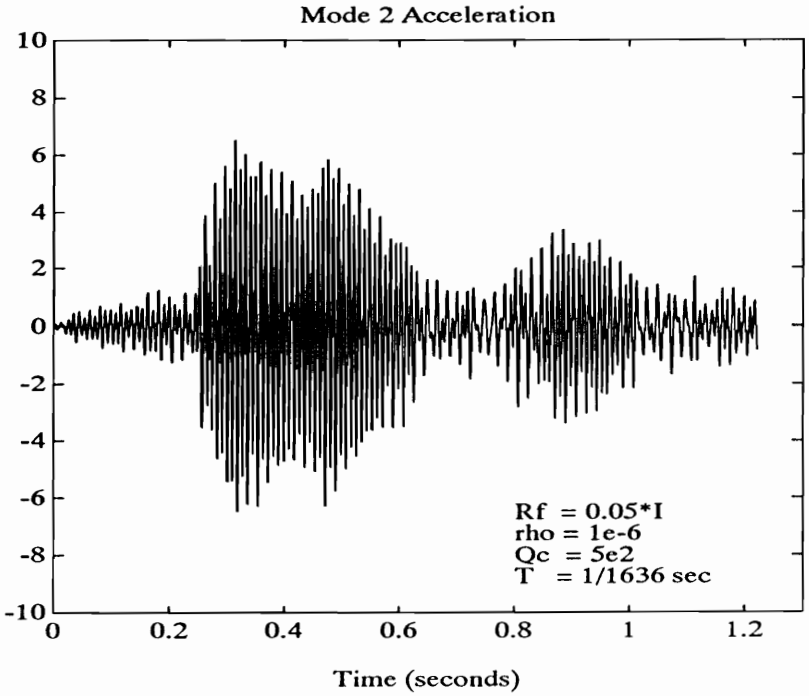


Figure 6.4: Self-Tuning LQG Disturbance Rejection (Simulated, Simplified Algorithm)

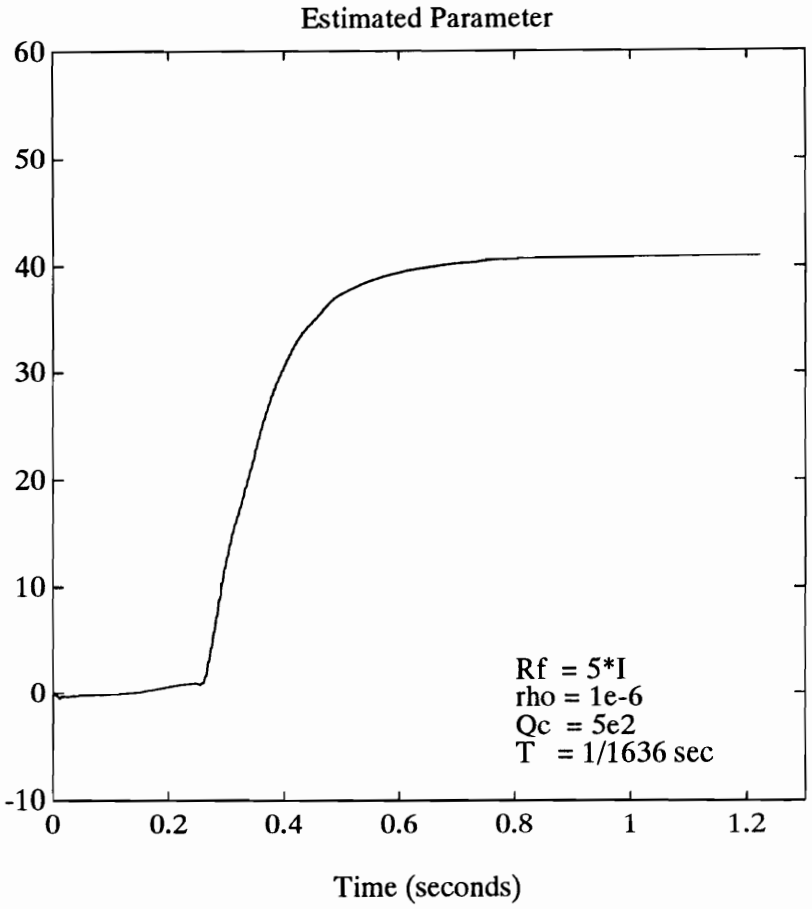
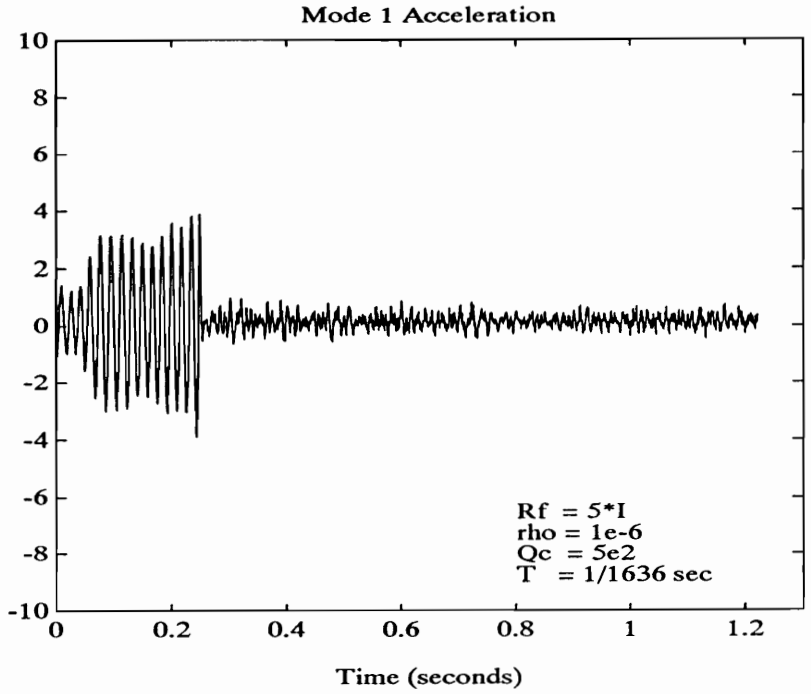


Figure 6.5: Estimation of $\Delta\omega_2$ (Experimental, Simplified Algorithm)



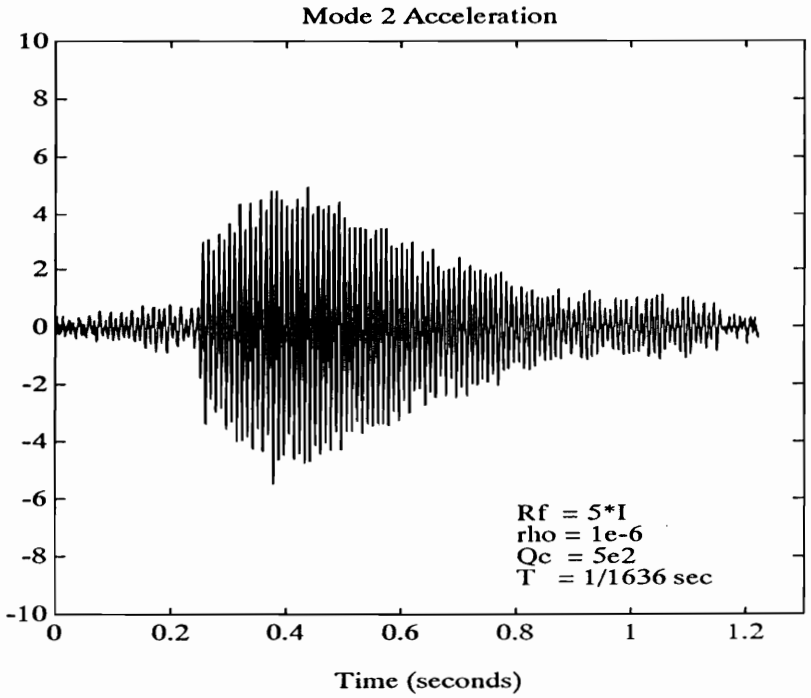


Figure 6.6: Self-Tuning LQG Disturbance Rejection (Experimental, Simplified Algorithm)

avoided. However, it was found that the assumption $\|\delta\| \ll \|\Delta\|$ is not always true, especially when the estimate is close to the actual value, and the parameter estimate won't be able to converge.

Matrix inversions exist in most parameter estimation algorithms for multiple-output cases and they are still the bottle-neck of successful on-line application of those algorithms.

7. STR Design: Unknown Disturbance Location

In addition to the uncertainty in the system parameters, another uncertainty occurs when the location where the disturbance enters the structure is modeled. Recursively identifying the disturbance location is desired to get better performance. While this problem could be treated by the technique described in Chapter 5 as a special case, since the disturbance influence matrix parameters, just like the natural frequencies, can be looked upon as variables in an augmented system matrix, extra assumptions let us avoid the nonlinear filtering problem. In this chapter, we assume that the disturbance can be measured and the augmented model, upon which the Kalman filter state estimator and feedback gains calculations are based, is approximated such that only linear filtering is required for the self-tuning controller to get estimated state and unknown parameters. A simply-supported plate example is given and simulated results are presented.

7.1 Discrete-Time Approximate Design Model

Given the combined model of a flexible structure and a control signal smoothing filter,

$$\dot{x} = Fx + Gu + Ld \tag{7.1.1}$$

$$y = Cx + Du + Ed + \theta, \tag{7.1.2}$$

the assumption that where the disturbance enters the plant is unknown corresponds to having unknown coefficient matrices L and E. The approach to solving this problem is to factor out the unknown parameters and augment them to the state variables of the plant such that a state estimator can be applied to the augmented state vector to get the estimated state for feedback and the estimated unknown parameters for a more accurate model.

If the discrete-time design model is obtained by the continuous-to-discrete transformation of the augmented continuous-time model, the unknown matrix L gets nonlinearly smeared over the whole state transition matrix. This will force us to either deal with a nonlinear filtering problem or to linearly estimate all of the parameters. An approximation can be made on the model, however, such that nonlinear filtering is avoided and the parameters to be identified retain their physical meanings.

