Estimation of Nonlinear Systems using Jump Matrix Technique

by

Jastej S. Dhingra

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Electrical Engineering

APPROVED:

R. L. Moose

Dr. R. L. Moose

Dr. H. F. Vanlandingham

Dr. K. B. Yu

July, 1988

Blacksburg, Virginia

Estimation of Nonlinear Systems using Jump Matrix Technique

by

Jastej S. Dhingra

Dr. R. L. Moose

Electrical Engineering

(ABSTRACT)

An estimator for Nonlinear system estimation using Jump Matrix Technique is developed. The idea of the Jump Matrix Technique is to break up the nonlinear system into linear and nonlinear subsystems. The linear subsystem projection, during the sampling interval, is on the basis of the transition matrix of the linear subsystem. At each sampling instant, the system variables are updated using an instantaneous "jump" in the values. The estimator for the nonlinear case is developed on the lines of Kalman filter for the linear case. The Jump Matrix technique is used for system simulation for three different nonlinear systems and then estimation of the system state is carried out using the above mentioned estimator. The system response and the estimated states are considered and the effect of change in system noise parameters is observed. Feedback control of nonlinear system response using the estimated state variables is investigated. Two different systems are considered and their responses are changed using feedback of linear combination of the system state variables.

Acknowledgements

I wish to express my sincere appreciation and thanks to Dr. Richard L. Moose, whose constant encouragement, support and guidance made this research work a pleasurable and enlightening experience. I have benefitted a great deal from working with him and I hope to get an opportunity to work with him again in future.

I would like to thank Dr. H. F. Vanlandingham and Dr. K. B. Yu for their time and help in bringing this thesis to the present form.

I dedicate this thesis to my parents and my family, for without their constant support and encouragement, this work would not have been possible. Finally thanks are due to all my friends here at Virginia Tech who made my stay here a memorable one.

Table of Contents

List of Illustrations

List of Tables

CHAPTER I

INTRODUCTION

1.1 Preview

Nonlinearities are considered to be an inherent part of physical systems. The output state of such systems subject to random inputs is always of great interest to engineers and system analysts. Various techniques have been developed to observe the state of such systems. In many cases the state of such systems is not fully observable. For such cases, state estimation techniques have been developed in which based on non-linear, noise corrupted measurements and a good model of the system, the output state is estimated. In the early sixties, such estimators were developed for linear cases and came to be known as the celebrated Kalman Filters. Later, work was carried out for development of state estimators for non-linear systems. These filters in general were of infinite dimension and quite cumbersome computationally, but with the aid of computers, new techniques like extended Kalman Filters and linearized filter models have been used for estimation of non-linear system state.

A recent technique known as Jump Matrix Technique was developed by Vanlandingham, Moose, Lee [15], Moose, Godiwala [5], for simulation of non-linear systems. Moose, Lauzon [8], used the Jump Matrix Technique to develop an estimator for nonlinear stochastic systems. In this thesis an attempt is being made to further investigate this technique and carry out system simulation and estimation for different non-linear control systems. Analysis of results and comparison with results obtained from other variations in the technique is to be carried out.

1.2 Literature review

State estimation techniques for linear cases have been studied thoroughly and the results are well documented. But for non-linear systems, studies are still going on. In majority of such studies, attempts have been made to somehow linearize or quasi-linearize the system model, so that the well known / documented results of linear systems can be applied to nonlinear system estimation. This approach has been widely used and the results have been quite satisfactory.

Ramnath and Paynter [12], have used the idea of scaling transformation, for linearizing the system model. The basic idea is to find a scaling transformation that will convert a given complex problem into an obviously solvable form. They have given a simple example to convert the nonlinear Riccati equation (1.1)

$$\dot{z} + z^2 = t \qquad [1.1]$$

using variable change $z(t) = \dot{y}(t) \,/\, y(t)$

to obtain $\ddot{y} - ty = 0$

which has a solution using Airy functions [12]. A major drawback with this technique is that it may not be always possible to find a transformation which will lead to a solvable system model. However, an advantage [12] is, that the transformation leads to useful and accurate approximations, and can be studied at an abstract level or used for constructive application procedures.

Statistical linearization is another technique which has been used for non-linear system simulation. Crandall [3], explains, the idea is to replace the non-linear system equation by the equivalent "linear equation" involving a linear parameter. The linear parameter is so selected so as to minimize in some sense, the error response. As an example he has discussed a case of non-linear oscillator of the form $\ddot{x} + 2\alpha\dot{x} + g(x) = f(t)$. The key step in this technique is the evaluation of expectations of certain functions of the non-linear response. This is so because the discrepancy or error can only be measured as mean or variance, so the expectations of functions of system response are part of the expression for evaluation of linear parameter. This method works well for small non-linearities, but for case of large non-linearities the results vary and are not so accurate.

Hedrick and Arslan [6] have used this Statistical Linearization technique for Rail Vehicle Lateral response and Stability analysis. The results are presented in [6]. Beaman and Hedrick [2], go a step further and use series expansion of the unknown probability density function including upto fourth order terms to come up with an Extended Statistical Linearization Technique.

Yavin [16] discusses a version of discrete Kalman filter for the non-linear case. The essential idea is to have a nonlinear transition matrix "A" for the continuous case and then to find the equivalent discrete transition matrix using the first two terms of the

Taylor series expansion of e^{At}. He then proceeds to develop an algorithm on the lines of a linear Kalman filter. The algorithm is used to develop an estimator for a frequency perturbed sine wave oscillator and the results are presented.

Moose, Vanlandingham, Lee [15], Moose, Godiwala [5], have presented a new technique for simulation of non-linear stochastic systems. The basic idea is to decompose the system in to linear continuous time subsystem and nonlinear updates at the sampling instants. Moose and Lauzon [8], went a step further and used this technique for estimation of state of a nonlinear system. Making a simple assumption reduced the non-linear filtering problem to that of an linear Kalman Filter case. They have considered two examples and compared the results with those obtained using Extended Kalman Filters.

1.3 Organization of Thesis.

The aim of this research was to further investigate this technique and if possible develop some kind of analytical reasoning for this technique. In Chapter II the basic control problem and the Jump Matrix Technique application for system simulation and estimation are discussed. Chapter III is devoted to analysis of the technique. Three different systems are considered and for different sets of assumptions and system parameters, the results analyzed. An attempt has been made to present some kind of analytical or intuitive reasoning for system behavior. Chapter IV deals with the case of feedback control problem for case of partially observable open loop systems. The Jump Matrix Estimator is used to estimate the state of partially observable system and then the desired closed loop response is obtained using the estimated state as the feedback. The whole algorithm is first used on a linear system and then a saturating gain is introduced in the system. Finally, Chapter V will be results and conclusions. Some ideas for further research are also presented.

CHAPTER II

JUMP MATRIX TECHNIQUE

2.1 Introduction

Starting with a basic control problem, in this chapter, the Jump Matrix Technique will be discussed. The non-linear system simulation and design of the estimator using Jump Matrix Technique will form a major part of this chapter. The complete development of the technique is given in [8].

Consider a general set of state equations describing a linear control system:

$$\dot{x} = Ax + Bu \qquad [2.1\ a]$$

$$y = Cx + Du \qquad [2.1\ b]$$

The solution to equation (2.1a) is given as

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau. \qquad [2.2]$$

If the system input is sampled at every T seconds and then held constant over the time interval T, equation (2.2) can be written as

$$x(kT+T) = e^{AT}x(kT) + \int_0^T e^{A\lambda}Bd\lambda u(kt). \qquad [2.3]$$

Equations (2.1 a,b) can now be written as

$$x(kT+T) = \phi x(kT) + \Gamma u(kT) \qquad [2.4\,a]$$

$$y(kT) = Cx(kT) + Du(kT) \qquad [2.4\,b]$$

where ϕ and Γ are defined as in (2.5 a,b) and can be found using the power series expansion.

$$\phi = e^{At} = I + AT + A^2T^2/2! \ldots. \qquad [2.5\,a]$$

$$\Gamma = \int_0^T e^{A\lambda}Bd\lambda = [IT + T^2A/2! + T^3A^2/3! + \ldots\ldots]B. \qquad [2.5\,b]$$

Thus equations (2.4 a,b) define the discrete time equivalent of the continuous time system defined in (2.1 a,b), and these equations form the basis of system simulation for this linear time invariant control system. Nonlinear system simulation is more involved because of the presence of non-linearities.

The basis of Jump Matrix technique is to decompose the non-linear system in to linear and non-linear subsystems, using fictitious samplers and clamps. During the sample interval the linear subsystem behaves like an ordinary linear system and the whole analysis, carried out above, holds true. At the sampling instant the nonlinearity output is updated and held constant over the sampling period. One implicit assumption in this analysis is that the nonlinearities are memoryless. Over the sampling period the clamping of nonlinearity output acts as an ideal integration, so the output of each non-linearity is considered to be a state variable. Therefore, the nonlinear system is modelled as linear system during the sample interval with a nonlinear "jump" in sytem state at the sampling instant.

For small sampling interval, this becomes a piecewise constant approximation of the actual non-linearity. In terms of a set of state equations, the system using Jump Matrix Technique becomes

$$x(kT+T)^{-} = \phi x(kT)^{+} + \Gamma u(kT) \qquad [2.6\,a]$$

$$x(kT)^{+} = J(x(kT)^{-}) \qquad [2.6\,b]$$

where J(.) is a non-linear matrix operator, acting on the state vector $x(kT^{-})$, and will be referred to as the Jump Matrix. At the sampling instant only the non-linear state variable is changed instantly, the linear variables are kept constant, which is in direct contrast to keeping the non-linear state variable constant over the sampling period.

As an example consider the system defined by the equation given below.

$$\ddot{x} + \dot{x}^2 x + x^2 \dot{x} = 0$$

Choosing the two nonlinearities as two state variables, the state equations can now be written as

$$x_1 = x \qquad \dot{x}_1 = x_2$$

$$x_2 = \dot{x}_1 \qquad \dot{x}_2 = -x_3 - x_4$$

$$x_3 = x_2^2 x_1 \qquad \dot{x}_3 = 0$$

$$x_4 = x_1^2 x_2 \qquad \dot{x}_4 = 0$$

Written in a matrix form the state equations are :

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} X \qquad kT^+ < t < (k+1)T^-$$

$$X(kT^+) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ & x_2^2 x_1 & & \\ & x_1^2 x_2 & & \end{bmatrix} X(kT^-)$$

2.2 Estimator Design

On the basis of the theory developed in Section 2.1, the nonlinear stochastic system can be modelled as

Nonlinear Jump:	$x_k^+ = J(x_k^-)$	[2.7 a]
Linear Projection:	$x_{k+1}^- = \phi x_k^+ + \Gamma u_k + \psi w_k$	[2.7 b]
Measurement:	$z_k = H x_k^+ + v_k$	[2.7 c]

The system noise w_k and measurement noise v_k are statistically independent, uncorrelated, normal random noises with covariance matrices Q and R respectively.

Intuitively, the estimated state just after the sampling instant should be proportional to the non-linear update of state before the sampling instant and also the measurement made at the sampling instant. So it is assumed that the filter will take the following form:

$$\hat{x}_k^+ = \Theta J(\hat{x}_k^-) + \Delta z_k^+ \qquad [2.8]$$

Over the sampling interval it just behaves as an ordinary linear estimator filter and the intersample response is given as

$$\hat{x}_{k+1}^- = \phi \hat{x}_k^+ + \Gamma u_k \qquad [2.9]$$

The filter error is given as :

$$e_{k+1} = x_{k+1}^+ - \hat{x}_{k+1}^+ \qquad [2.10]$$

Substituiting in the values from equations (2.7) and (2.8),

$$e_{k+1} = J(x_{k+1}^-) - \Theta J(\hat{x}_{k+1}^-) - \Delta(HJ(x_{k+1}^-) + v_{k+1}) \qquad [2.11]$$

Taking the statistical expectation

$$E[e_{k+1}] = (I - \Delta H)E[J(x_{k+1}^-) - (I - \Delta H)^{-1}\Theta J(\hat{x}_{k+1}^-)] - \Delta E[v_{k+1}] \qquad [2.12]$$

Since $E[v_{k+1}] = 0$ and for an unbiased estimator (i.e. as k $\rightarrow \infty$, $\hat{x}_k \rightarrow x_k$ and $E[e_{k+1}] \rightarrow$ 0), Θ should be equal to (I - Δ H). Thus equation (2.12) reduces to

$$E[e_{k+1}] = (I - \Delta H)E[J(x_{k+1}^-) - J(\hat{x}_{k+1}^-)] \qquad [2.13]$$

The problem now is to evaluate the statistical expectation of the term $[J(x_{k+1}^-) - J(\hat{x}_{k+1}^-)]$. Since J(.) is a non-linear operator, it cannot be exchanged with the expectation operator. Lauzon, Moose [8], used the idea of a time varying matrix a_{k+1} such that

$$[J(x_{k+1}^-) - J(\hat{x}_{k+1}^-)] = a_{k+1}[x_{k+1}^- - \hat{x}_{k+1}^-] \qquad [2.14]$$

The validity of the assumption will be discussed later. Proceeding forward with filter development, equation (2.13) and (2.14) leads to

$$E[e_{k+1}] = (I - \Delta H)a_{k+1}\phi E[e_k] \qquad [2.15]$$

Using equation (2.11) and (2.14) the error expression can now be written as

$$e_{k+1} = (I - \Delta H)a_{k+1}\phi e_k + (I - \Delta H)a_{k+1}\psi w_k - \Delta v_{k+1} \qquad [2.16]$$

Using equation (2.16) the Error Covariance matrix P_{k+1}, defined as $E[e_{k+1}e_{k+1}^T]$ is found out to be

$$P_{k+1} = (I - \Delta H)[a_{k+1}(\phi P_k \phi^T + \psi Q \psi^T)a_{k+1}^T](I - H^T\Delta^T) + \Delta R \Delta^T \qquad [2.17]$$

Defining

$$M_{k+1}^- = \phi P_k \phi^T + \psi Q \psi^T \qquad [2.18]$$

and

$$M_{k+1}^+ = a_{k+1} M_{k+1}^- a_{k+1}^T \qquad [2.19]$$

Equation (2.17) can now be written as

$$P_{k+1} = (I - \Delta H)M_{k+1}^+(I - H^T\Delta^T) + \Delta R \Delta^T \qquad [2.20]$$

In order to determine the filter gain, the trace of error covariance matrix P_{k+1} is minimized.