Under the assumption that the disturbance can be measured, we start with obtaining an approximate discrete-time design model. Since the plant is controlled by a computer-generated control force, it can be best described by a zero-order-hold(ZOH) equivalent discrete-time model. The disturbance affects the plant continuously and because the sampling rate is much higher than the modeled modes and the disturbance frequencies, this can be described approximately by a ramp invariant(RI) model. If we assume that the control force input remains constant during sampling instants and the disturbance force input varies linearly between samples, we will end up with a discrete-time equivalent model described by the equations

$$x_k = dF\ x_{k-1} + dG\ u_{k-1} + dL\ d_{k-1} \tag{7.1.3}$$

$$y_k = dC\ x_k + dD\ u_k + dE\ d_k\ , \tag{7.1.4}$$

and in short notations

$$\{F,G,C,D,\} \overset{\text{ZOH}}{\rightarrow} \{dF,dG,dC,dD\}$$

$$\{F,L,C,E,\} \overset{RI}{\rightarrow} \{dF,dL,dC,dE\}.$$

Since the disturbance between two sampling instants is approximated by a linear relation, the state and output of a RI model at a given time instant are affected by the disturbance at and one time step before the given instant. In this chapter, we will modify the work done by Bingulac and VanLandingham[43] of obtaining a standard quadruple of matrices RI equivalent model and describe how this modification can allow us to use a linear estimator. The derivation of the

standard quadruple of matrices RI model is given in the next section.

7.2 Sample-and-Hold Discrete-Time Models

Sample-and-hold equivalence is an approach to find a discrete transfer function $G(z)$ that will have approximately the same characteristics as a continuous-time transfer function $G(s)$. Suppose that the input signal $u(t)$ of $G(s)$ is approximated by a piecewise constant signal, $\widehat{u}(t)$, resulting from sampling $u(t)$ at discrete instants k to get $u(k)$ and holding it constant between the instants k and $k+1$. This operation is called a zero-order-hold(ZOH) and the approximate discrete-time model is called a zero-order-hold equivalent model. The effect of the ZOH operation is a time delay of $T/2$ on the average between the original continuous-time input $u(t)$ and the piecewise constant input $\widehat{u}(t)$. In computer control of physical systems where analog-to-digital(A/D) and digital-to-analog(D/A) conversions are involved, the signal is sampled before it goes into a digital controller and at each instant the output signal from the controller is kept constant during that sampling period in order to have a piecewise continuous signal to control the physical plant. These computer controlled systems can be exactly described by ZOH models.

While the ZOH equivalent model is obtained by approximating the continuous input signal by a piecewise continuous signal, a piecewise constant signal actually, there are a lot of other piecewise continuous functions that can be used to approximate the input signal and obtain equivalent discrete-time transfer functions. If we use a first-order polynomial for extrapolation, we have a first-order-hold(FOH) and so on for second- and nth-order holds. A very special hold equivalent called ramp-invariant(RI) or triangle hold can be obtained if we approximate the input signal by a piecewise continuous signal resulting from connecting sample to sample in a straight line.

In this section, after reviewing the ZOH model, we will derive the RI equivalent model in standard quadruple of matrices state space form by two different methods and point out the difference between FOH and RI equivalent models.

7.2.1 ZOH Equivalent Model

Given a transfer function $\widehat{G}(s)$ with input $u(t)$ and output $y(t)$, the sample-and-hold equivalents can be constructed as in Figure 7.1 . For a zero-order-hold equivalent, the input signal $u(t)$ is sampled and held as constant such that

$$\widehat{u}(t) \;=\; u(kT) \qquad\qquad kT \leq\; t < kT + T \;, \tag{7.2.1}$$

where $\widehat{u}(t)$ is piecewise continuous signal and T is the sampling period. Thus, the input of the hold block in Figure 7.1(b) is a sequence of pulses and the output of the hold block is a piecewise constant signal as shown in Figure 7.2 . The Laplace transform of the input pulses is

$$\mathcal{L}\{\sum_{k\,=\,0}^{\infty} u(kT)\;\delta(t-kT)\} = \sum_{k\,=\,0}^{\infty} u(kT)e^{\,-\,ksT}\;, \tag{7.2.2}$$

and the Laplace transform of the output signal is

$$\begin{aligned}
&\mathcal{L}\{\sum_{k\,=\,0}^{\infty} u(kT)\;1(t-kT) - u(kT)\;1(t-(k+1)T)\} \\
&\qquad = \sum_{k\,=\,0}^{\infty} u(kT)\;\frac{e^{\,-\,ksT}}{s} \;-\sum_{k\,=\,0}^{\infty} u(kT)\;\frac{e^{\,-\,(k\,+\,1)sT}}{s} \\
&\qquad = \sum_{k\,=\,0}^{\infty} u(kT)\;e^{\,-\,ksT}\;[\;\frac{1-e^{\,-\,sT}}{s}\;]\;.
\end{aligned} \tag{7.2.3}$$

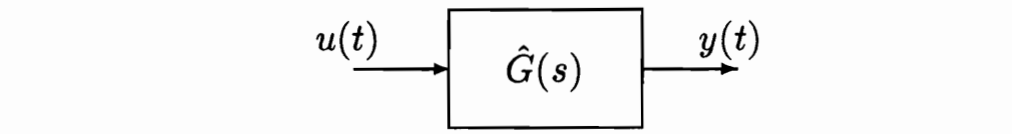
The transfer function of the hold block can be obtained from the Laplace transforms of the input and the output and is denoted by

$$ZOH(s) = \;\frac{1-e^{\,-\,sT}}{s}\;, \tag{7.2.4}$$

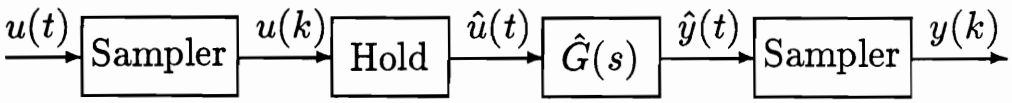
and the ZOH equivalent model of the transfer function $\widehat{G}(s)$ is

$$\widehat{G}(z) = \mathcal{Z}[(1-e^{\,-\,sT})\;\frac{\widehat{G}(s)}{s}\;] = (1-z^{\,-\,1})\;\mathcal{Z}\Big\{\;\frac{\widehat{G}(s)}{s}\;\Big\}\;. \tag{7.2.5}$$

Based upon the block diagram of the ZOH equivalent in Figure 7.3, the



(a) A continuous-time transfer function



(b) Block diagram of an equivalent system

Figure 7.1: System Construction of Hold Equivalents

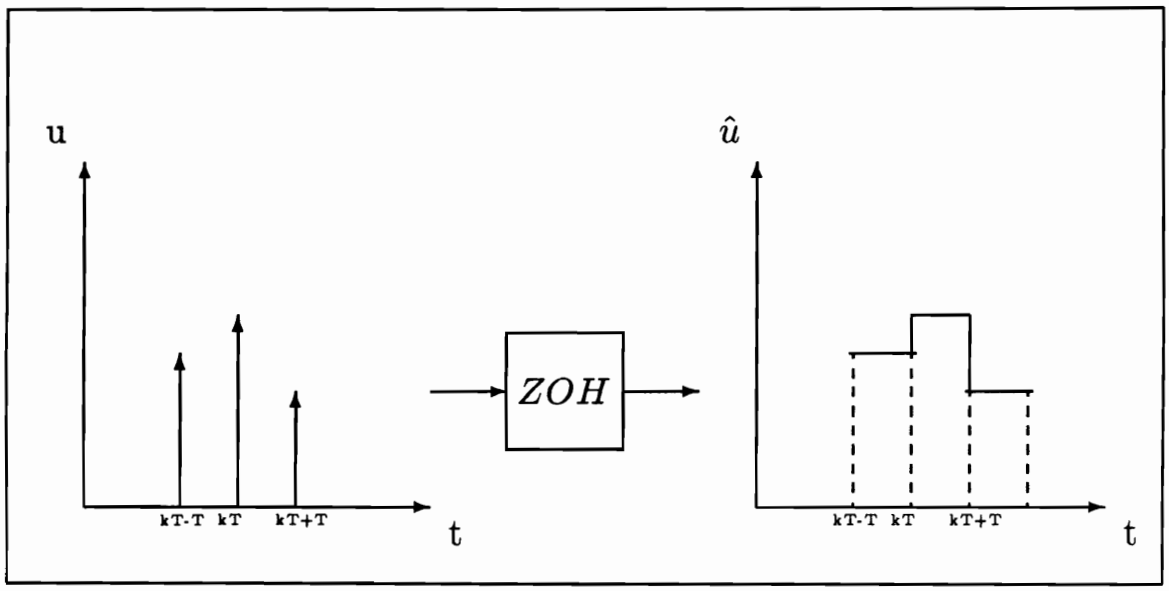


Figure 7.2: Input and Output of ZOH Block

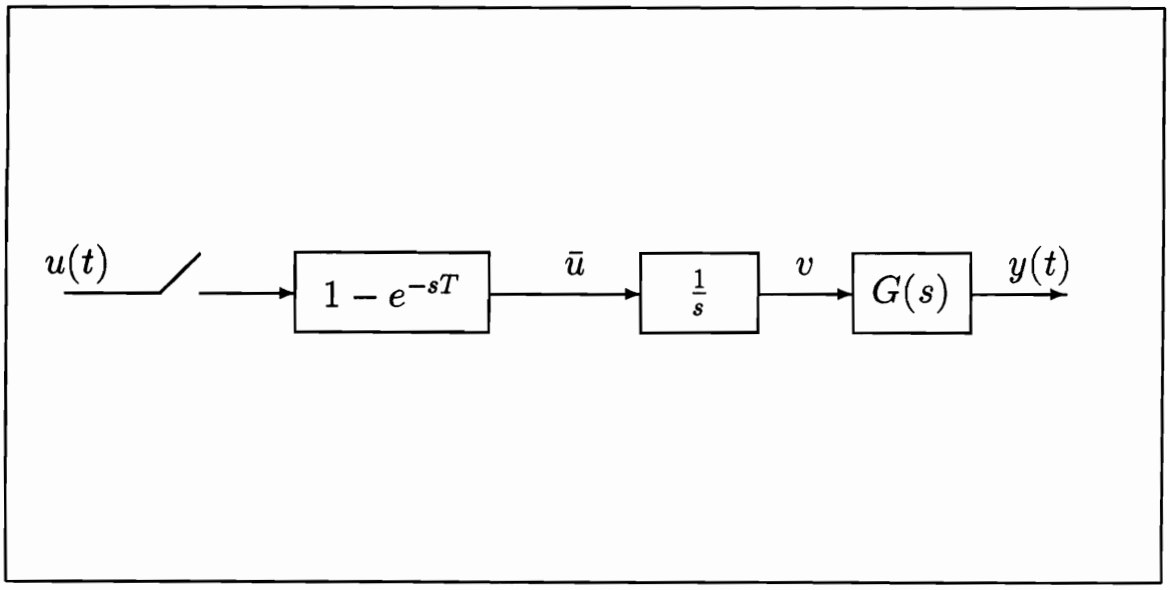


Figure 7.3: Block Diagram of ZOH Equivalent Models

state space representation of the ZOH model can be obtained as

$$\dot{x} = F\,x + G\,v \tag{7.2.6}$$

$$\dot{v} = \overline{u} \tag{7.2.7}$$

where

$$\overline{u} = u(t)\delta(t) - u(t-T)\delta(t-T)\,, \tag{7.2.8}$$

and in matrix form

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} F & G \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \overline{u}\,. \tag{7.2.9}$$