Defining C = tr(P_{k+1}) = $E[e_{k+1}^T e_{k+1}]$ and differentiating C with respect to Δ and setting it equal to zero leads to

$$\frac{\delta C_{k+1}}{\delta \Delta} = -2(HM_{k+1}^{+}H^T) + 2\Delta HM_{k+1}^{+}H^T + 2\Delta R = 0. \qquad [2.21]$$

Solving for Δ yields

$$\Delta = M_{k+1}^{+}H^T[HM_{k+1}^{+}H^T + R]^{-1} \qquad [2.22]$$

This expression looks very similar to that of Kalman gain Vector for the case of linear Kalman Filter design. So defining Δ as the time varying filter gain K_{k+1} , P_{k+1} can be written in the recursive form as

$$P_{k+1} = (I - K_{k+1}H)M_{k+1}^{+}(I - K_{k+1}H)^T + K_{k+1}RK_{k+1}^T \qquad [2.23]$$

Substituiting in value of K_{k+1} in equation (2.23) yields

$$P_{k+1} = (I - K_{k+1}H)M_{k+1}^{+} \qquad [2.24]$$

This completes the specifications for the estimator. Summarizing the filter equations in a conventional set of equations :

Linear Prediction: $\hat{x}_{k+1}^{-} = \phi\hat{x}_k^{+} + \Gamma u_k.$ [2.25 a]

Nonlinear Update: $\hat{x}_{k+1}^{+} = J(\hat{x}_{k+1}^{-}) + K_{k+1}(z_{k+1}^{+} - HJ(\hat{x}_{k+1}^{-}))$ [2.25 b]

$$M_{k+1}^{+} = a_{k+1}[\phi P_k \phi^T + \psi Q \psi^T] a_{k+1}^T \qquad [2.25\ c]$$

Gain Calculation: $$K_{k+1} = M_{k+1}^T H^T [H M_{k+1}^T H^T + R]^{-1} \qquad [2.25\ d]$$

Covariance Update: $$P_{k+1} = (I - K_{k+1} H) M_{k+1}^{+}. \qquad [2.25\ e]$$

So starting with an initial value of the Covariance Matrix P_0 and time varying gain matrix a_{k+1}, the Kalman gain vector K_{k+1} is calculated and the state estimated. The flow chart of the algorithm is shown in Fig. 1.

All along during the development of this Kalman Filter for nonlinear systems, the validity of the assumption in equation (2.14) was not discussed. For calculation of the time-varying gain matrix a_k, the basic equation is

$$[J(x_{k+1}^{-}) - J(\hat{x}_{k+1}^{-})] = a_{k+1}[x_{k+1}^{-} - \hat{x}_{k+1}^{-}] \qquad [2.26]$$

In a practical sense, the actual state is never known, therefore it is not possible to calculate the gain matrix a_{k+1} on-line. For the purpose of digital simulation and estimation, the state is actually calculated and so for theoretical analysis the gain matrix a_{k+1} can be calculated. Lauzon, Moose [8], have suggested three different ways of calculating this gain matrix. The significance of all the three cases along with some intuitive mathematical reasoning is as follows.

The first alternative is to let the gain matrix be an Identity matrix I. This assumption cuts down drastically the computation time. But the inherent assumption is that the Non-linear Jump operator is nothing but a linear amplifier with unity gain. This as-

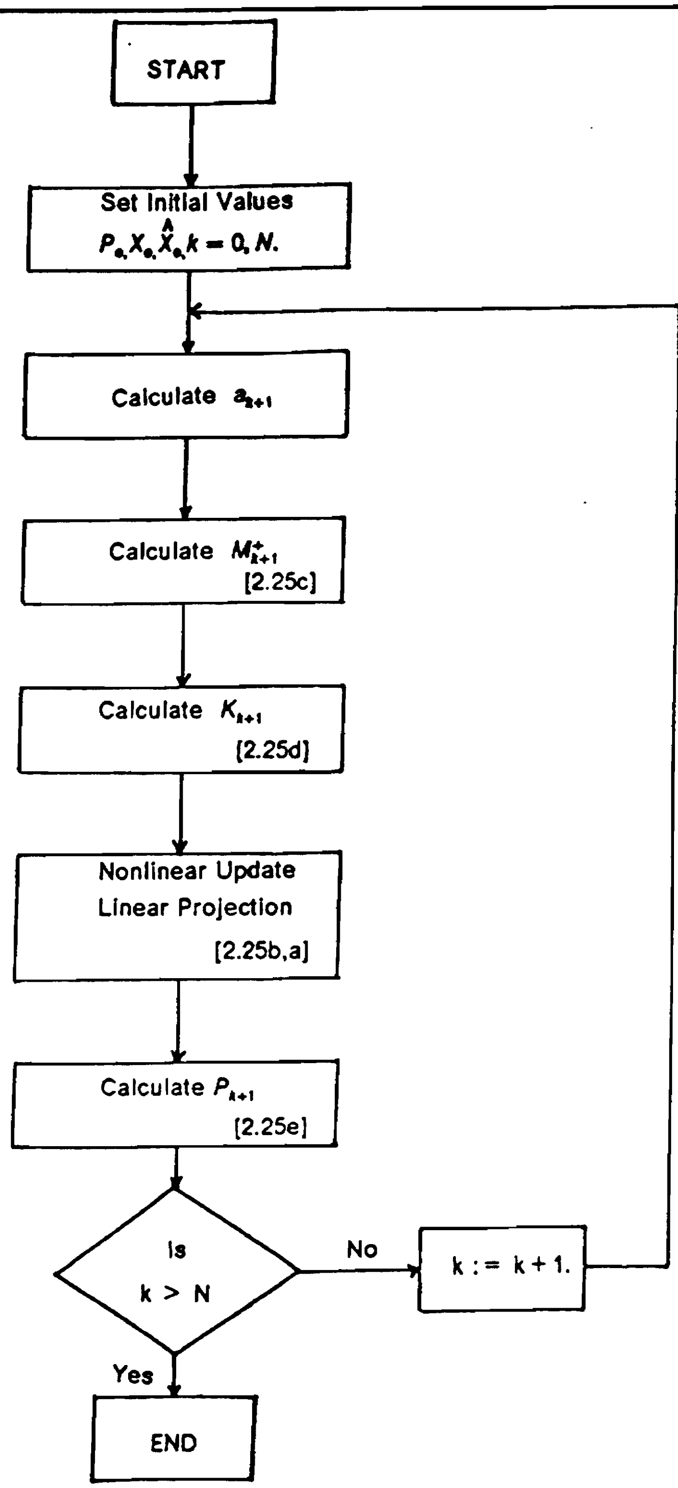

Figure 1. Flow chart for Estimator Algorithm.

sumption simplifies the whole algorithm and the Kalman gains can be calculated off-line and the algorithm is the same as that of a linear Kalman filter. The disadvantage is that it linearizes the non-linearity and also sometimes suppresses the cross-relationship between state-variables and can lead to some vague results. For example if the Jump matrix is of the form

$$J \equiv \begin{bmatrix} 1 & 0 & 0 \\ & f(x_k) = x_1^2 x_3 & \\ 0 & 0 & 1 \end{bmatrix} \qquad [2.27]$$

Making $a_k = I$ leads to equalizing

$$x_1^2 x^3 - \hat{x}_1^2 \hat{x}_3 \equiv x_2 - \hat{x}_2 \qquad [2.28]$$

which doesnot justify the validity of this assumption.

So this assumption holds good to some extent, that is, only if the nonlinearity involves one state variable. Summarizing, computational time is reduced considerably, but the estimator so developed does not track the state so well and the estimator error is large.

The second case is that of using the online gain calculation i.e.

$$J(x_k) - J(\hat{x}_k) = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} (x_{k,1} - \hat{x}_{k,1}) \\ (x_{k,2} - \hat{x}_{k,2}) \\ (x_{k,3} - \hat{x}_{k,3}) \end{bmatrix} \qquad [2.29]$$

There are nine variables a_{ij}: i,j = 1..3, but there are only three equations, so the system of equations does not have any solution. To simplify the problem, all off-diagonal terms are assumed to be zero. For calculation of a_{ii} i = 1..3, the ratio of the actual differences is used i.e. if $J(x)$ is denoted as x_{+i} and $J(\hat{x})$ as $\hat{x}_{+i}$ then

$$a_{ii} = \frac{x_{+i} - \hat{x}_{+i}}{x_i - \hat{x}_i} \qquad [2.30]$$

Here again the actual state vector x_k and J(x_k) are not known. So the validity of this alternative is also questionable. But again, for the purpose of theoretical analysis the state vector x_k is known from the simulation and hence this alternative can be used. This alternative is a slight better case than the first one, because here the unity gain assumption has been replaced by a time varying gain as defined in equation (2.30). But there exists the same disadvantage as in the previous case, namely that of suppression of cross terms between the state variables and application of this technique to inappropriate systems leads to vague results.

The third alternative is to use the Jacobian of J(.), evaluated at some mean value point $x_k^{\bullet}$, as the time varying matrix a_k. This alternative holds only if J(.) is differentiable and the vector mean value theorem can be applied to it. The mean value point $x_k^{\bullet}$ lies in the closed interval $[x_{k+1}^-, \hat{x}_{k+1}^-]$. Here again, the actual state is not known and thus the closed interval can not be defined. This is circumvented by the fact that as the estimator is unbiased therefore as $k \rightarrow \infty$, $x_{k+1}^- \rightarrow \hat{x}_{k+1}^-$ and the closed interval $[x_{k+1}^-, \hat{x}_{k+1}^-]$ becomes extremely small and $x_{k+1}^{\bullet}$ may be approximated by the estimate $\hat{x}_{k+1}^-$.

So this alternative, allows one to use, only the estimated state and does not require the actual state x_k for calculation of a_{k+1}. As an example, consider a second order system with one nonlinearity. So in all, there are three state variables - first two corressponding to linear system variables and the third being the output of the non-linearity. The Jump matrix can be modelled as :

$$J \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & f(.) & \end{bmatrix} \qquad [2.31]$$

here f(.) is the nonlinear function of state vector x_k. If f(.) is differentiable with respect to x_k then

$$J(x_k) - J(\hat{x}_k) = \begin{bmatrix} (x_{k,1} - \hat{x}_{k,1}) \\ (x_{k,2} - \hat{x}_{k,2}) \\ (f(x_k) - f(\hat{x}_k)) \end{bmatrix} = a_{k+1} \begin{bmatrix} (x_{k,1} - \hat{x}_{k,1}) \\ (x_{k,2} - \hat{x}_{k,2}) \\ (x_{k,3} - \hat{x}_{k,3}) \end{bmatrix} \qquad [2.32]$$

Solving for $a_{i,j}$ i,j = 1..3, yields

$$a_{k+1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad [2.33]$$

where $\quad a_{3i} = \frac{\delta f}{\delta x_i} \mid_{x_i = x_i^*}$

The question now is - What if f(.) is not differentiable ?

For the case of a piecewise linear f(.), the partial derivative can still be used provided x^* lies on the linear region of the function f(.). If x^* lies on a discontinuity then an intermediate value i.e. some mean value between the right side limit and left side limit of f(x) at $x = x^{*}$ can be used for purpose of gain calculations. As an example, consider f(.) of the form :

$$f(x_k) = \begin{bmatrix} 1.0 & x_{k,1} > 0.1 \\ 10x_{k,1} & |x_{k,1}| < 0.1 \\ -1.0 & x_{k,1} < -0.1 \end{bmatrix} \qquad [2.34]$$

Here a_{32} , a_{33} are both zero and a_{31} can be defined as

$$a_{31} = \begin{bmatrix} 0.0 & |x_{k,1}| > 0.1 \\ 10.0 & |x_{k,1}| < 0.1 \\ 5.0 \text{ or } 0 < a_{31} < 10 & |x| = 0.1 \end{bmatrix} \qquad [2.35]$$

The third alternative seems to be the most useful in practical sense as it doesnot require the actual state to be known for the purpose of calculation of a_{k+1}. Moreover the cross-relationship between state variables is maintained while taking the Jacobian. The first two alternatives are also useful in theoretical analyis and can be used as an aid to determine ideal filter performance.

In Chapter 3, the estimator designed above will be used to estimate the state for three different systems. The results using the last two alternatives discussed above will be presented and comparison between the results using the two techniques will be made.

CHAPTER III

ANALYSIS OF THE ESTIMATOR

3.1 Introduction

In Chapter II, the simulation of non-linear systems using the Jump Matrix Technique was discussed and then the same technique was used to develop an estimator for non-linear system state estimation. In this chapter, three nonlinear systems are being considered for the purpose of analysis of the estimator. The systems were simulated and the system state estimation was done on the basis of the theory developed in the previous chapter. The results of simulation and estimation are presented towards the end of each section and engineering intuitive reasoning for system results is presented. In each case, system analysis is carried out using the last two alternatives discussed in the previous chapter. Comparison between the two approaches is made using the mean square error at the output as the comparison criteria.

3.2 Non Linear Spring

The first nonlinear system studied is that of a nonlinear spring. The system behaviour is well documented in [4]. Lauzon [8] estimated the state of this system but the results left certain questions unanswered. A comparison was made between the performance of an Extended Kalman Filter and that of an estimator developed using the Jump Matrix Technique. Analysis of the estimator on its own was not done. This system was chosen for further investigation, to use it to analyze the technique further and also because it helped in getting started with the estimator, before other new systems could be handled using the Jump Matrix Technique.

The nonlinear differential equation governing the response of the Non linear spring with a stochastic input w(t) is

$$\ddot{x} + \mu\dot{x} + \omega_o^2 x + hx^3 = w(t) \qquad [3.1]$$

where

ω_o : natural frequency of oscillation

h : spring constant (h < 0 : soft; h > 0 : hard)

μ : damping factor

choosing $\quad x_1 = x \qquad \dot{x}_1 = x_2$

$x_2 = \dot{x}_1 \qquad \dot{x}_2 = -\,\omega_o^2 x_1 - \mu x_2 - hx_1^3 + w(t)$

$x_3 = x_1^3 \qquad \dot{x}_3 = 0$

The state equations can now be written in the matrix form as

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 \\ -\omega_o^2 & -\mu & -h \\ 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} w(t) \qquad kT^+ < t < (k+1)T^- \qquad [3.2a]$$

$$X(kT^+) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & x_1^3 & \end{bmatrix} X(kT^-) \qquad [3.2b]$$

The block diagram for the system is given in Fig 2.

The system simulation was carried out for T = 0.01 sec, μ = 0.1, ω_o^2 = 2.00. The spring constant 'h' was assigned three different values -0.5, 0.0, 1.0 for the three different cases of soft, linear and hard spring response respectively. Figs 6, 7, 8, show the system state 'x', estimated state '$\hat{x}$' and the measurement 'z' from which the system state was estimated. The three figures correspond to the three different values of 'h' respectively. The system noise and measurement noise variances were taken to be 1.0 and 4.0 respectively. Also for these figures, for calculation of the time varying gain matrix a_{k+1} , the ratio of actual differences involved was used i.e. the second alternative discussed in the previous chapter was used. This approach is later referred to as the 'online gain calculation' approach.

Figs 9, 10, 11, depict the system responses for the same three values of 'h' discussed earlier. The difference from the previous case being that the third alternative, of taking the Jacobian of J(.), was used to find the time varying gain matrix a_{k+1}. This approach is later referred to as the 'Jacobian' approach.

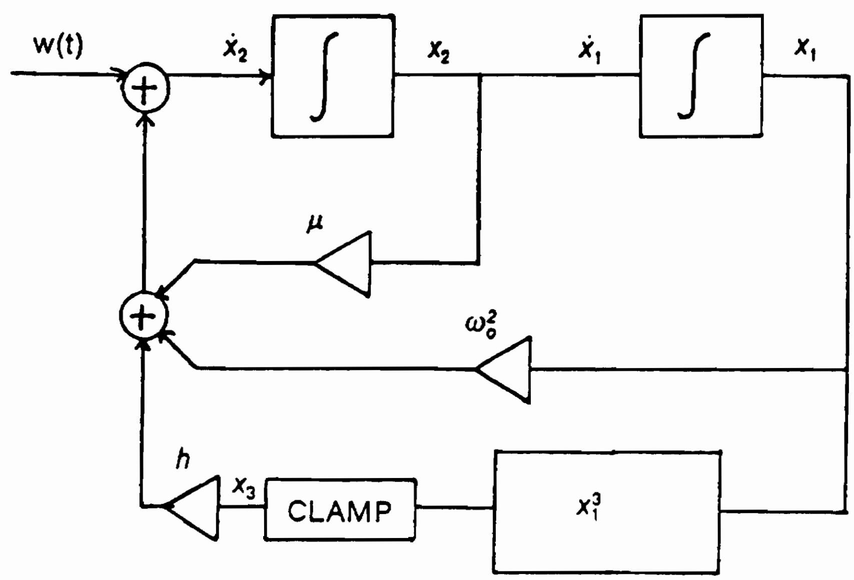

Figure 2. Block Diagram for Nonlinear Spring.

System simulation and estimation of state for the case of non-linear spring was carried out for various combinations of input noise variance, measurement noise variance and spring constant 'h'. The idea was to investigate the error in estimated state and to come up with some reason to explain the variation of mean square error of the output with respect to the parameters discussed above. Also comparison was carried out between the online gain approach and Jacobian approach with the mean square error as the criterion.

All along this thesis, the mean square error at the output refers to the time average of the error between the actual and the estimated state. That is, the mean square error defined as $E[(x_{k,1} - \hat{x}_{k,1})^2]$ averaged over the time 'k'. The values were obtained using 1000 iterations and reflects the total mean square error (starting from k = 0 to 1000).

Table 1 shows values of mean square error for various possible combinations of σ_v^2 and spring constant 'h'. All along these calculations, σ_w^2 and sampling interval T were kept constant at 1.0 and 0.01 sec respectively and all these observations were made using the online gain approach. Table 2 presents the mean square error variation for the same combinations as above, the difference being that the Jacobian approach was used.

Tables 3,4 depict the variation in mean square error with respect to σ_w^2 and spring constant 'h'. σ_v^2 and T were kept constant at 4.0 and 0.01 secs respectively. On examining the values in Table 1, for a constant 'h' there is not much difference in the mean square error for various values of σ_v^2. Obviously as σ_v^2 increases there is some increase in output mean square error, but the increase is not significant, implying that the mean square error at the output depends only on 'h', for fixed values of σ_w^2, T, and a reasonable value of σ_v^2. This can be explained by the fact that as $\hat{x} \rightarrow x$, the terms in a_{k+1} corressponding to the non-linear functions in the system model, approach zero and thus the kalman gains according to equations [2.25] become exceptionally small hence reducing dependence on measurement 'z' and causes the estimate $\hat{x}$ to track the system state x closely and results in low mean square error.

Table 2 presents another interesting observation. For a fixed value of spring constant 'h', as σ_v^2 increases the mean square error of output also shows an increase. This is different from what was observed in the previous case. Here the gain terms in 'a_{k+1}' are calculated using Jacobian evaluated at $x^* = \hat{x}$. Thus the gain terms don't tend to zero unless or until the actual derivative goes to zero. The steady state Kalman gains are greater than in the previous case and thus more dependence of $\hat{x}$ on 'z' which explains the different behaviour from that of a previous case.

Another interesting observation is that the mean square error using Jacobian approach is greater than that in case of using the online approach. In Jacobian case only $\hat{x}$ is used for estimator design and the gain terms in ' a_{k+1}' are derivatives evaluated at $\hat{x}$ and thus the noise plays a major role in output mean square error. In case of online gain approach the differences used for calculation of a_{k+1} cancel out much of the noise and moreover the steady state kalman gains are of smaller value which reduce dependence on measurement z. Thus it is a positive feedback case and at each iteration reduces the output error till a steady state is reached.

The values for the mean square error for h = 0.0 in Tables 1 and 2 are different. As they represent the same linear spring the values should have been the same. The reason lies in the modelling of the system. That is the hardness factor is included in the transition matrix ϕ, so the matrix a_{k+1} obtained using the two approaches are different and are not dependent on the value of 'h'. So for same transition matrix ϕ and different a_{k+1} for the two approaches the results will obviously be different.

Comparing Tables 3 and 4, both cases show a similar trend of increase in output mean square error as σ_w^2 increases for fixed value of spring constant 'h'. The measurement noise σ_v^2 and sampling interval T were kept constant at 4.0 and 0.01 respectively. From equations [2.25] it is obvious that M_{k+1}^+ is directly proportional to Q (or σ_w^2) and thus any increase in σ_w^2 results in proportional increase in output mean square error. The mean square errors for the case of online gain approach are still small compared to Jacobian approach. The reasoning discussed earlier for Tables 1 and 2 i.e. small Kalman gain values still holds here and is the cause for low mean square error at the output. The errors for case of h = -0.5 are higher than for other two values of h. The value h = -0.5 implies a soft spring and has a slow (sluggish) response and thus the estimator takes more time to respond to change in actual state. This is evident in Figs 6 and 9.

The system simulation and estimation was carried out using the same Jump Matrix Technique. So to check the validity of the estimator a mathematical subroutine was used to simulate the system response for a particular set of system parameter values and then the estimator was used to estimate the system response. Fig.12. shows the system state generated using the IMSL subroutine, the measurement, and the estimated state. Initially the estimator does not track the system state properly but as time progresses and the gains reach a steady state the tracking is excellent and it becomes difficult to distinguish between the estimated and actual state (Fig.12.).

3.3 Van Der Poll's Equation

The second example to be considered is the classic Van Der Poll's equation. Solution of many non-linear equations is found by converting them into a form of Van Der Poll's equation by using time scaling, normalizing, scaling transformation etc. The solution to this equation using phase plane trajectories is discussed in [4].

The basic Van Der Poll's equation with a stochastic input can be written in a form :

$$\ddot{x} - \varepsilon(1 - x^2)\dot{x} + 9x = w(t). \qquad [3.3]$$

with a set of initial conditions $x(0) = 5$, $\dot{x}(0) = 0$.

Setting up in state equation form

Let

$$x_1 = x \qquad \dot{x}_1 = x_2$$

$$x_2 = \dot{x}_1 \qquad \dot{x}_2 = -9x_1 + \varepsilon x_2 - x_3 + w(t)$$

$$x_3 = -\varepsilon x_1^2 x_2 \qquad \dot{x}_3 = 0$$

Writing it in a matrix form

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 \\ -9 & \varepsilon & 1 \\ 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} w(t) \qquad kT^{+} < t < (k+1)T^{-} \qquad [3.4a]$$

$$X(kT^{+}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & -\varepsilon x_1^2 x_2 & \end{bmatrix} X(kT^{-}) \qquad [3.4b]$$

The block diagram for the system is given in Fig 3.

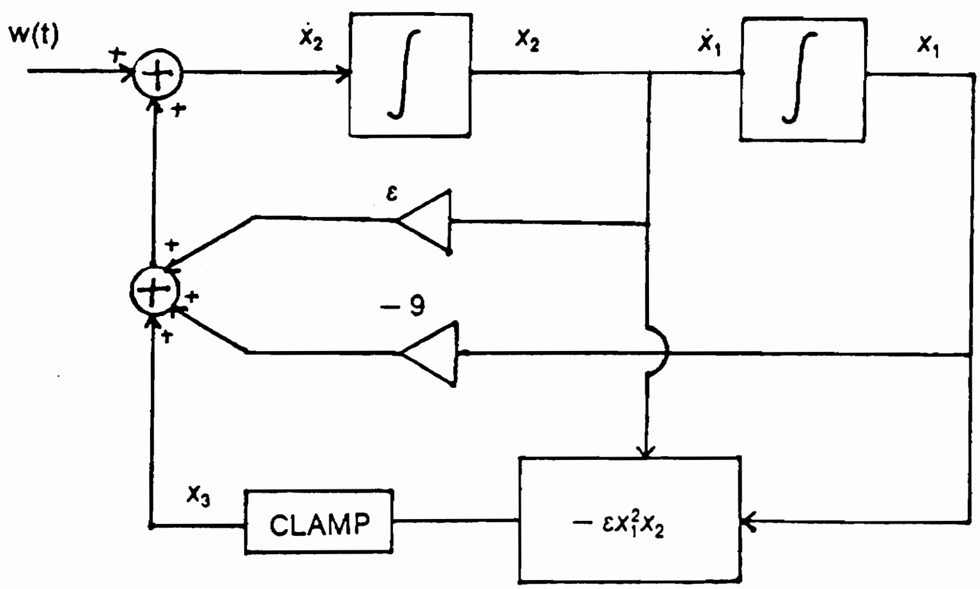

Figure 3. Block Diagram for Van Der Poll's equation.

The system was simulated for three different values of ε i.e. 0.5, 2.0 and 5.0. The parameter 'ε' is a kind of skewing factor and as can be seen from the output response in Figs. 13, 14, and 15, the extent of skewness is different in all three cases. The system was estimated for all the three values of ε. Figs 13, 14, 15, depict system response, estimated state and measurement for ε = 0.5, 2.0 and 5.0 respectively. The online gain approach was used for calculation of a_{k+1} Figs 13, 14 and 15 show the same three cases as earlier but using the Jacobian approach for calculation of a_{k+1}.

For the case of Van Der Poll's equation, the system simulation and state estimation was carried out for various values of σ_v^2, σ_w^2 and skewness factor ε. Tables 5 and 6 show the values of mean square error at output for different values of σ_v^2 and ε. Table 5 represents the simulation and estimation carried out using online gain approach while Table 6 was obtained using the Jacobian approach. The same values of ε i.e. 0.5, 2.0, 5.0 were used with σ_v^2 varying from 1.0 to 100.0 with intermediate values of 10.0 and 25.0. All along σ_w^2 and sampling interval T were kept constant at 10.0 and 0.01 respectively.

The figures in these tables present a similar picture to that in case of non-linear spring. For online gain approach, the mean square error at the output was of the same order for all values of σ_v^2 if ε was kept fixed. Increase in ε brought about an increase in the order of magnitude of the output mean square error. For the Jacobian approach case, the values suggest an increase in the output mean square error, as the measurement noise variance σ_v^2 is increased. This was also seen in the case of Non-linear spring discusssed earlier. The reasons here are also the same as previously, namely that $\hat{x}_k \rightarrow x_k$ as $k \rightarrow \infty$ which leads to low values in the corresponding a_{k+1}. This in turn leads to low values in M_{k+1} and hence the low Kalman gains reducing the dependence of the estimated state on the measurement process. The dependence of the output mean square error on σ_v^2 is thus negligible because of the direct dependence of K_{k+1} on M_{k+1} . In case of Jacobian approach no such problem exists.