Let F_T be the system matrix in the above equation and define

$$e^{F_T T} = \begin{bmatrix} \Phi_1 & \Gamma_1 \\ 0 & 1 \end{bmatrix}, \tag{7.2.10}$$

then the x equation becomes

$$x_{k+1} = \Phi_1\, x_k + \Gamma_1\, v_k\,. \tag{7.2.11}$$

By integrating Eq. (7.2.7), it can be shown that

$$v_k = u_k\,, \tag{7.2.12}$$

so the ZOH equivalent state space model becomes

$$x_{k+1} = \Phi_1\, x_k + \Gamma_1\, u_k\,. \tag{7.2.13}$$

Another way to derive the ZOH equivalent discrete-time state space model is to start from a continuous-time state-space model. Let $\{F,G,C,D\}$ be a realization of the transfer function $\widehat{G}(s)$ such that

$$\dot{x}(t) = F\,x(t) + G\,u(t) \tag{7.2.14}$$

$$y(t) = C\,x(t) + D\,u(t)\,. \tag{7.2.15}$$

The general solution to Eq. (7.2.14) is

$$x(t) = e^{F(t-t_0)}\,x(t_0) + \int_{t_0}^{t} e^{F(t-\tau)}\,Gu(\tau)\,d\tau\,, \tag{7.2.16}$$

and if we let $t_0 = kT$ and $t = (k+1)T$ then Eq. (7.2.16) becomes

$$x((k+1)T) = e^{FT}\,x(kT) + \int_{kT}^{(k+1)T} e^{F\left((k+1)T-\tau\right)}\,Gu(\tau)\,d\tau\,. \tag{7.2.17}$$

Since

$$u(\tau) = u(kT) \qquad\qquad kT \le \tau < (k+1)T\,,$$

we can write Eq.(7.2.17) as

$$x((k+1)T) = e^{FT}\,x(kT) + \int_{0}^{T} e^{F\eta}\,G\,u(kT)\,d\eta\,. \tag{7.2.18}$$

by changing variables in the integral such that

$$\eta = (k+1)T - \tau\,. \tag{7.2.19}$$

By making T implicit, Eq. (7.2.18) can be expressed by

$$x_{k+1} = \Phi_2 x_k + \Gamma_2 u_k , \tag{7.2.20}$$

where

$$\Phi_2 = e^{FT}$$

$$\Gamma_2 = \int_0^T e^{F\eta} d\eta \, G .$$

Now, since $e^{F_T t}$ can be written in series expansion as

$$e^{F_T t} = \sum_{i=0}^{\infty} \frac{F_T^i T^i}{i!} = I + F_T T + \frac{F_T^2 T^2}{2!} + \frac{F_T^3 T^3}{3!} + \cdots$$

$$= I + \begin{bmatrix} F & G \\ 0 & 0 \end{bmatrix} T + \begin{bmatrix} F^2 & FG \\ 0 & 0 \end{bmatrix} \frac{T^2}{2!} + \begin{bmatrix} F^3 & F^2 G \\ 0 & 0 \end{bmatrix} \frac{T^3}{3!} + \cdots , \tag{7.2.21}$$

it is easily seen that

$$\Gamma_1 = 0 + GT + FG\frac{T^2}{2!} + F^2 G \frac{T^3}{3!} + \cdots$$

$$= (T + F \frac{T^2}{2!} + F^2 \frac{T^3}{3!} + \cdots) G$$

$$= \int_0^T e^{F\eta} d\eta \, G = \Gamma_2 ,$$

and similarly

$$\Phi_1 = \Phi_2 .$$

Thus, we have arrived at the same result deriving ZOH discrete-time state

space models by two methods with a different point of view.

7.2.2 RI Equivalent Model - Method 1

Usually, starting from a continuous-time state space model

$$\dot{x}(t) = F\ x(t) + L\ u(t) \tag{7.2.22}$$

$$y(t) = C\ x(t) + E\ u(t)\ , \tag{7.2.23}$$

an equivalent RI discrete-time model can be obtained as $\{dF_5, dL_0, dL_1, dC_5, dE_5\}$, a quintuple of matrices describing the model

$$x(k) = dF_5\ x(k-1) + dL_0\ u(k-1) + dL_1\ u(k) \tag{7.2.24}$$

$$y(k) = dC_5\ x(k) + dE_5\ u(k). \tag{7.2.25}$$

It can be shown[44] that

$$dF_5 = I + UFT \tag{7.2.26}$$

$$dC_5 = C \tag{7.2.27}$$

$$dE_5 = E \tag{7.2.28}$$

$$dL_0 = (U - V)\ L\ T \tag{7.2.29}$$

$$dL_1 = V\ L\ T, \tag{7.2.30}$$

where

$$U = \sum_{i=0}^{\infty} \frac{(FT)^i}{(i+1)!} \tag{7.2.31}$$

$$V = \sum_{i=0}^{\infty} \frac{(FT)^i}{(i+2)!}\ , \tag{7.2.32}$$

and T is the sampling period.

The problem of finding a standard quadruple of matrices equivalent to the discrete-time RI model through a quintuple of matrices was solved in [43] but no simple linear relationship between the two sets of matrices was obtained. In order to find a linear relation useful to deriving an on-line adaptive algorithm, we equate the two transfer functions in terms of the matrices in each state-space model as

$$dC_5\,(z\,I - dF_5)^{-1}(dL_0 + z\,dL_1) + dE_5 = dC\,(z\,I - dF)^{-1}dL + dE. \tag{7.2.33}$$

If we let $dC = dC_5$ and $dF = dF_5$, then we have for the left-hand side of the equation

$$C(z\,I - dF)^{-1}\,dL_0 + C(z\,I - dF)^{-1}dF\;dL_1 + C\;dL_1 + dE_5\,. \tag{7.2.34}$$

Comparison with the right-hand side of the equation results in

$$dL = dL_0 + dF\;dL_1 \tag{7.2.35}$$

$$dE = E + C\;dL_1\,. \tag{7.2.36}$$

Substituting Eq. (7.2.29) and Eq. (7.2.30) into Eq. (7.2.35), we get

$$\begin{aligned} dL &= (U - V)\;L\;T + dF\;V\;L\;T \\ &= (U - V + dF\;V)\;L\;T\,, \end{aligned} \tag{7.2.37}$$

and L can be written in terms of dL as

$$L = (U - V + dF\;V)^{-1}dL \times \frac{1}{T}\,. \tag{7.2.38}$$

Summarizing the results, we have

$$dF = I + U\ F\ T \tag{7.2.39}$$

$$dL = (U - V + dF\ V)\ L\ T \tag{7.2.40}$$

$$dC = C \tag{7.2.41}$$

$$dE = E\ +\ C\ V\ L\ T\ , \tag{7.2.42}$$

where

$$U = \sum_{i=0}^{\infty} \frac{(FT)^i}{(i+1)!} \tag{7.2.43}$$

$$V = \sum_{i=0}^{\infty} \frac{(FT)^i}{(i+2)!}\ . \tag{7.2.44}$$

7.2.3 RI Equivalent Model - Method 2

Similar to ZOH equivalent, the state space RI or triangle hold equivalent can be obtained by first finding the transfer function of the hold block in Figure 7.1, the general sample-and-hold equivalent block diagram.

Given the input signal of a sequence of pulses, the output piecewise continuous signal is shown in Figure 7.4 . Similarly, the Laplace transform of the input pulses is

$$\mathcal{L}\{\sum_{k=0}^{\infty} u(kT)\ \delta(t-kT)\} = \sum_{k=0}^{\infty} u(kT)e^{-ksT}\ . \tag{7.2.45}$$

The output signal can be expressed in summation form

$$\begin{aligned} v(t) = &\sum_{k=0}^{\infty} u(kT)1(t-kT) + \frac{u((k+1)T) - u(kT)}{T}\,(t-kT) \\ &- \frac{u((k+1)T) - u(kT)}{T}\,(t-(k+1)T) - u((k+1)T)1(t-(k+1)T)\ , \end{aligned} \tag{7.2.46}$$

and the Laplace transform of the first and the fourth terms in the right hand side of Eq. (7.2.46) cancel, so we have

$$\mathcal{L}\{v(t)\} = \sum_{k=0}^{\infty} u((k+1)T)\,\frac{e^{-ksT}}{Ts^2} - \sum_{k=0}^{\infty} u(kT)\,\frac{e^{-ksT}}{Ts^2} - \sum_{k=0}^{\infty} u((k+1)T)\,\frac{e^{-(k+1)sT}}{Ts^2} + \sum_{k=0}^{\infty} u(kT)\,\frac{e^{-(k+1)sT}}{Ts^2}$$

$$= \sum_{k=0}^{\infty} u(kT)\,e^{-ksT}\left(\frac{e^{-sT}}{Ts^2}\right) - \sum_{k=0}^{\infty} u(kT)\,e^{-ksT}\left(\frac{1}{Ts^2}\right) - \sum_{k=0}^{\infty} u(kT)\,e^{-ksT}\left(\frac{1}{Ts^2}\right) + \sum_{k=0}^{\infty} u(kT)\,e^{-ksT}\left(\frac{e^{sT}}{Ts^2}\right)$$

$$= \left[\sum_{k=0}^{\infty} u(kT)\,e^{-ksT}\right]\left[\frac{e^{-sT} - 2 + e^{sT}}{Ts^2}\right]. \tag{7.2.47}$$

Since the Laplace transform of the output signal is expressed in terms of the Laplace transform of the input, the transfer function of the hold block is shown to be

$$RI(s) = \left[\frac{e^{-sT} - 2 + e^{sT}}{Ts^2}\right], \tag{7.2.48}$$

and the RI equivalent model of the transfer function $\widehat{G}(s)$ is

$$\widehat{G}(z) = \mathcal{Z}\left[\left(\frac{e^{-sT} - 2 + e^{sT}}{T}\right)\frac{\widehat{G}(s)}{s^2}\right] = \left(\frac{z^{-1} - 2 + z}{T}\right)\mathcal{Z}\left[\frac{\widehat{G}(s)}{s^2}\right]. \tag{7.2.49}$$