Tables 7 and 8 represent the case of fixed σ_w^2 versus the same three values of the skewness factor ε discussed earlier with σ_v^2 being kept constant. The system noise variance σ_w^2 is varied and takes up the three values 1.0, 2.0 and 5.0. The results here have the same trend as seen in the case of Nonlinear spring, that is as σ_w^2 increases the output mean square error also increases almost proportionally and also the mean square error at the output has larger value for the Jacobian approach case (Table 8.) than in online gain approach (Table 7.). The direct dependence of M_{k+1}^- on σ_w^2 is again the reason for such a behaviour.

3.4 Relay Control System.

The next example considered here is a Nonlinear Relay Control System. Kuo [7] has provided an analysis for calculation of unit step response using phase plane trajectories. Here in this section a stochastic model is considered, with a nonlinear saturating amplifier followed by a second order Low Pass filter with negative feedback completing the loop. The block diagram for simulation of such a system is given in Fig.4.

The system is modelled as follows :

Let $\quad x_1 = y \qquad \dot{x}_1 = x_2$

$$x_2 = \dot{x}_1 \qquad \dot{x}_2 = x_3 - x_2$$

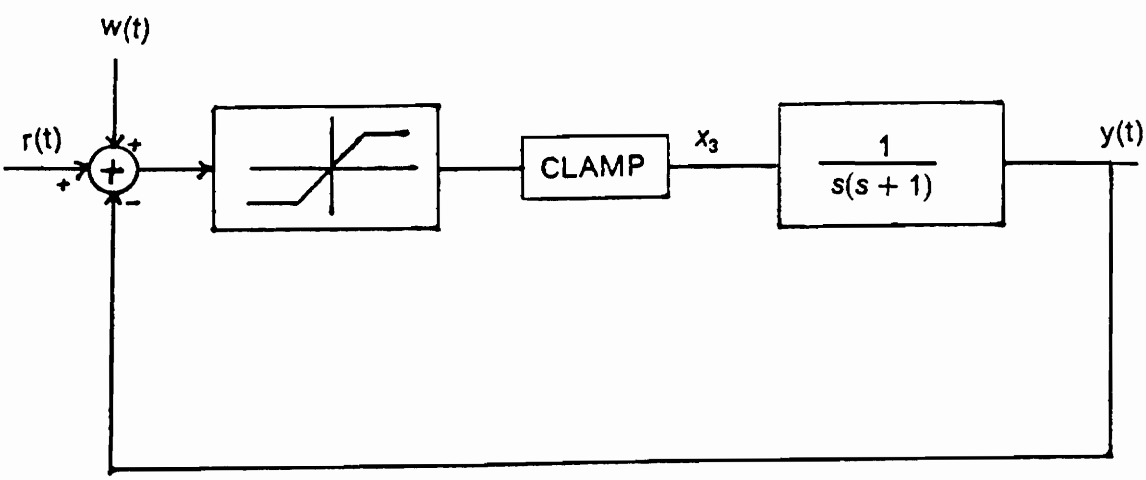

Figure 4. Block Diagram for Relay Control System.

$$x_3 = f(r - x_1 + w_k) = 0$$

The state equations can now be set up in the matrix form as

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} X \qquad kT^+ < t < (k+1)T^- \qquad [3.3a]$$

$$X(kT^+) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & f(r - x_1^- + w_k) & \end{bmatrix} X(kT^-) \qquad [3.3b]$$

where

$$f(a) = \begin{bmatrix} 1.0 & a > 0.1 \\ 10.0a & |a| < 0.1 \\ -1.0 & a < -0.1 \end{bmatrix}$$

The system simulation and estimation was carried out for T=0.01 sec, $\sigma_w^2 = 0.1$ and $\sigma_v^2 = 1.0$. The slope of the saturating amplifier was kept at 10.0. Figs 19 and 20 show the system state, measurement process and system state estimate for the two cases of online gain approach and Jacobian approach respectively. As is evident from the two figures, the error for the Jacobian gain case is greater than the error for the case of online gain approach. The results match with the expectations on the basis of investigation of the previous two cases.

A slightly modified relay control system was also considered. Initially the noise was being applied at the input of the system but in this case the noise was introduced in the low pass filter. The second order low pass filter was divided into two single order filters in cascade and the noise introduced as part of input to the second filter. The system configuration is shown in Fig.5.

The zero mean white gaussian measurement noise v(t) is added to the system output y(t) to obtain the measurement process z(t). This is not shown in the system block diagrams. The measurement process is then used as an input for the estimator to estimate the system state variables.

The system can be modelled as follows :

Let $x_1 = y \qquad \dot{x}_1 = x_2 - x_1 + w(t)$

$$\dot{x}_2 = x_3$$

$$x_3 = f(r - x_1) \qquad \dot{x}_3 = 0$$

Setting up the state equations in a matrix form

$$\dot{X} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} w(t) \qquad kT^+ < t < (k+1)T^- \qquad [3.3a]$$

$$X(kT^+) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & f(r - x_1) & \end{bmatrix} X(kT^-) \qquad [3.3b]$$

where f(.) is the same as discussed earlier.

Figures 21 and 22 depict the system state, measurement, and the estimated state for the case of Modified Relay control system. Both approaches for calculation of the gain matrix a_{k+1} were used and as is evident from Figs. 21 and 22, the Jacobian case approach leads to a noisier estimator as was the case in all previous examples con-

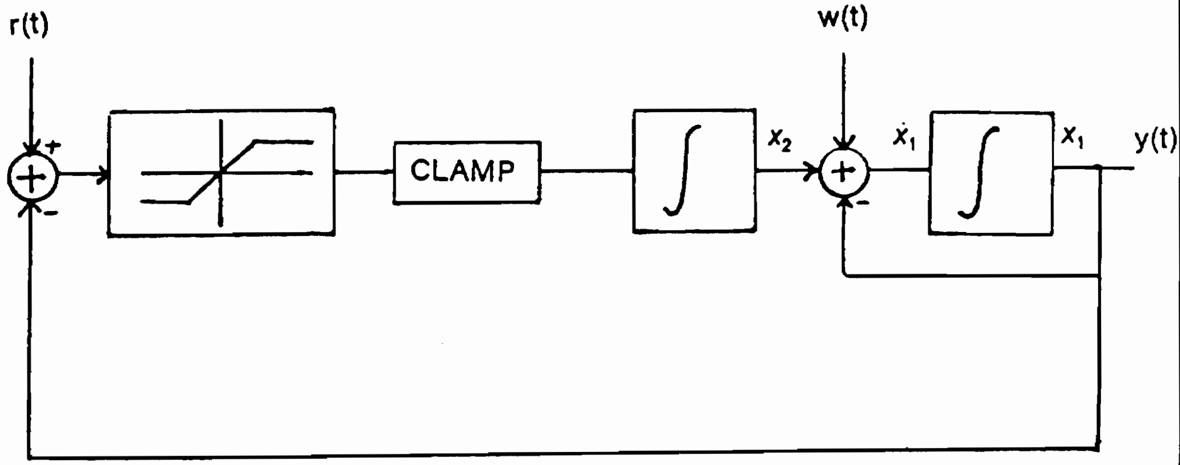

Figure 5. Modified Relay Control System.

sidered earlier. So this result doesn't come up as a surprise but just validates the estimator and endorses the earlier given reasoning for its behaviour in the case of online gain approach.

Comparison between the two systems was carried out on the basis of mean square error at the output for different values of the slope of the saturating amplifier. For identical conditions of input noise variance (0.01), measurement noise (0.1), sampling interval (0.01) the mean square error at the output was calculated for slopes varying from 200.0 to 0.1. For a given slope 's' the linear region was determined as [-1/s, 1/s]. Table 9. tabulates the values of the mean square error for the cases discussed above. In this case only the Jacobian approach was used for comparison.

Observing the values for both the systems the output mean square error is of the same order for all values of σ_w^2 but in comparison to each other there is a difference of two orders of magnitude. The modified relay control system has noisier output and results in higher output mean square error. The reason being the noise insertion position. In the modified case the noise is applied after the first low pass filter and thus it is filtered only through a single order low pass filter than a second order filter which is the case in the original relay system. Also in the original case the noise is just before the saturating amplifier and thus majority of it is clipped out due to the amplifier. This also reduces the noise content of the system state. Figs. 19, 20, 21, and 22 validate this idea of the position of noise insertion as the cause of differing output error response. The system response in terms of rise time and time taken to reach steady state is identical for both cases for fixed values of slope and other system parameters. This follows logically because without the system noise the systems are identical and therefore the responses should be identical.

An interesting observation was made regarding the error $x_{k,1} - \hat{x}_{k,1}$. Using the histogram approach, the 1000 values of the above mentioned error were used to plot out a probablity distribution of the error. The probability distribution, very surprisingly, was found to be almost normal. To check it out further, using the calculated mean and the variance of the given 1000 values a normal curve with the same mean and variance was superimposed. This is shown in Fig.23. The normal curve was very close to the histogram. To check it up further gaussian random numbers of the same statistics were added to to the original values of the error. The histogram approach was again used to plot out the probability distribution and a normal curve of the same statistics was superimposed. The resulting plot is shown in Fig.24. and validates the result that the output error is nearly gaussian and thus only the mean and the variance is required to fully define its statistics. The feedback in the system, in conjuction with the Central Limit theorem, explains the gaussian nature of the output error.

In this chapter an attempt was made to analyze the estimator behaviour by investigating three different non-linear problems and come up with some intuitive reasoning for the results obtained. In Chapter IV the estimator using the Jump Matrix Technique will be used for application purpose in Feedback control problems.

Table 1. Non Linear Spring -- m.s.e vs σ_v^2, h (Online gain approach).

	h		
σ_v^2	-0.5	0.0	1.0
0.1	0.0690064	0.0106900	0.0092802
1.0	0.0727319	0.0109177	0.0104527
4.0	0.0730655	0.0109309	0.0106106
10.0	0.0731314	0.0109314	0.0106549
100.0	0.0731694	0.0109389	0.0106970

Table 2. Non Linear Spring -- m.s.e. vs σ_v^2, h. (Jacobian approach).

	h		
σ_v^2	-0.5	0.0	1.0
0.1	0.0647215	0.0103383	0.1010077
1.0	0.1075495	0.0332378	0.3021388
4.0	0.1648555	0.0619046	0.5951792
10.0	0.1982963	0.0796488	1.1655544
100.0	0.6054380	0.1735884	4.5283955

Table 3. Nonlinear Spring -- m.s.e. vs σ_w^2, h. (Online gain approach).

	h		
σ_w^2	-0.5	0.0	1.0
0.1	0.0063701	0.0010937	0.0010566
1.0	0.0730655	0.0109309	0.0106106
10.0	3.3030772	0.1091218	0.1017887

Table 4. Non Linear Spring -- m.s.e. vs σ_w^2, h. (Jacobian approach).

	h		
σ_w^2	-0.5	0.0	1.0
0.1	0.0904251	0.0638663	0.5385664
1.0	0.1648555	0.0619046	0.5951792
10.0	0.5619093	0.1197612	0.7621210

Table 5. Van Der Poll's Eqn. -- m.s.e vs σ_v^2 ,h. (Online gain approach).

	ε		
σ_v^2	0.5	2.0	5.0
1.0	0.0047758	0.0030296	0.0090176
10.0	0.0048009	0.0104871	0.1021448
25.0	0.0048131	0.0204461	0.2755777
100.0	0.0048159	0.0572452	1.0297842

Table 6. Van Der Poll's Eqn. -- m.s.e vs σ_v^2 ,h. (Jacobian approach).

	ε		
σ_v^2	0.5	2.0	5.0
1.0	0.0247902	0.1028206	0.1925998
10.0	0.0878251	0.4383502	0.8300945
25.0	0.1611247	1.0085481	1.7858306
100.0	0.4390831	2.9658808	4.0496443

Table 7. Van Der Poll's Eqn. -- m.s.e. vs σ_w^2, h. (Online gain approach).

	ε		
σ_w^2	0.5	2.0	5.0
1.0	0.0004780	0.0146168	0.2772748
10.0	0.0048131	0.0204461	0.2755777
25.0	0.0121194	0.0249862	0.2776340

Table 8. Van Der Poll's Eqn. -- m.s.e. vs σ_w^2, h. (Jacobian approach).