Similarly, based upon the block diagram of the RI equivalent in Figure 7.5, the state space representation of the RI model can be obtained as

$$\dot{x} = F\,x + L\,v \tag{7.2.50}$$

$$\dot{v} = w/T \tag{7.2.51}$$

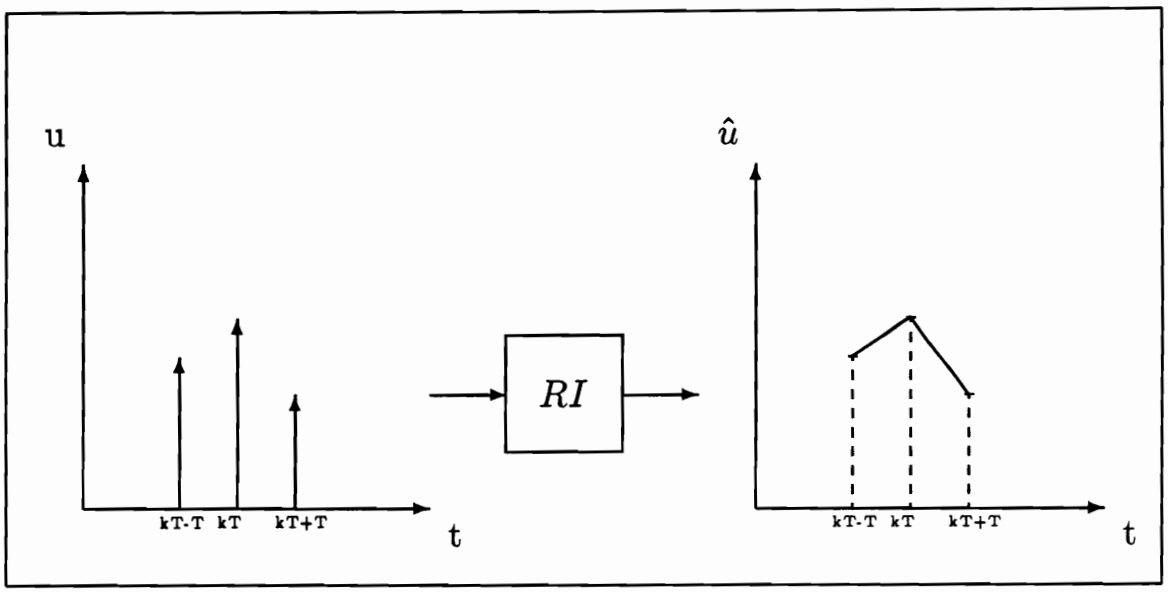


Figure 7.4: Input and Output of RI Block

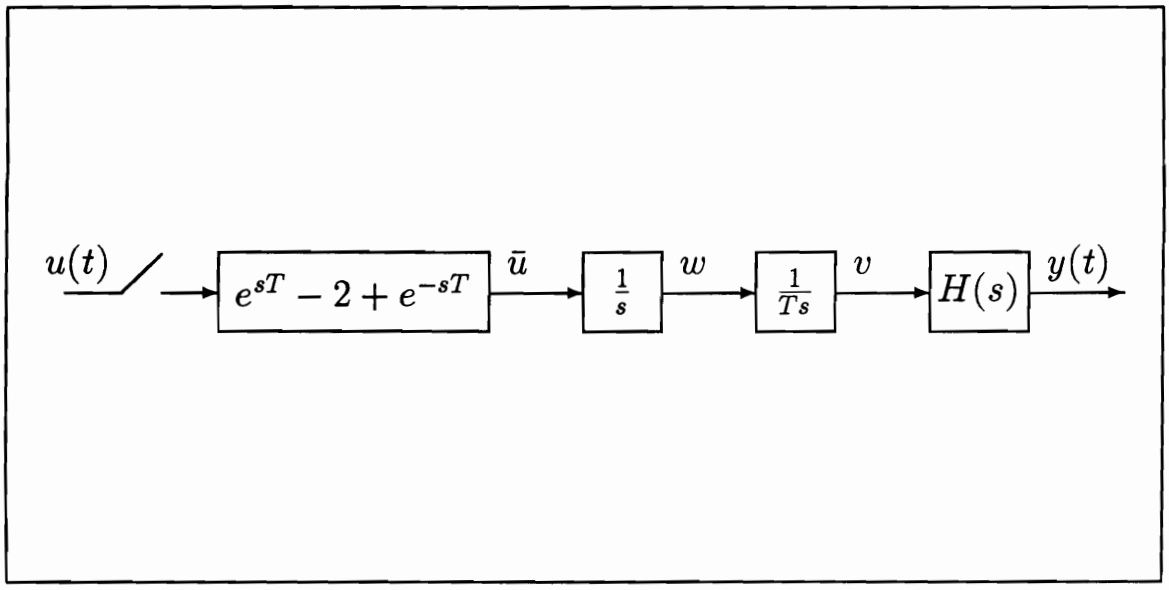


Figure 7.5: Block Diagram of RI Equivalent Models

$$\dot{w} = \overline{u}\ , \tag{7.2.52}$$

where

$$\overline{u}\ = u(t+T)\delta(t+T) - 2u(t)\delta(t) + u(t-T)\delta(t-T)\ , \tag{7.2.53}$$

and in matrix form

$$\begin{bmatrix} \dot{x} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} F & L & 0 \\ 0 & 0 & 1/T \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ v \\ w \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \overline{u}\ . \tag{7.2.54}$$

Now, let F_T be the square matrix in the above equation and define

$$e^{F_T T}\ = \begin{bmatrix} \Phi & \Gamma_1 & \Gamma_2 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \tag{7.2.55}$$

then the x equation becomes

$$x_{k+1} = \Phi\ x_k - \ \Gamma_1\ v_k\ +\ \Gamma_2 w_k\ . \tag{7.2.56}$$

Eq. (7.2.51) and Eq. (7.2.52) can be integrated to get

$$v_k = u_k \tag{7.2.57}$$

$$w_k = u_{k+1} - u_k\ , \tag{7.2.58}$$

and if we define a new state

$$\xi_k = x_k - \Gamma_2\ u_k\ , \tag{7.2.59}$$

then the RI equivalent state equation becomes

$$\xi_{k+1} = \Phi\, \xi_k + (\Gamma_1 + \Phi\Gamma_2 - \Gamma_2)u_k \tag{7.2.60}$$

and the output equation is

$$\begin{aligned} y_k &= C\, x_k + E\, u_k \\ &= C\, \xi_k + (E + C\Gamma_2)\, u_k\,. \end{aligned} \tag{7.2.61}$$

While in this section the RI discrete-time state space model is obtained as a quadruple of matrices $\{dF, dL, dC, dE\}$ such that

$$dF = \Phi$$

$$dL = \Gamma_1 + \Phi\Gamma_2 - \Gamma_2$$

$$dC = C$$

$$dE = E + C\Gamma_2\,,$$

it can be shown easily that

$$\Gamma_2 = dL_1$$

$$\Gamma_1 = dL_0 + \Gamma_2\,,$$

and we arrive at the same result in Section 7.2.2 .

Although in this chapter only the RI equivalent model is used to design our self-tuning regulator, it is instructive to clarify the difference between the FOH and the RI equivalent models. The distinction between the two models can be best seen by the derivation of the transfer function of the hold block. The input pulses to the hold block are the same for both models, however, the output piecewise continuous signals are different as can be seen in Figure 7.6 , and the

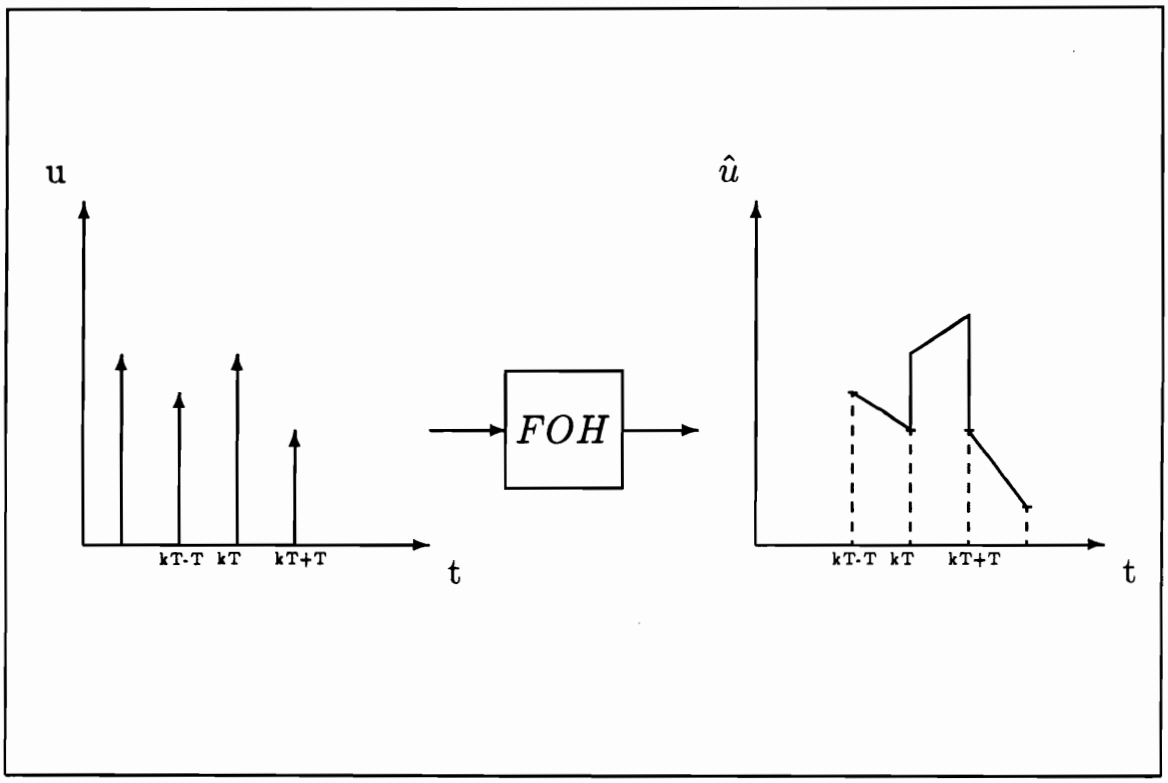


Figure 7.6: Input and Output of FOH Block

output signal in summation form for the FOH block is

$$v(t) = \sum_{k=0}^{\infty} u(kT)1(t-kT) + \frac{u(kT)-u((k-1)T)}{T}(t-kT)$$
$$-\frac{u(kT)-u((k-1)T)}{T}(t-(k+1)T) - u((k+1)T)1(t-(k+1)T)\ . \qquad (7.2.62)$$

The Laplace transform of the above signal is different from that of a RI model hold block output signal and the FOH equivalent model is totally different from the RI model.