	ε		
σ_w^2	-0.5	0.0	1.0
1.0	0.1514495	1.0061652	1.8455638
10.0	0.1611247	1.0085481	1.7858306
25.0	0.1729297	1.0060306	1.7301761

Table 9. Relay control System Original vs Modified Configuration.

output MSE		
slope	Original	Modified
200.0	0.0022999	0.1206731
100.0	0.0021383	0.1206731
20.0	0.0017647	0.1206861
10.0	0.0021493	0.1207784
5.0	0.0020572	0.1208955
3.0	0.0020138	0.1209236
2.0	0.0020205	0.1209290
1.0	0.0018697	0.1207281
0.5	0.0017565	0.1203282
0.1	0.0016322	0.1106010

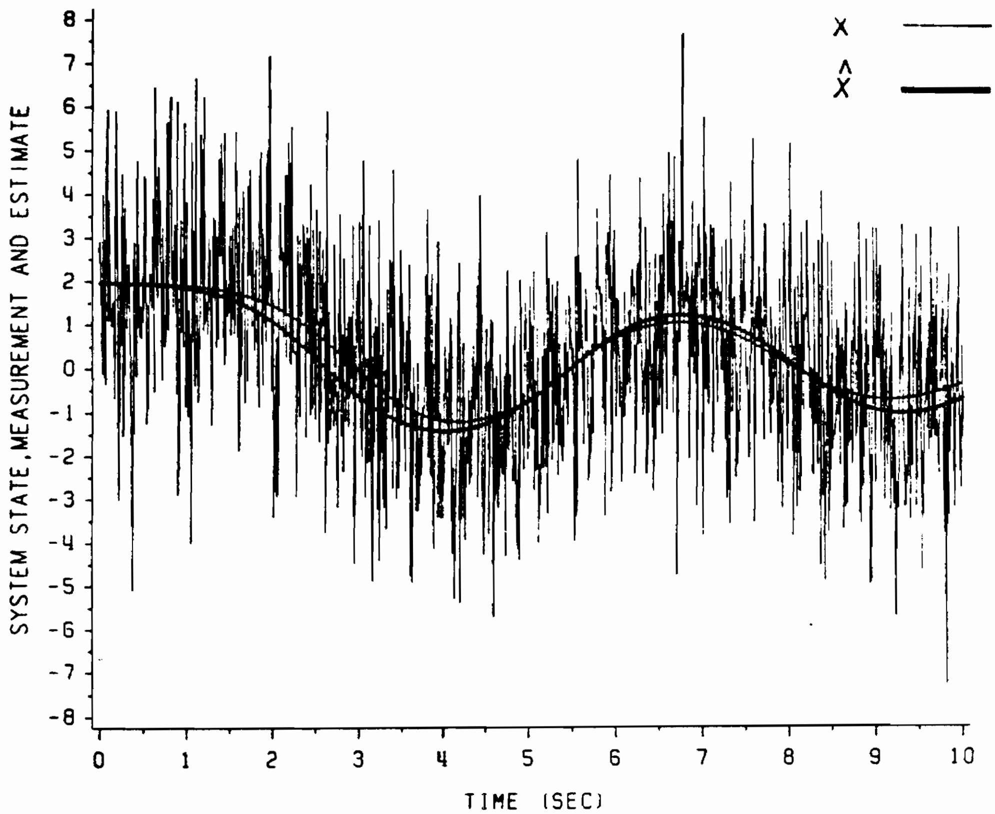

Figure 6. Nonlinear Spring Response using Online Gain. (h = -0.5)

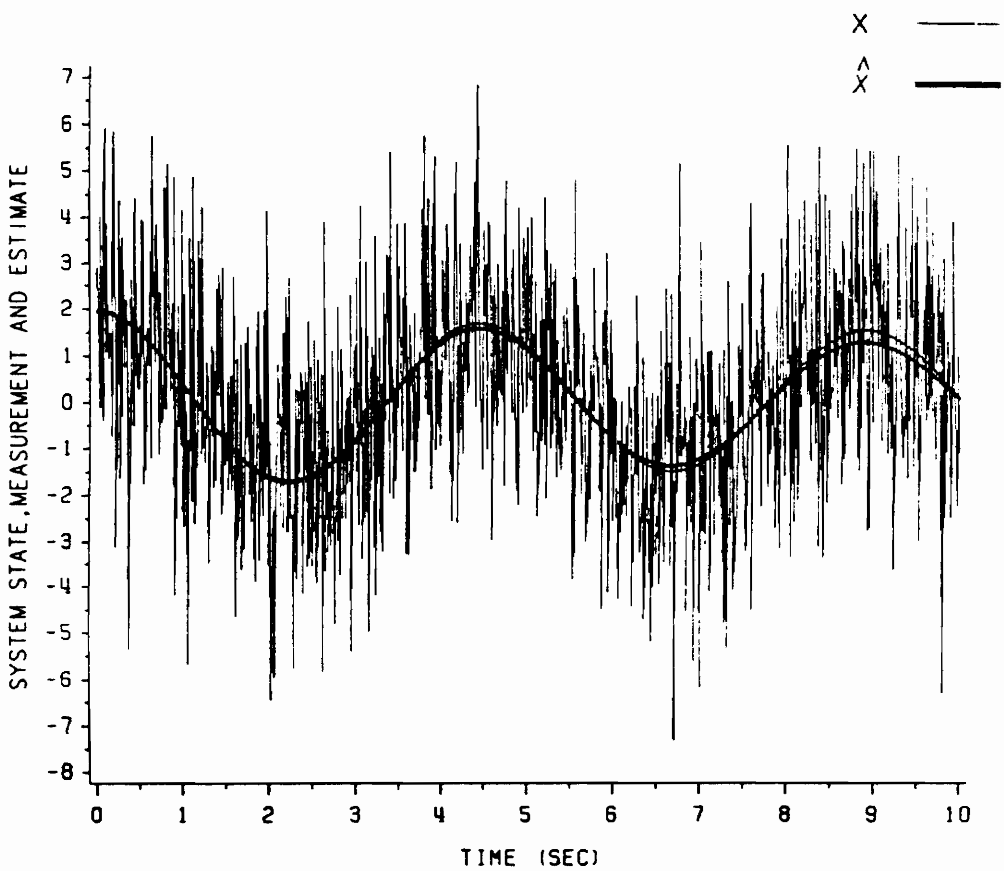

Figure 7. Nonlinear Spring Response using Online Gain. (h = 0.0)

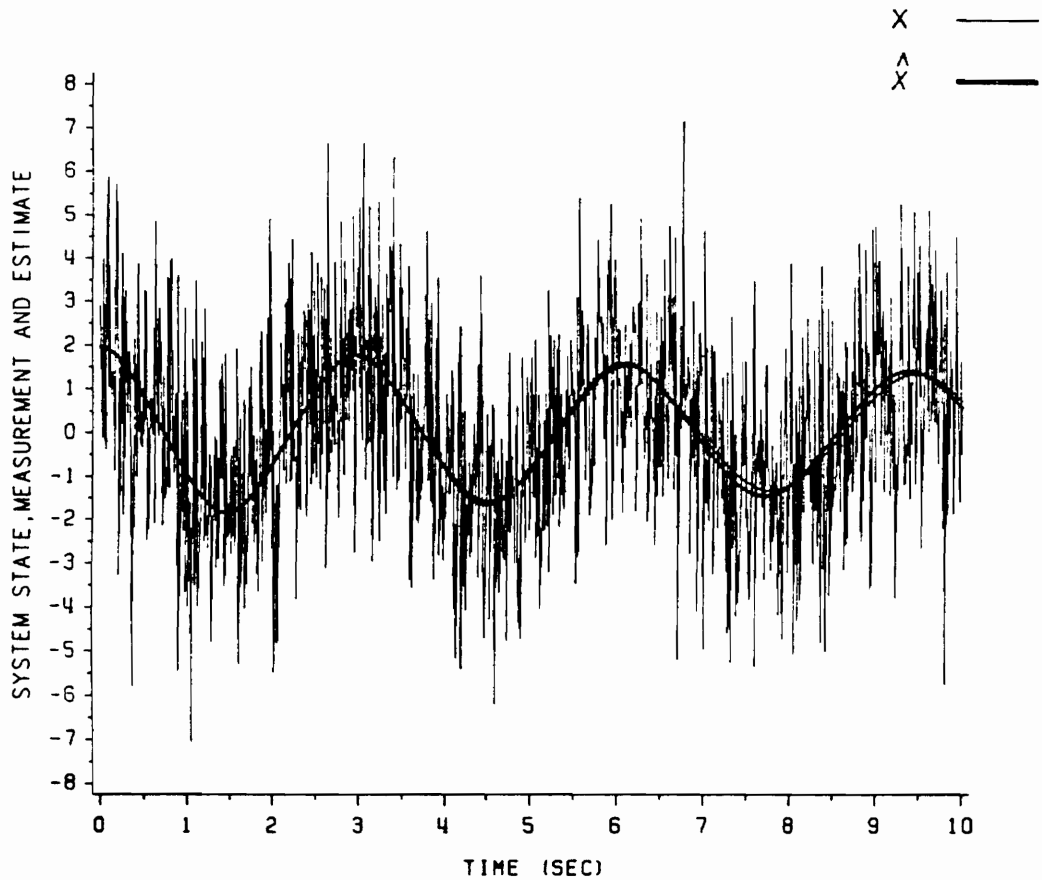

Figure 8. Nonlinear Spring Response using Online Gain. (h = 1.0)

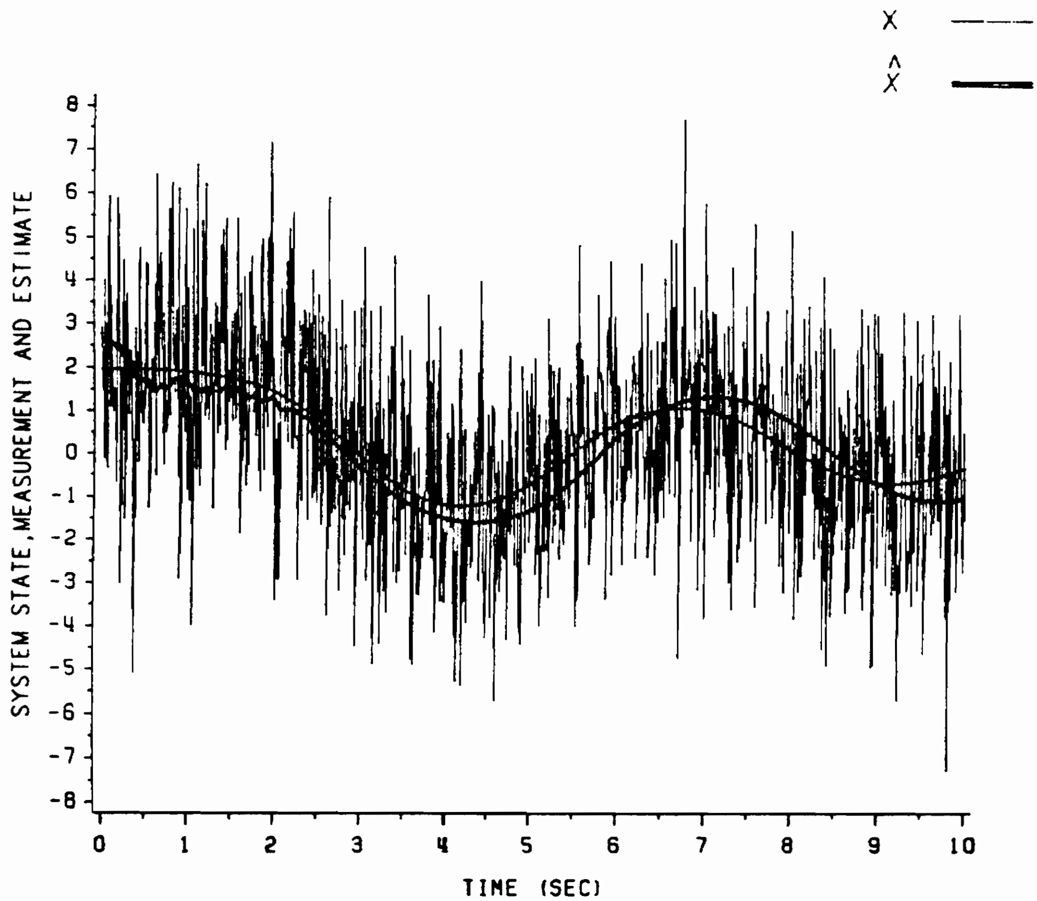

Figure 9. Nonlinear Spring Response using Jacobian Gain. (h = -0.5)

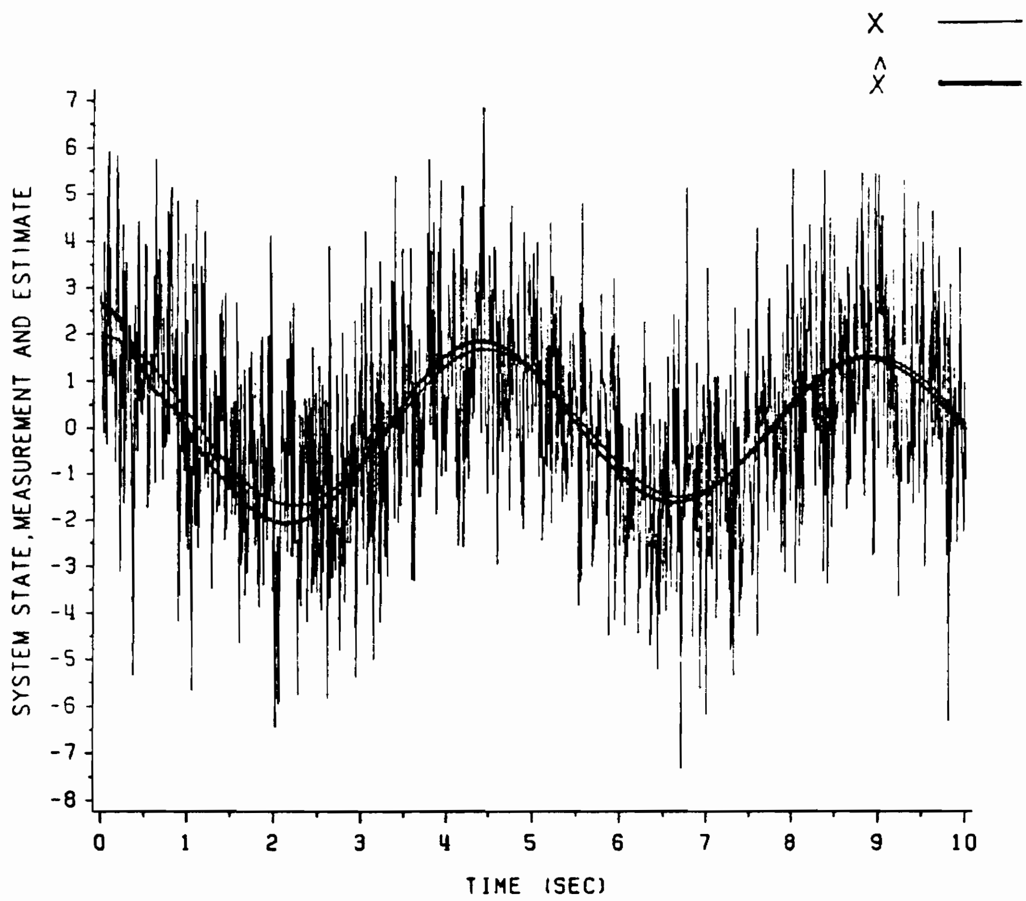

Figure 10. Nonlinear Spring Response using Jacobian Gain. (h = 0.0)

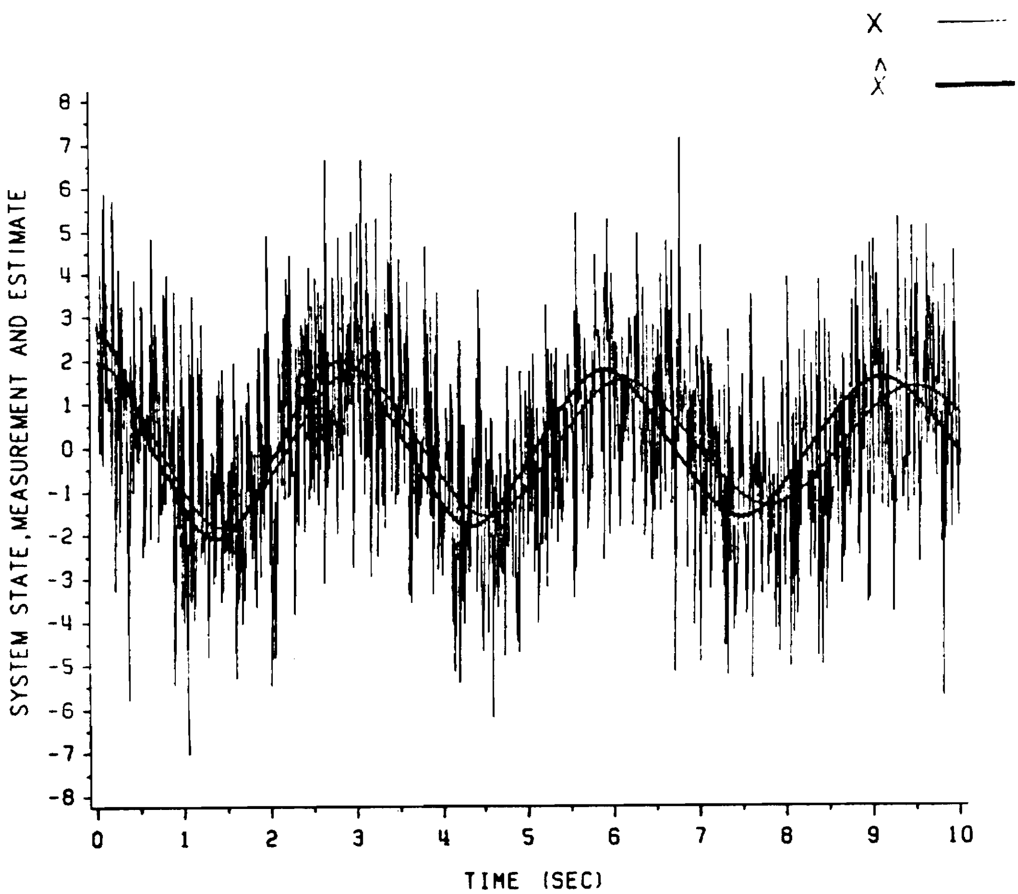

Figure 11. Nonlinear Spring Response using Jacobian Gain. (h = 1.0)

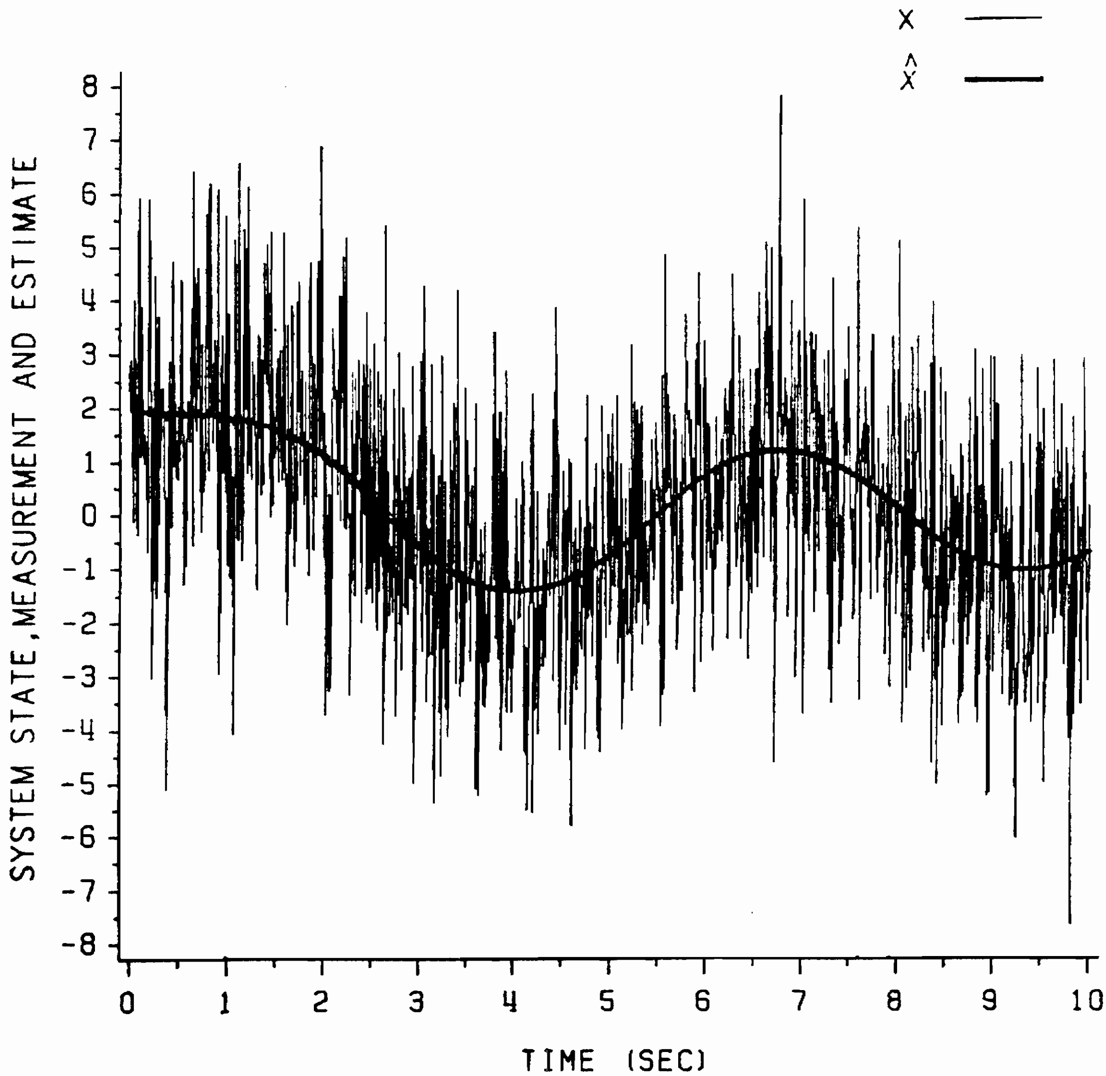

Figure 12. Nonlinear Spring Response using IMSL subroutine. (h = -0.5)

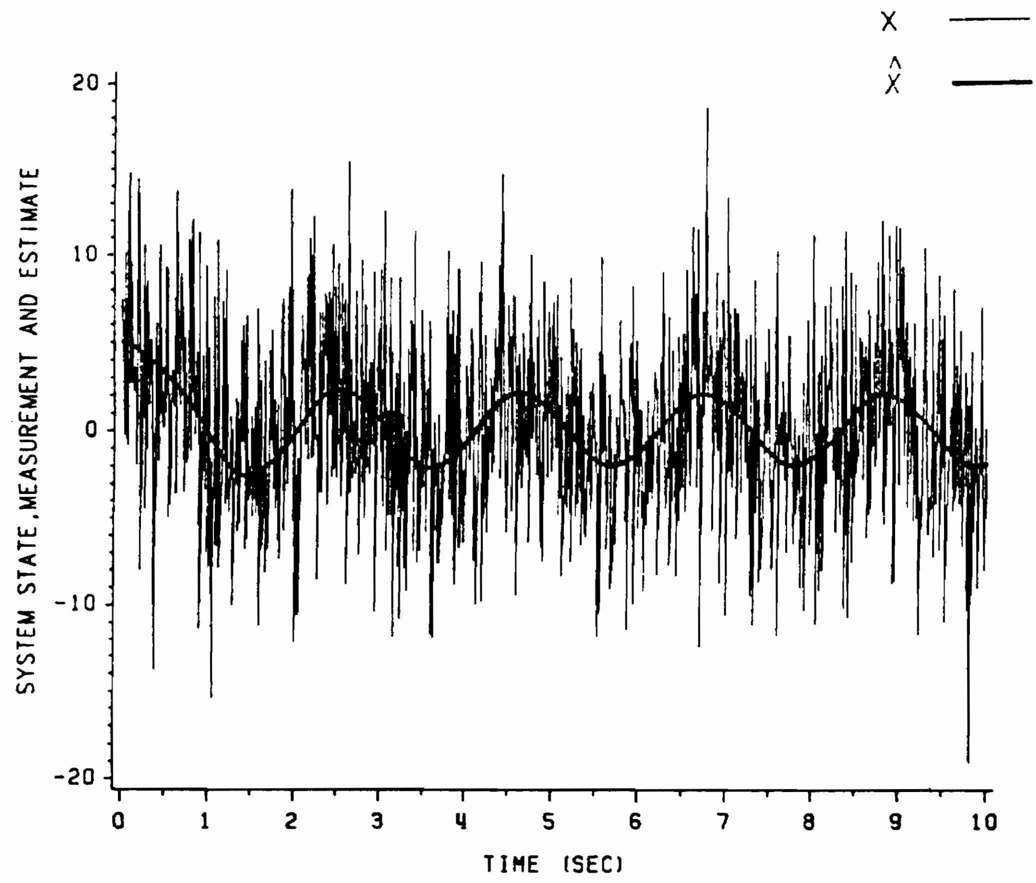

Figure 13. Solution of Van Der Poil's Equation using Online Gain. (ε = 0.5)

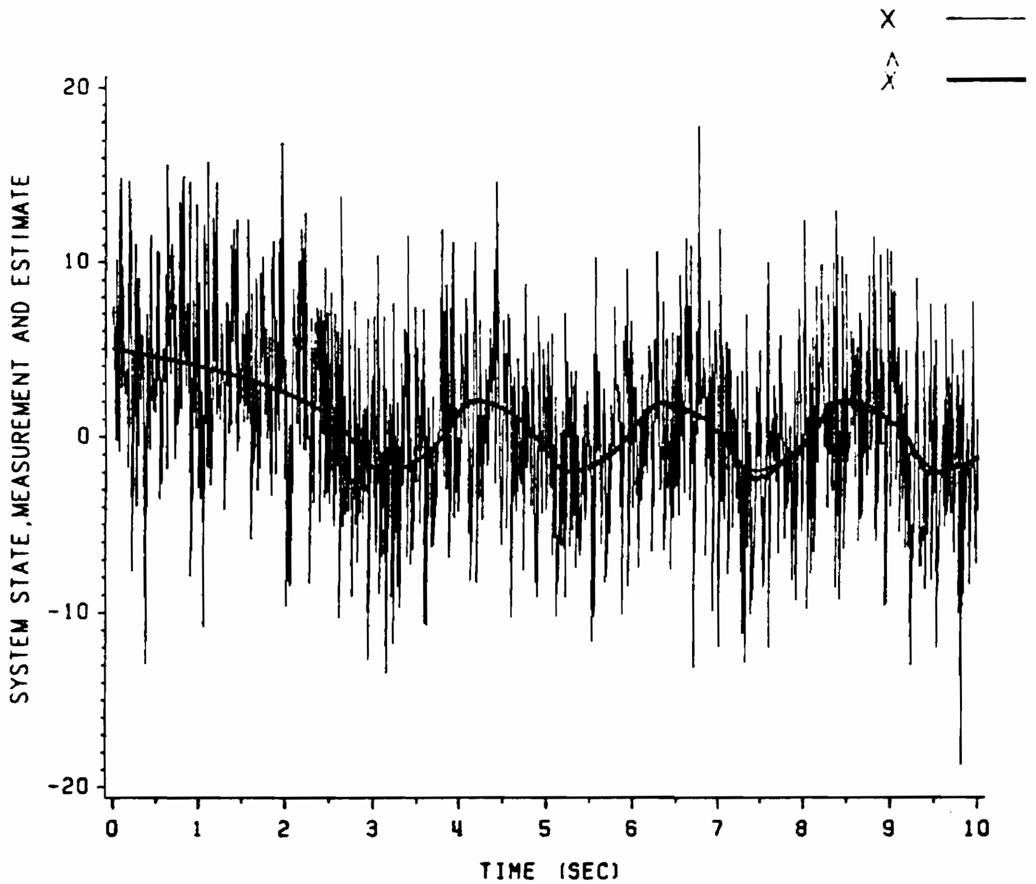

Figure 14. Solution of Van Der Poll's Equation using Online Gain. (ε = 2.0)