7.3 Joint Parameter and State Estimation

The shaping filter used to model the disturbance represents a continuous system driven by white noise. Discretization of the system results in

$$r_k = dF_w\, r_{k-1} + v_{k-1} \qquad (7.3.1)$$

$$d_k = dC_w\, r_k\ . \qquad (7.3.2)$$

where $v_{k-1} = N(0, Q_d)$ and Q_d is the covariance matrix

$$Q_d = \int_0^T dF_w\, G_w\, G_w^{\mathrm{T}}\, dF_w^{\mathrm{T}}\, dt\ . \qquad (7.3.3)$$

Series connection of the two models described by Eq. (7.3.1), (7.3.2) and Eq. (7.1.3), (7.1.4) gives

$$\begin{bmatrix} x_k \\ r_k \end{bmatrix} = \begin{bmatrix} dF & dL\ dC_w \\ 0 & dF_w \end{bmatrix}\begin{bmatrix} x_{k-1} \\ r_{k-1} \end{bmatrix} + \begin{bmatrix} dG \\ 0 \end{bmatrix} u_{k-1} + \widehat{v}_{k-1}$$
$$= dF_{as}\begin{bmatrix} x_{k-1} \\ r_{k-1} \end{bmatrix} + dG_{as}\, u_{k-1} + \widehat{v}_{k-1}\ , \qquad (7.3.4)$$

where

$$\mathbb{E}\{\widehat{v}_{k-1}\,\widehat{v}_{k-1}^{\mathrm{T}}\} = \begin{bmatrix} 0 & 0 \\ 0 & Q_d \end{bmatrix}. \tag{7.3.5}$$

Thus, instead of applying one-step exact discretization to the augmented model, an approximate discrete-time model is obtained in two steps. The first discretization gives statistically correct sample values and the second is ramp invariant. While this approximate discrete-time model is good only for fast sampling and low-frequency disturbance, it keeps things linear in L and allows us to use a linear estimator.

Since Eq.(7.3.4) can be written as

$$\begin{bmatrix} x_k \\ r_k \end{bmatrix} = \begin{bmatrix} dF & 0 \\ 0 & dF_w \end{bmatrix} \begin{bmatrix} x_{k-1} \\ r_{k-1} \end{bmatrix} + \begin{bmatrix} dL \\ 0 \end{bmatrix} d_{k-1} + \begin{bmatrix} dG \\ 0 \end{bmatrix} u_{k-1} + \widehat{v}_{k-1}\,, \tag{7.3.6}$$

if we let $S = [\, x^{\mathrm{T}}\; r^{\mathrm{T}}\; dL^{\mathrm{T}}]^{\mathrm{T}}$, then we have from Eq. (7.3.7)

$$\begin{aligned} S_k &= \begin{bmatrix} dF & 0 & d_{k-1} \times I_{2n} \\ 0 & dF_w & 0 \\ 0 & 0 & I_{2n} \end{bmatrix} S_{k-1} + \begin{bmatrix} dG \\ 0 \\ 0 \end{bmatrix} u_{k-1} + \widetilde{v}_{k-1} \\ &= F_{s,k-1}\, S_{k-1} + G_s\, u_{k-1} + \widetilde{v}_{k-1}\,, \end{aligned} \tag{7.3.7}$$

where

$$\mathbb{E}[\,\widetilde{v}_{k-1}\,\widetilde{v}_{k-1}^{\mathrm{T}}\,] = Q_a = \begin{bmatrix} 0 & 0 & 0 \\ 0 & Q_d & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{7.3.8}$$

Since E can be written in terms of L as

$$E = [\ 0_n\ I_n\ 0_t]\ L = f_2\ L, \tag{7.3.9}$$

based on Eq. (7.2.38), we have

$$E\ = [\ 0_n\ \ I_n\ \ 0_t]\ [\ U - V + dF\ V\ ]^{-1} \times \frac{1}{T}\ dL\ .$$

$$= f\ \times\ dL \tag{7.3.10}$$

Combining measurements on disturbance and modal accelerations, we have

$$z_{a,k} = \begin{bmatrix} dC & 0 & d_k \times f \\ 0 & dC_w & 0 \end{bmatrix} S_k + \begin{bmatrix} dD \\ 0 \end{bmatrix} u_k + \theta \tag{7.3.11}$$

$$= \ H_s\ S_k + D_s\ u_k + \theta\ , \tag{7.3.12}$$

where $z_{a,k} = [\ z_k^{\mathrm{T}}\ \ d_k\ ]^{\mathrm{T}}$, and θ is the measurement noise.

If an initial estimate of S_0 with a covariance matrix P_0 is assumed, the discrete Kalman filter is known to be the optimal recursive estimator for S_k. The Kalman filter is implemented as a one-step-ahead predictor similar to that in Chapter 5 and the algorithm is given as follows.

$$P_{k/k-1} = F_{s,k-1}\ P_{k-1/k-1}\ F_{s,k-1}^{\mathrm{T}} + Q_a \tag{7.3.13}$$

$$K_k\ = P_{k/k-1}\ H_{s,k}^{\mathrm{T}}\ (H_{s,k}\ P_{k/k-1}\ H_{s,k}^{\mathrm{T}} + R_a)^{-1} \tag{7.3.14}$$

$$\widehat{S}_{k/k-1}\ = F_{s,k-1}\ \widehat{S}_{k-1/k-1} + G_s\ u_{k-1} \tag{7.3.15}$$

$$\widehat{z}_{a,k}\ = H_{s,k}\ \widehat{S}_{k/k-1} + D_s\ u_k \tag{7.3.16}$$

$$\widehat{S}_{k/k}\ = \widehat{S}_{k/k-1} + K_k\ [z_{a,k} - \widehat{z}_{a,k}\ ] \tag{7.3.17}$$

$$P_{k/k}\ = (\ I - K_k\ H_{s,k}\ )\ P_{k/k-1} \tag{7.3.18}$$

A closer examination on Eq.(7.3.7) and Eq.(7.3.11) reveals that we can estimate L instead of dL since a linear relation exists between dL and L as shown in Eq.(7.2.38). Because many entries in L are known to be zeros, the number of parameters to be estimated is cut down by half and save a lot of calculations. By substituting Eq.(7.2.38) into Eq.(7.3.4), we have

$$\begin{bmatrix} x_k \\ r_k \end{bmatrix} = \begin{bmatrix} dF & f_1\, L\, dC_w \\ 0 & dF_w \end{bmatrix} \begin{bmatrix} x_{k-1} \\ r_{k-1} \end{bmatrix} + \begin{bmatrix} dG \\ 0 \end{bmatrix} u_{k-1} + \widehat{v}_{k-1}$$

$$= dF_{as} \begin{bmatrix} x_{k-1} \\ r_{k-1} \end{bmatrix} + dG_{as}\, u_{k-1} + \widehat{v}_{k-1}\,, \tag{7.3.19}$$

where $f_1 = [U - V + dF\, V]\, T$.

Thus, if we define the extended state vector $S = [x^{\mathrm{T}}\ r^{\mathrm{T}}\ L^{\mathrm{T}}]^{\mathrm{T}}$, Eqs.(7.3.19) can be written by

$$S_k = \begin{bmatrix} dF & 0 & d_{k-1} \times f_1 \\ 0 & dF_w & 0 \\ 0 & 0 & I_{2n+t} \end{bmatrix} S_{k-1} + \begin{bmatrix} dG \\ 0 \\ 0 \end{bmatrix} u_{k-1} + \widetilde{v}_{k-1}$$

$$= F_{s,k-1}\, S_{k-1} + G_s\, u_{k-1} + \widetilde{v}_{k-1}\,, \tag{7.3.20}$$

and Eq.(7.3.11) becomes

$$z_{a,k} = \begin{bmatrix} dC & 0 & d_k \times f_2 \\ 0 & dC_w & 0 \end{bmatrix} S_k + \begin{bmatrix} dD \\ 0 \end{bmatrix} u_k + \theta$$

$$= H_s\, S_k + D_s\, u_k + \theta\,, \tag{7.3.21}$$

7.4 Control Law Design

The control law is exactly the same as that described in Chapter 5. However, the discrete-time model upon which the feedback gains are calculated is an approximate one and the model is more accurate for a higher sampling rate of the system.

7.5 Simulated Example

The disturbance rejection of a simply-supported plate is considered in this section. The model of the plate is assumed to be correct and the only unknown is the disturbance influence matrix. Simulation is performed for both a 60 Hz and a narrow-band disturbance. For a two mode controller,

$$dL = \begin{bmatrix} dl_1 \\ dl_2 \\ dl_3 \\ dl_4 \\ 0 \\ 0 \end{bmatrix},$$

and only $dL_4 = [dl_1\ dl_2\ dl_3\ dl_4]^{\mathrm{T}}$ needs to be estimated.

The initial conditions are set to

$$P_{0/0} = P_{0/-1} = 1000 \cdot I$$

$$\hat{S}_{0/0} = \hat{S}_{0/-1} = \begin{bmatrix} 0_{8\times 1} \\ dl_1(0) \\ dl_2(0) \\ dl_3(0) \\ dl_4(0) \end{bmatrix},$$

and $P_{lq,0}$, the initial condition of the solution of the discrete-time algebraic Riccati equation needed to calculate the feedback gains, and $[dl_1(0)\ dl_2(0)\ dl_3(0)\ dl_4(0)]^{\mathrm{T}}$ are calculated from the originally guessed model.

Table 7.1 lists the results of parameter estimation for a 60 Hz disturbance and Table 7.2 lists the results for a narrow-band disturbance. Figure 7.7 and Figure 7.8 plot the history of estimation. It is shown that the estimate converges in less than 10 iterations for the case of a 60 Hz disturbance and in less than 100 iterations for the narrow-band case. The performance of the controller is improved by better knowledge about the discrete-time disturbance influence matrix dL as can be seen by Figure 7.9 and Figure 7.10, the controlled output(accelerations) of the plate. The closed-loop response between 0.1 second and 0.3 second shows that when the compensators are designed based upon $dL = 0$, they just add damping to the plate but don't try to reject the disturbance directly. Much more effective rejections of the disturbances are shown in the plots between 0.3 second and 0.5 second when the disturbance influence matrices are correctly estimated.

Since

$$L = \begin{bmatrix} 0 \\ 0 \\ l_3 \\ l_4 \\ 0 \\ 0 \end{bmatrix},$$

in the case that L instead of dL is estimated, only $L_2 = [l_1\ l_2]^{\mathrm{T}}$ needs to be estimated.

Table 7.1 Estimation of D-T Disturbance Influence Matrix(60 Hz Disturbance)

actual $dL_4(\times 10^{-3})$	Initial Guess of $dL_4(\times 10^{-3})$	Estimate of $dL_4(\times 10^{-3})$ After			
		2	4	10	50
		Iterations			
0.0002	0	−0.0150	0.0002	0.0002	0.0002
−0.0001	0	0.0708	−0.0000	−0.0000	−0.0000
0.3380	0	0.3353	0.3358	0.3358	0.3358
−0.1388	0	−0.0486	−0.1405	−0.1405	−0.1405

Table 7.2 Estimation of D-T Disturbance Influence Matrix(Narrowband Disturbance)

actual $dL_4(\times 10^{-3})$	Initial Guess of $dL_4(\times 10^{-3})$	Estimate of $dL_4(\times 10^{-3})$ After			
		10	100	500	800
		Iterations			
0.0002	0	0.0023	0.0002	0.0002	0.0002
−0.0001	0	0.0001	−0.0000	−0.0001	−0.0000
0.3380	0	0.1043	0.3414	0.3370	0.3375
−0.1388	0	−0.1275	−0.1376	−0.1411	−0.1407

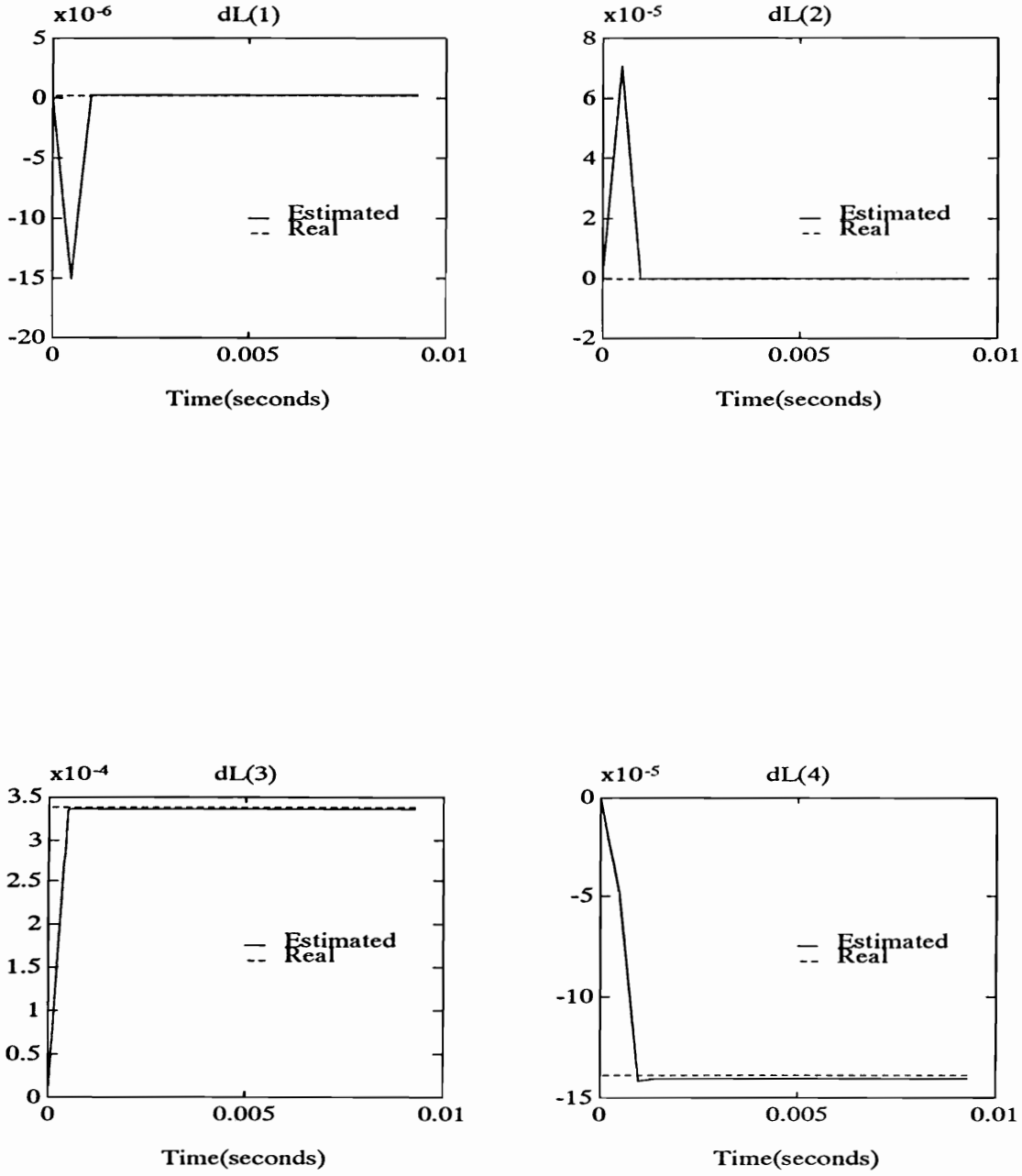


Figure 7.7: Estimation of D-T Disturbance Influence Matrix (60 Hz)

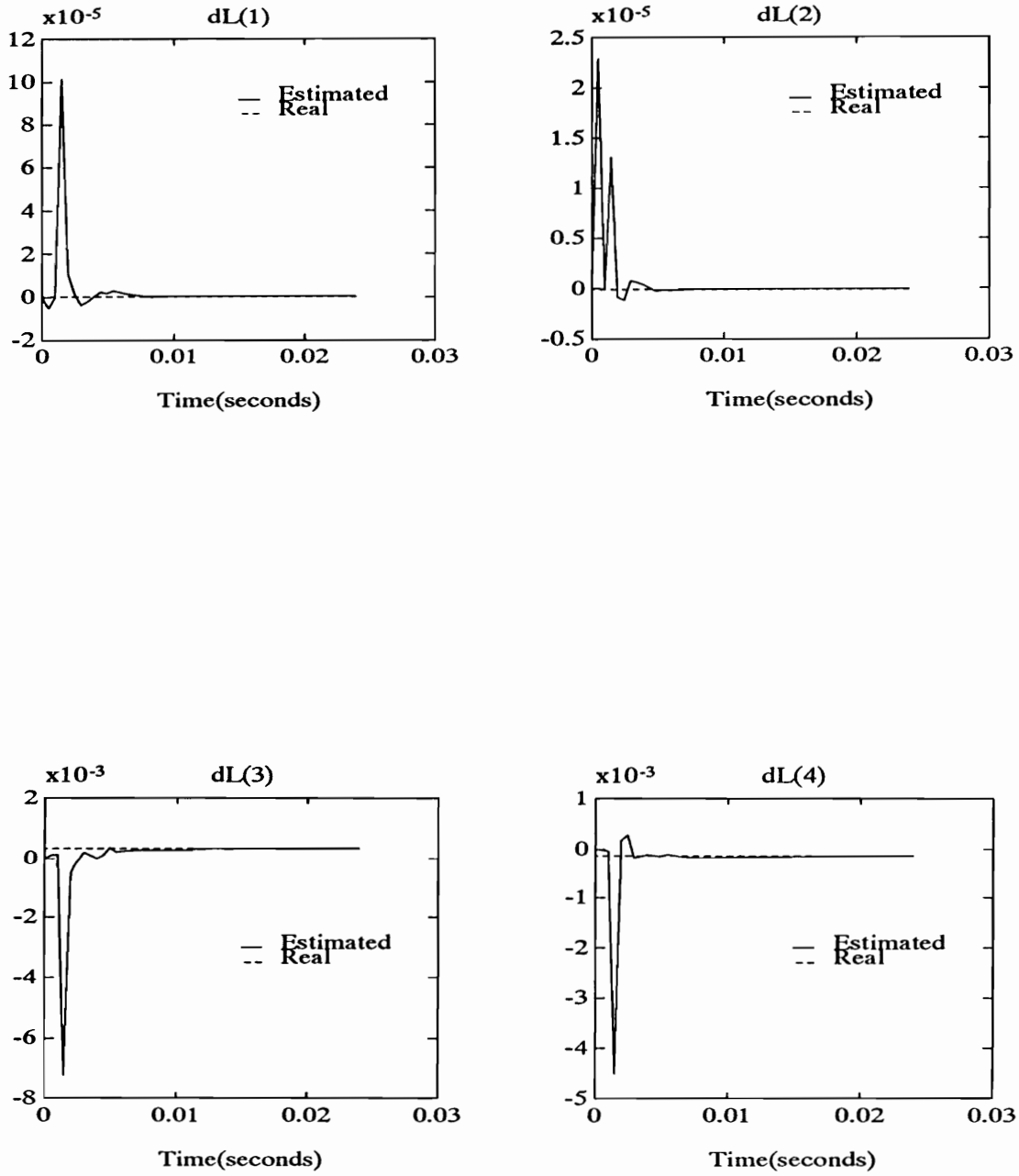


Figure 7.8: Estimation of D-T Disturbance Influence Matrix (Narrowband)

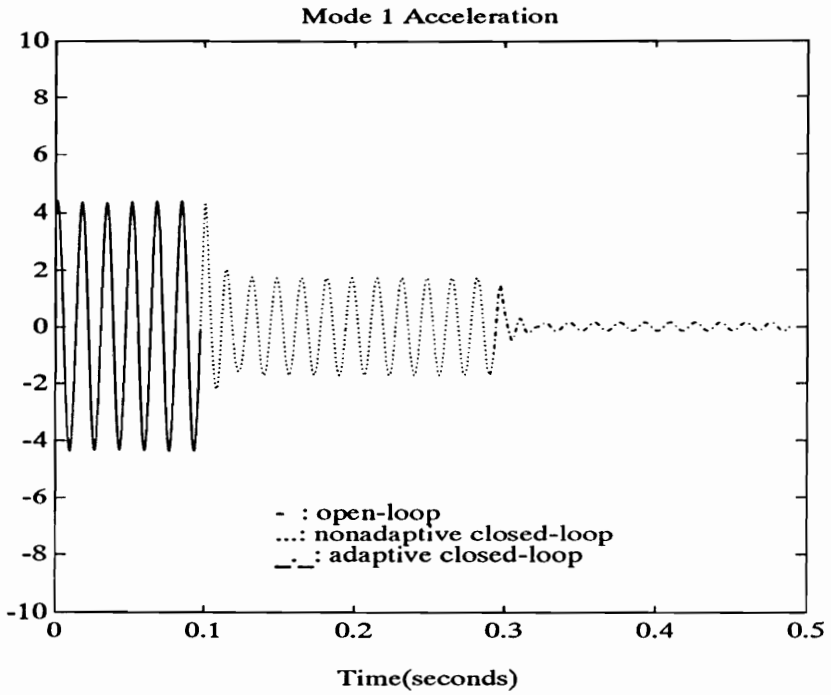


Figure 7.9: Self-Tuning LQG Disturbance Rejection (60 Hz, dL Tuned)