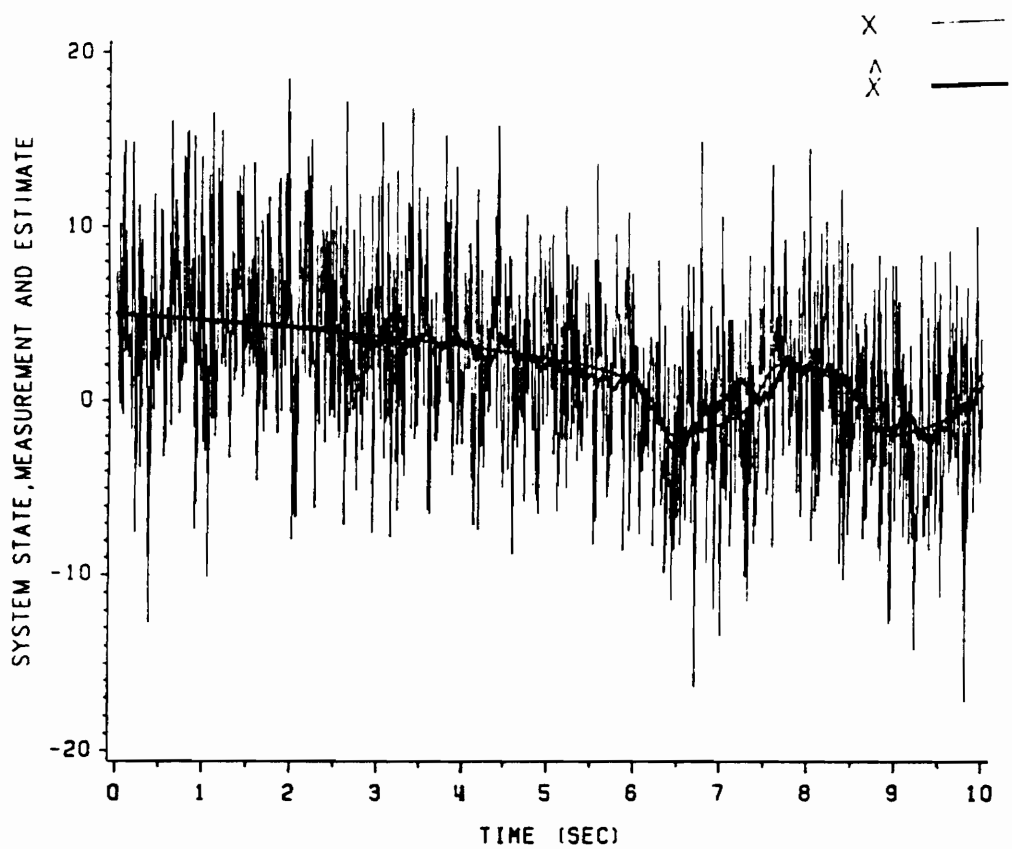

Figure 15. Solution of Van Der Poll's Equation using Online Gain. (ε = 5.0)

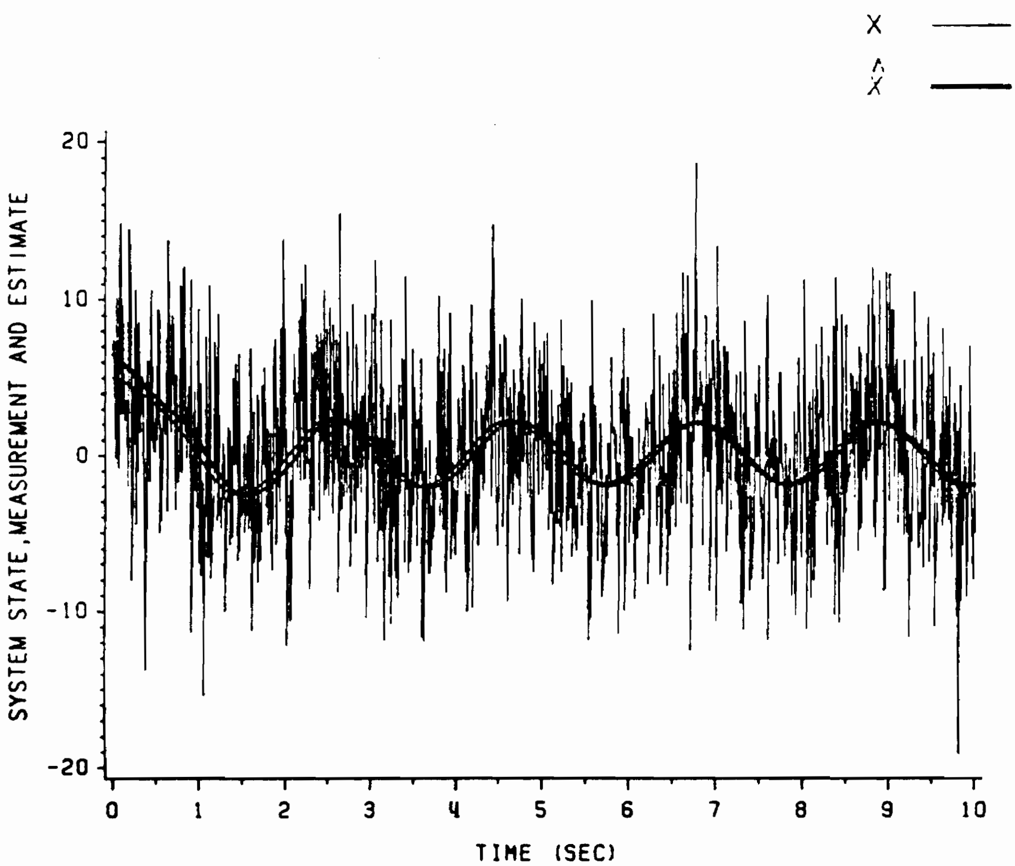

Figure 16. Solution of Van Der Pol's Equation using Jacobian Gain. (ε = 0.5)

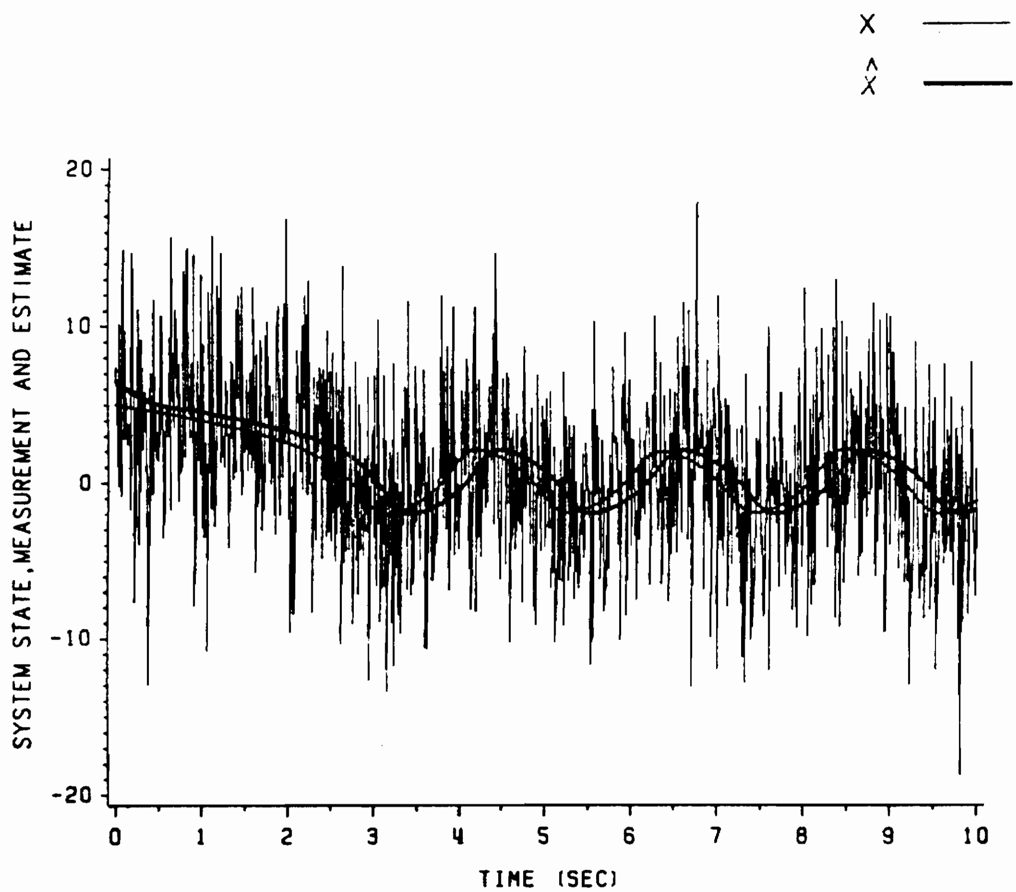

Figure 17. Solution of Van Der Poll's Equation using Jacobian Gain. (ε = 2.0)

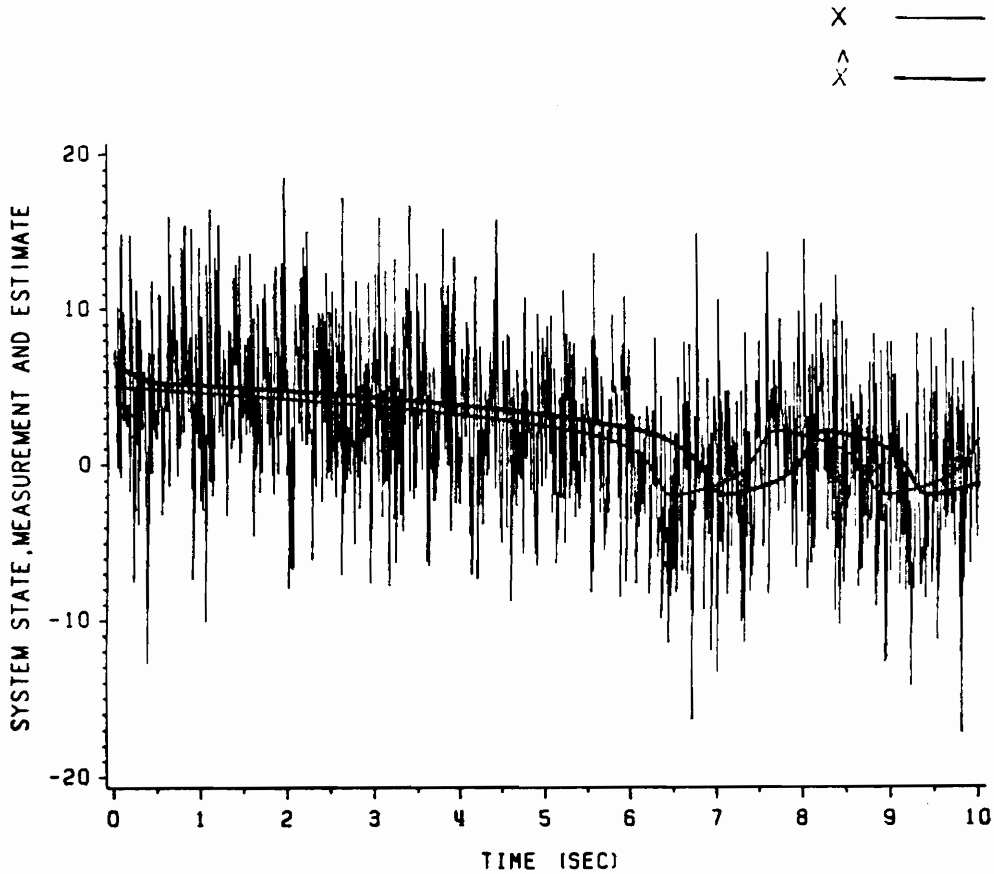

Figure 18. Solution of Van Der Poll's Equation using Jacobian Gain. (ε = 5.0)

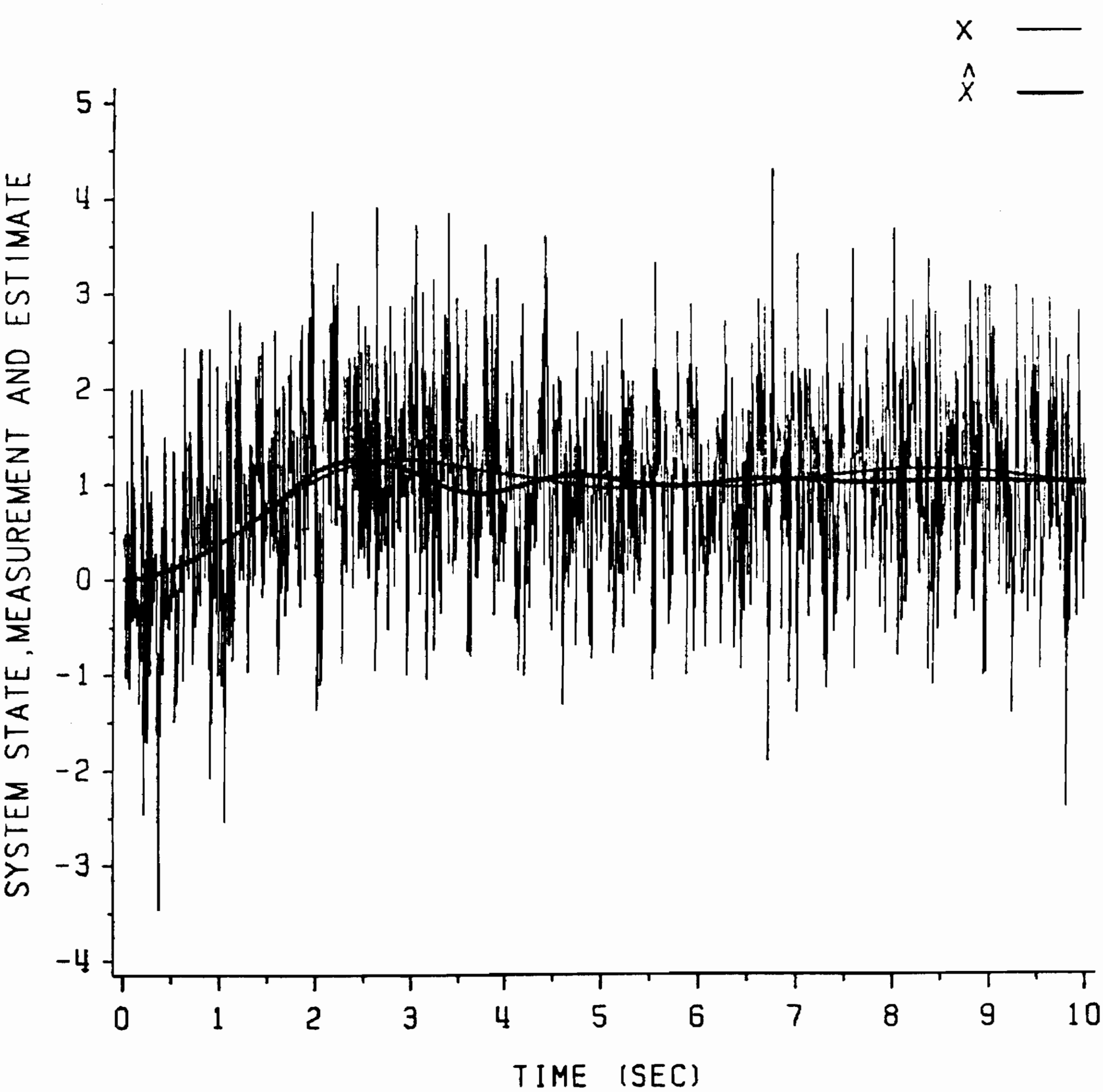

Figure 19. Step response for Relay control system using Online gain approach.