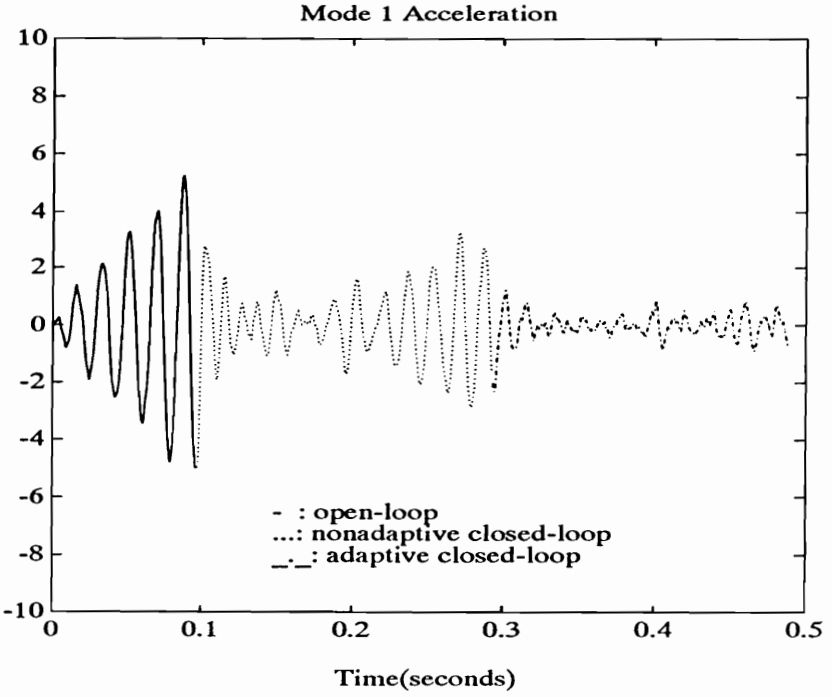


Figure 7.10: Self-Tuning LQG Disturbance Rejection (Narrowband, dL Tuned)

The initial conditions are set to

$$P_{0/0} = P_{0/-1} = 1000 \cdot I$$

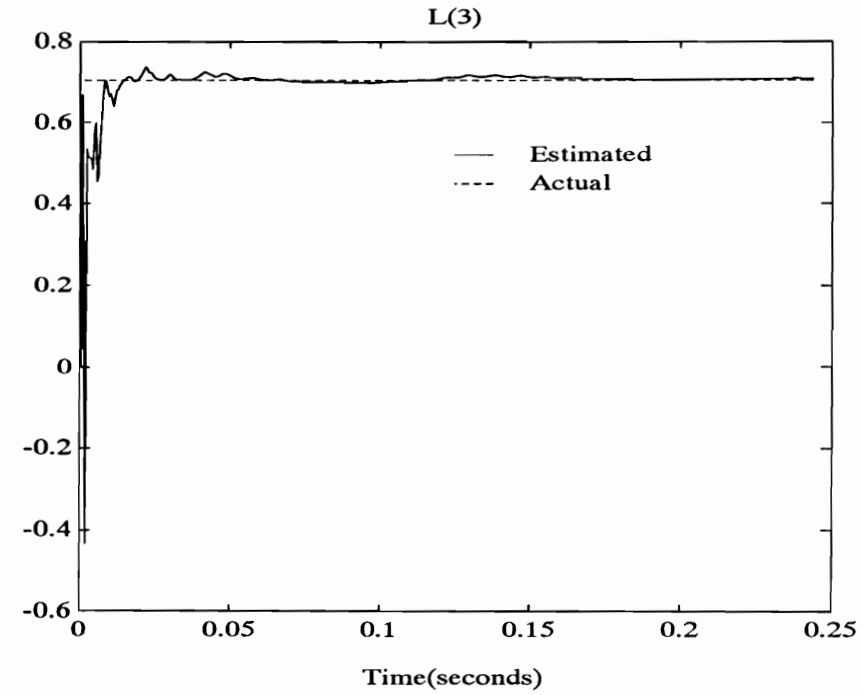
$$\hat{S}_{0/0} = \hat{S}_{0/-1} = \begin{bmatrix} 0_{8\times 1} \\ l_3(0) \\ l_4(0) \end{bmatrix}$$

and $P_{lq,0}$ and $[l_3(0)\ l_4(0)]^{\mathrm{T}}$ are calculated from the originally guessed model.

Table 7.3 lists the results of parameter estimation for a narrow-band disturbance. Figure 7.11 plots the history of estimation. Similarly, the estimate is shown to converge in a few steps and the performance of the controller is improved by better knowledge about the disturbance influence matrix L as can be seen by Figure 7.12, the controlled output(accelerations) of the plate.

Table 7.3 Estimation of C-T Disturbance Influence Matrix(Narrowband Disturbance)

actual L_2	Initial Guess $L_2(0)$	Estimate of L_2 After			
		10	100	300	400
		Iterations			
0.7033	0	0.4842	0.7204	0.7146	0.7043
-0.3062	0	-0.3368	-0.3201	-0.3013	-0.3039



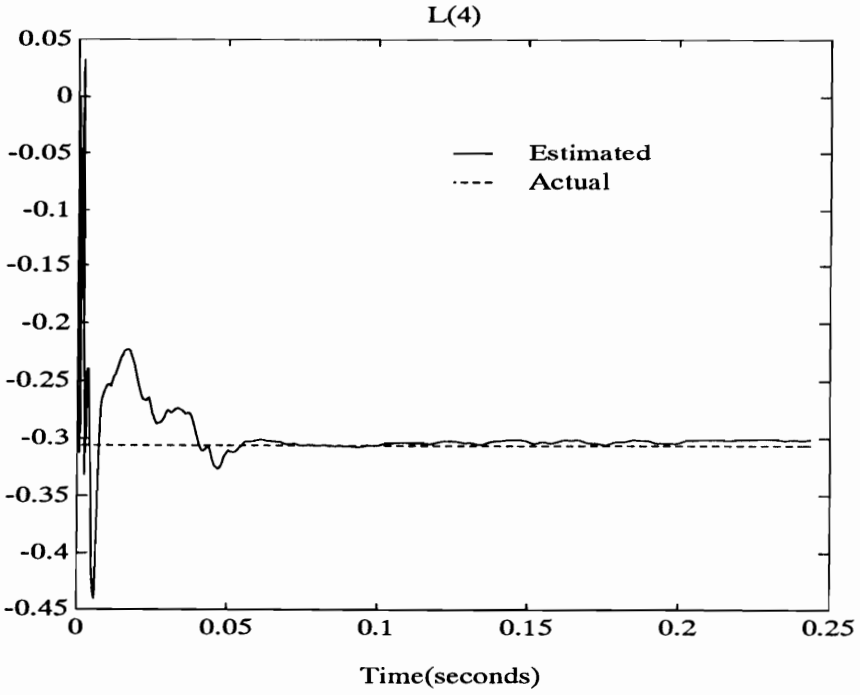


Figure 7.11 Estimation of C-T Disturbance Influence Matrix(Narrowband)

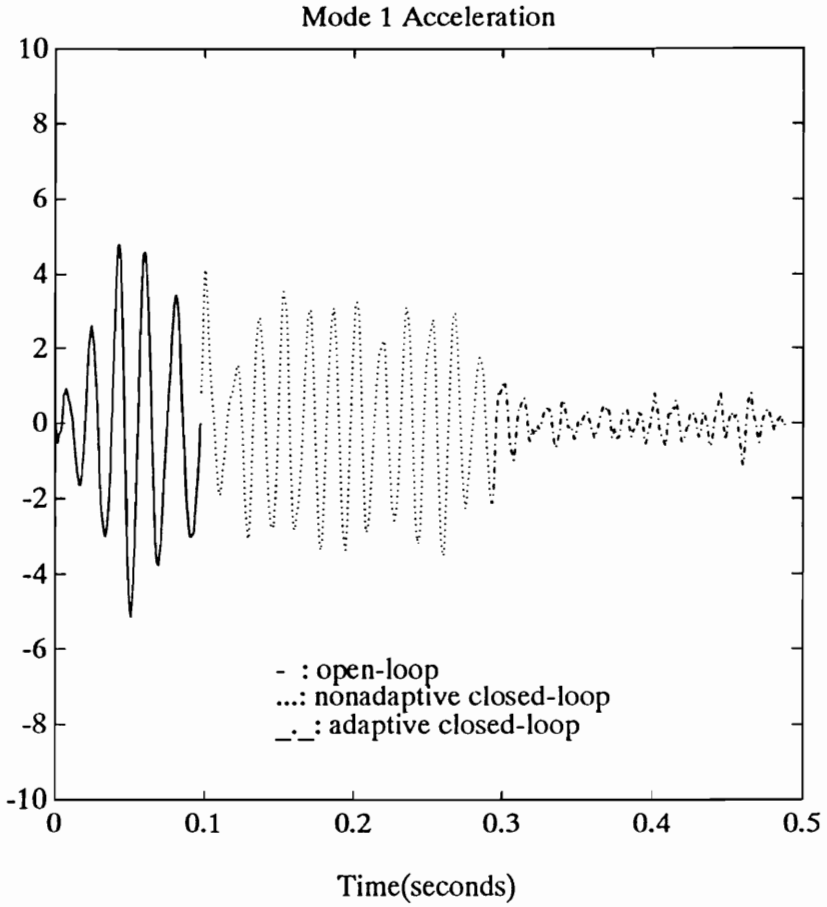


Figure 7.12 Self-Tuning LQG Disturbance Rejection (Narrowband, L Tuned)

8. Conclusions

Self-tuning regulators designed based upon the state-space LQG method demand enormous computation. The need to estimate the parameters of the model, reconstruct the state and evaluate the feedback gains based upon the estimated model parameters all in a single sampling interval makes an LQG self-tuner computationally intensive. As described in the introduction, the objective of this research was to investigate LQG self-tuner design methods that can reduce the computing load.

In this thesis, an LQG self-tuning regulator design method was proposed to be applied to the disturbance rejection problem for flexible structures. It was shown that an LQG regulator for disturbance rejection in a flexible structure is particularly sensitive to only a small number of physical parameters in the continuous-time model. The LQG self-tuner was aimed at selectively tuning the particularly sensitive parameters in a continuous-time model instead of all the parameters in a discrete-time model. This certainly saves a lot of computing effort.

The concept of the extended state model was exploited. The state and the deviations of the parameters were combined to form an extended state and a joint state and parameter estimation problem was formulated. The problem was recognized as a non-linear filtering problem and an extended Kalman filter was used to solve it. Based upon the estimated model parameters, the feedback gains were calculated by iteratively solving the control Riccati equation. In order to save computing effort, only one iteration was performed in a sampling interval. The effectiveness of this design method was experimentally demonstrated on a simply-supported plate. It was shown that an LQG regulator was particularly sensitive to the natural frequency of the 2nd vibration mode. In the experiment, we did not know the actual parameters of the plate. The only knowledge we had about the plate was a model. By assuming an artificial 6% variation on the 2nd natural frequency in the model, a fixed LQG controller would drive the system unstable. With the assumed 6% deviation, the LQG self-tuner was able to obtain

an estimate of the parameter that is close to the parameter in the model while stabilizing the system.

It is a fairly natural thing to include the unknown parameters in the state vector and apply a Kalman filter for the estimation. However, since the system becomes time-varying, significant computation involved in the state error covariance propagation and the Kalman gains update, is still the critical problem in the realization of the LQG self-tuner. A simplified method was developed by fixing part of the error covariance matrix. With this method, the performance of the LQG self-tuner remained as demonstrated experimentally and the computing load was reduced significantly.

The computational method of obtaining a standard quadruple of matrices RI discrete-time model developed by Bingulac and VanLandingham[9] is modified to result in a linear relation between the quadruple of matrices of the continuous-time model and the quadruple of matrices of the discrete-time model. If the disturbance influence matrix in the model of a flexible structure is the only unknown and the disturbance is assumed to be measurable, then the linear relation can be used with an approximate discrete-time model to formulate a joint state and parameter estimation problem such that only a standard Kalman filter is needed for this linear filtering problem. Simulated results showed that the controller is effective if the sampling rate is much higher than the modeled modes and the disturbance frequencies.

There are two conceivable problems associated with the proposed LQG self-tuner. Firstly, the extended Kalman filter is an approximate filter for nonlinear systems and, in general, there is no guarantee on the convergence. Secondly, for large dimension and fast sampling rate systems, the intensive computation involved in the error covariance propagation and Kalman filter gains update is still the major difficulty in realization of the proposed LQG self-tuner.

For the first problem, Ljung[45] proposed a modification of the algorithm, with which global convergence results can be obtained for a general case. The scheme can be categorized as a recursive prediction error algorithm.

The stochastic approximation method for parameter estimation has shown attractive computational features, as in the LMS algorithm. Since there is no actual difference between state estimation and parameter estimation, it would be

computationally attractive if a certain form of stochastic approximation scheme would be included in the joint state and parameter estimation. This is a very promising subject for future study.

References

[1] R. E. Kalman, and R. S. Bucy, "New Results in Linear Filtering and Prediction Theory," *Trans. ASME Ser. D., J. Basic Eng.* 83, December 1961.

[2] M. G. Safonov, and M. A. Athans, "Gain and Phase Margin for Multiloop LQG Regulators," *IEEE Transaction on Automatic Control*, Vol. AC-22, April 1977.

[3] J. C. Doyle, "Guaranteed Margins for LQG Regulators," *IEEE Transaction on Automatic Control*, Vol. AC-23, No. 4, August 1978.

[4] J. C. Doyle, and G. Stein, "Multivariable Feedback Design: Concepts for a Classical/Modern Synthesis," *IEEE Transaction on Automatic Control*, February 1981.

[5] M. Tahk, and J. L. Speyer, "Modeling of Parameter Variations and Asymptotic LQG Synthesis," *IEEE Transaction on Automatic Control*, September 1987.

[6] M. Tahk, and J. L. Speyer, "Parameter Robust Linear-Quadratic-Gaussian Design Synthesis with Flexible Structure Control Applications," *Journal of Guidance, Control and Dynamics*, 1989.

[7] J. Y. Lin, *Robust Design of LQG Controllers*, Ph.D. Dissertation, University of California, Los Angeles, CA, 1989.

[8] S. H. Jones, *Parameter Robust Reduced-Order Control of Flexible Structures*, Ph.D. Dissertation, VPI & SU, Blacksburg, VA, August, 1991.

[9] S. Bingulac, and H. F. VanLandingham, "A Unified Approach To the Discretization and Continualization of MIMO Systems", *Control Theory and*

Advanced Technology(C-TAT), MITA Press, Tokyo, Japan, 1992.

[10] M. J. Balas, "Feedback Control of Flexible Systems," *IEEE Transaction on Automatic Control*, Vol. AC-23, No. 4, August 1978.

[11] K. J. Astrom, *Introduction to Stochastic Control Theory*, Academic Press, New York, NY, 1970.

[12] L. Meirovitch, H. Baruh, and H. Oz, "A Comparison of Control Techniques for Large Flexible Systems," *Journal of Guidance and Control*, Vol. 6, No. 4 July-August 1983.

[13] P. Katz, *Digital Control Using Microprocessors*, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[14] R. F. Stengel, *Stochastic Optimal Control*, John Wiley & Sons, Inc., NY, 1986

[15] S. P. Rubenstein, *An Experiment in State-Space Vibration Control of Steady Disturbances on a Simply-Supported Plate*, M.S. Thesis, Virginia Tech, August 1991.

[16] J. C. Doyle, *Lecture Notes in Advances in Multivariable Control*, ONR/Honeywell Workshop, Minneapolis, MN, 1984.

[17] R. H. Bartels, and G. W. Stewart, "Solution of the Matrix Equation AX+XB=C," *Communications of the ACM*, Vol. 15, No.9, September 1972.

[18] J. C. Doyle, and G. Stein, "Robustness with Observers," *IEEE Transaction on Automatic Control*, Vol. AC-24, No. 4, August, 1979.

[19] L. Ljung, and T. Soderstrom, *Theory and Practices of Recursive Identification*, MIT Press, MA, 1983.

[20] H. W. Sorenson, *Parameter Estimation: Principles and Problems*, Marcel Dekker, INC., NY, 1980.

[21] B. D. Anderson, and J. B. More, *Optimal Filtering*, Prentice-Hall, Englewood Cliffs, NJ, 1979.

[22] A. Gelb, *Applied Optimal Estimation*, MIT Press, MA, 1974.

[23] G. Salut, J. Aguilar-Martin, and S. Lefebvre, "New Results on Optimal Joint Parameter and State Estimation of Linear Stochastic systems," *ASME J. of Dynamic Systems, Measurement and Control*, Vol. 102, March 1980.

[24] H. F. VanLandingham, and M. A. Hopkins, "Deadbeat Parameter Identification of DARMA Process," *Proceedings of 24th Annual Allerton Conference on Communication, Control, and Computing*, Allerton, October 1986.

[25] M. A. Hopkins, *Pseudo-Linear Identification: Optimal Joint Parameter and State Estimation of Linear Stochastic MIMO Systems*, Ph.D Dissertation, VPI & SU, Blacksburg, VA, March 1988.

[26] I. D. Landau, *Adaptive Control: The Model Reference Approach*, Marcel Dekker, New York, 1979.

[27] S. Sastry, and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[28] I. D. Landau, "A Survey of Model Reference Adaptive Techniques(Theory and Applications)," *Automatica*, Vol. 10, pp. 353–379, 1974.

[29] K. J. Astrom, "Theory and Applications of Adaptive Control: A Survey," *Automatica*, Vol. 19, pp.471–486, 1983.

[30] H. P. Whitaker, J. Yamron, and A. Kezer, "Design of model reference

adaptive control systems for aircraft," *MIT Instrument Lab Report*, R-164, 1958.

[31] P. C. Parks, "Lyapunov Redesign of Model Reference Adaptive Control Systems," *IEEE Transaction on Automatic Control*, Vol. AC-11, 1966.

[32] R. E Kalman, "Design of Self-Optimizing Control Systems," *ASME*, Vol. 80, pp. 371–400, 1960.

[33] K. J. Astrom, and B. Wittenmark, "On Self-Tuning Regulators," *Automatica*, Vol. 9, pp. 185–199, 1973.

[34] K. P. Lam, D.Phil. Thesis, Oxford University Engineering Labortary, 1980.

[35] V. Kucera, *Discrete Linear Control – the Polynomial Equation Approach*, Wiley, NY, 1979.

[36] B. Widrow, and S. D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1985.

[37] B. Widrow, D. Shur, and S. Shaffer, "On Adaptive Inverse Control," in *Proceedings of 15th Asilomar Conference on Circuits, Systems and Computers*, 9–11 Nov. 1981, Pacific Grove, CA.

[38] L. J. Eriksson, and M. C. Allie, "A Practical System for Active Attenuation in Ducts," *Sound Vibration* 22 (2), pp.30–34, Feb. 1988.

[39] S. J. Elliot, I. M. Stothers, and L. Billet, "Adaptive Feedforward Control of Flexural Waves Propagating in a Beam," in *Proceedings of the Institute of Acoustics* 12 (1), pp.613–622, 1990.

[40] L. J. Eriksson, "Development of the Filtered-U Algorithm for Active Noise Control," *J. Acoustical Society of American*, 89 (1), January 1991.

[41] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*, 2nd Edition, Addison-Wesley, 1990.

[42] C. Pak, and P. Friedmann, "Transonic Adaptive Flutter Suppression Using Approximate Unsteady Time Domain Aerodynamics," AIAA Paper 91-0986 -CP, *Proceedings of AIAA/ASME/ASCE/AHS/ASC 32nd Structures, Structure Dynamics and Material Conference*, pp. 1832–1846, 1991.

[43] S. Bingulac, and H. F. VanLandingham, "A Unified Approach To the Discretization and Continualization of MIMO Systems," *Control Theory and Advanced Technology(C-TAT)*, MITA Press, Tokyo, Japan, 1992.

[44] S. Bingulac, and D. L. Cooper, "Use of Pseudo-Observability Indices In Identification of Continuous-Time Multivariable Models," *Identification of Continuous-Time Systems(ed. N. Sinha and G. Rao)*, Kluwer Academic Publishers, Netherlands, 1991.

[45] L. Ljnug, "Asymptotic Behavior of the Extended Kalman Filter as a Parameter Estimator for Linear Systems," *IEEE Transaction on Automatic Control*, Vol. AC-24, No. 1, February 1979.

Vita

Fusheng Ho was born October 6, 1961 in Hsinchu, Taiwan, Republic of China. He graduated from National Taiwan University with a B.S. degree in 1983. After 2 years of military service, he came to the United State of America in 1985 for graduate studies. He earned a M.S. degree in Mechanical Engineering from the State University of New York at Buffalo. In 1987, he started his graduate program in the Department of Electrical Engineering at Virginia Polytechnic Institute and State University. He completed the M.S. and Ph.D. degrees in 1988 and 1993, respectively.

Mr. Ho is a member of the Phi Kappa Phi Honor Society and the IEEE Control Systems Society.