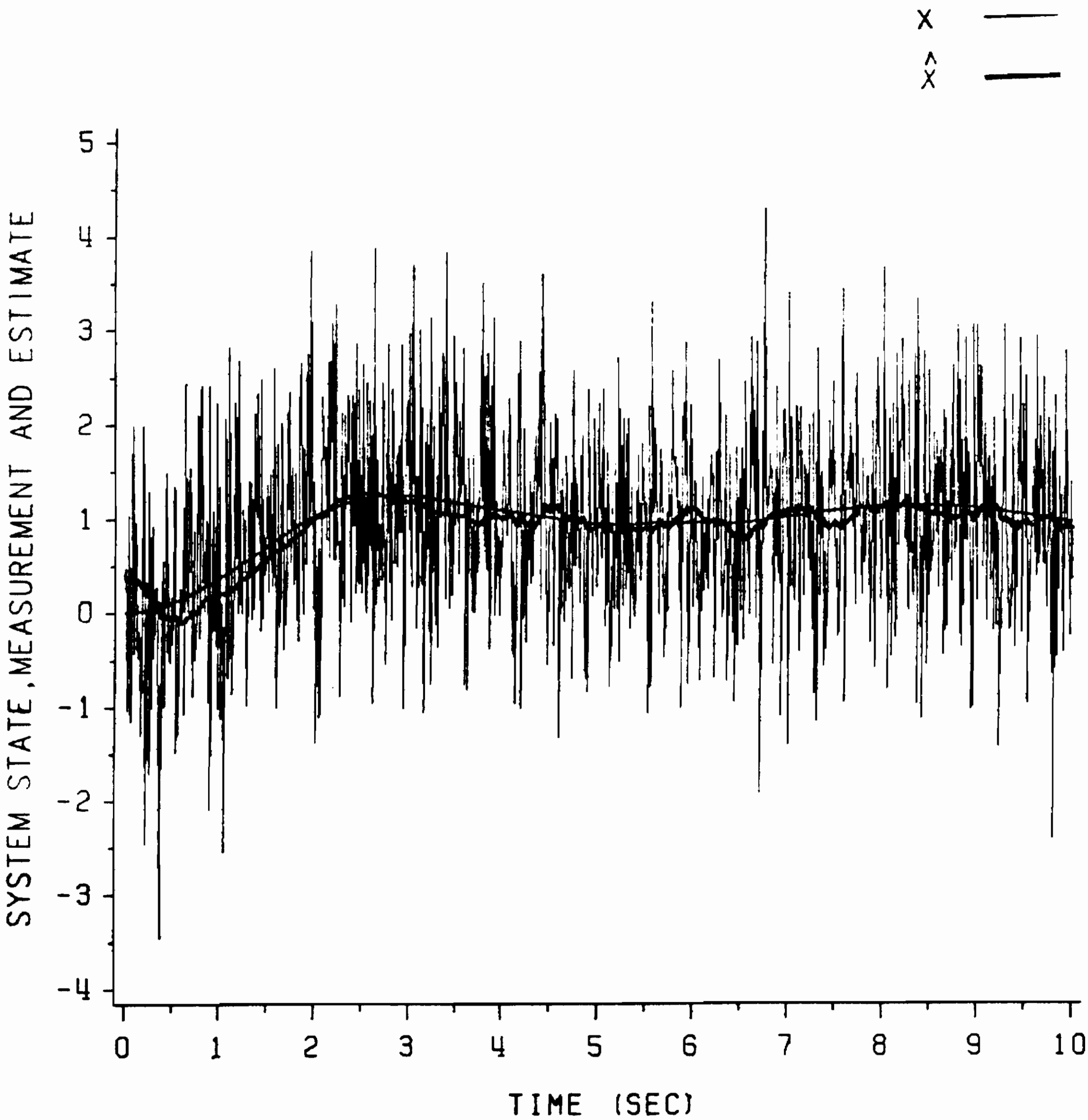

Figure 20. Step response for Relay control system using Jacobian approach.

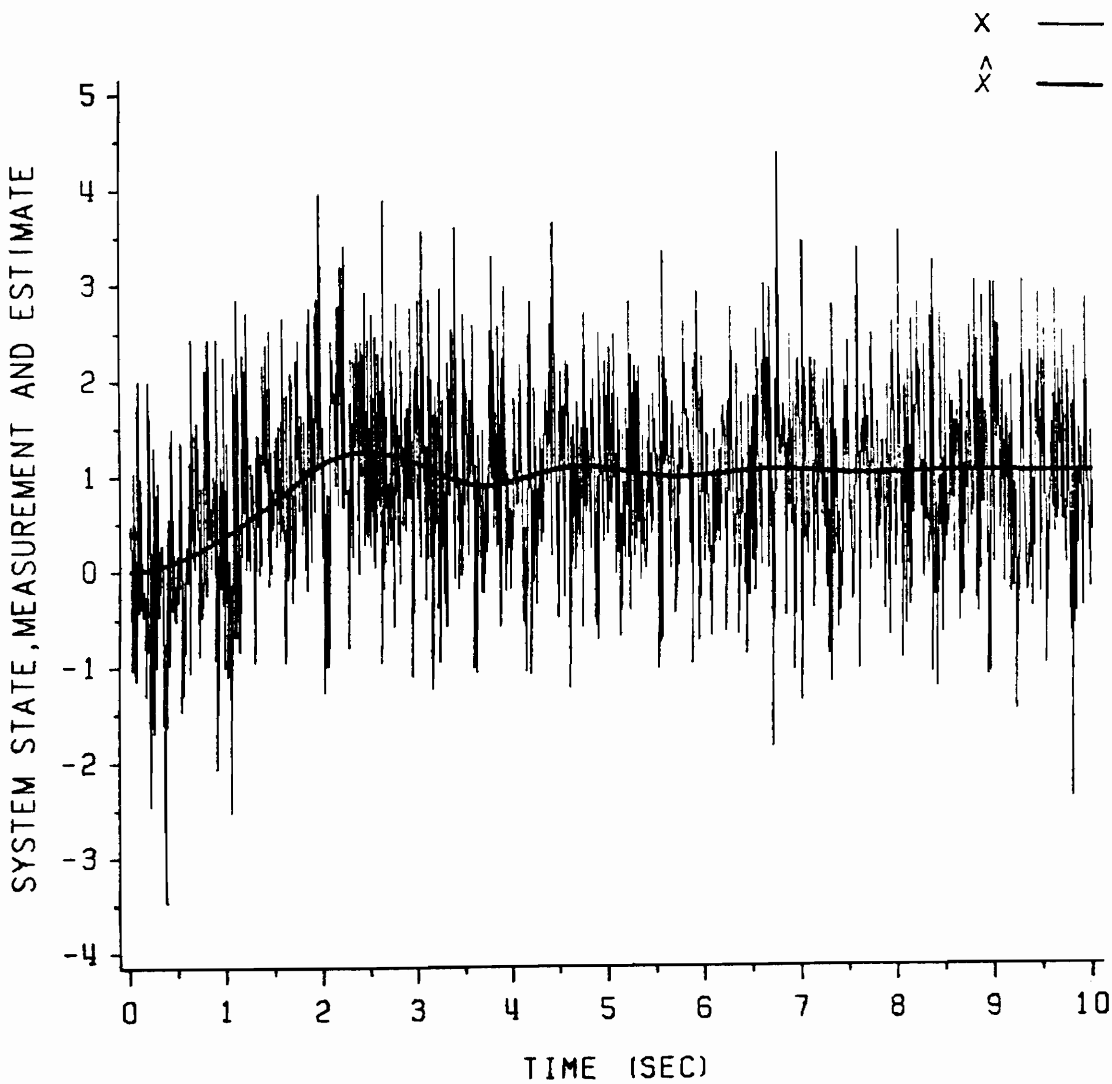

Figure 21. Step response for Modified Relay control system using Online gain approach.

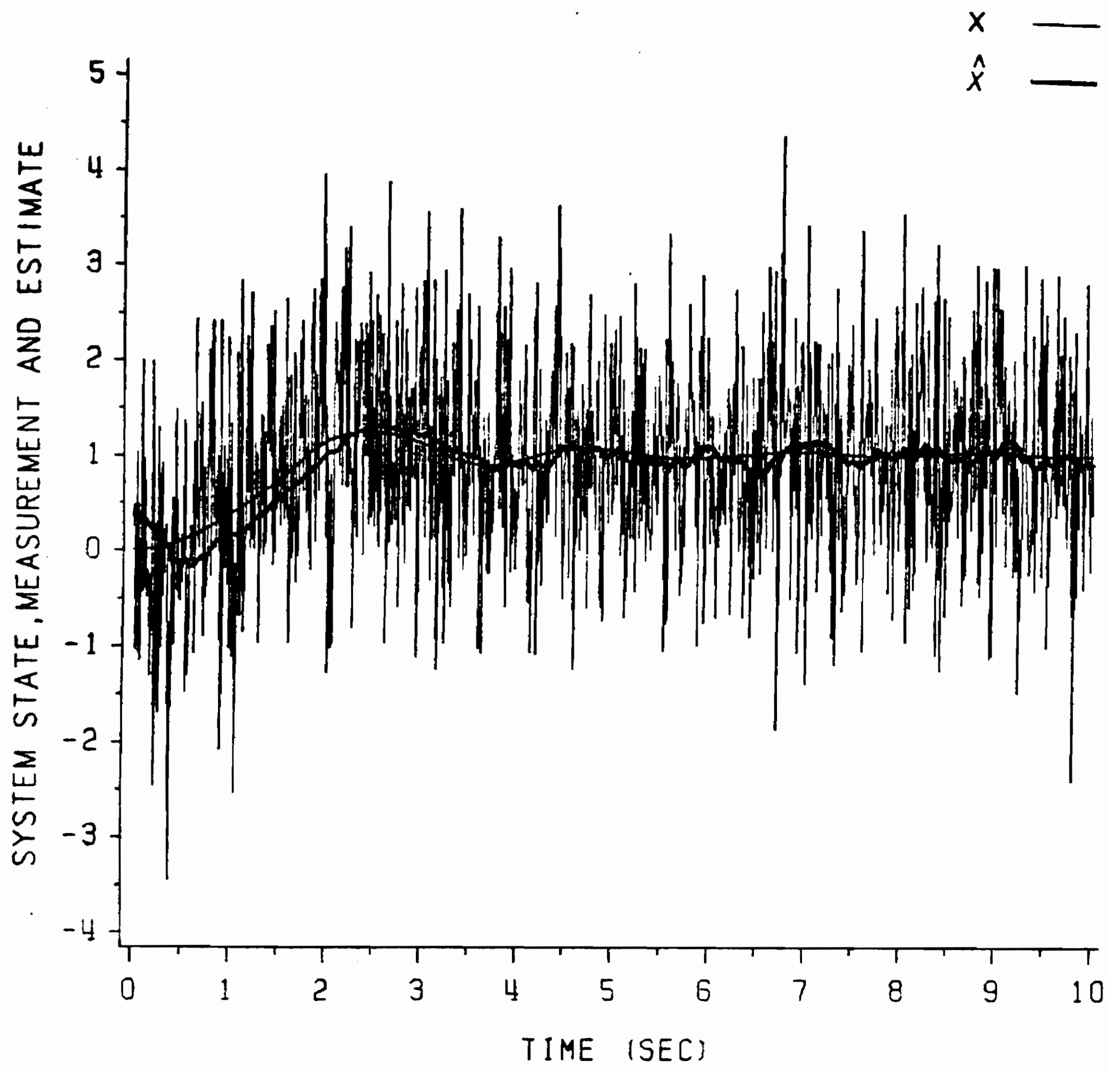

Figure 22. Step response for Modified Relay control system using Jacobian approach.

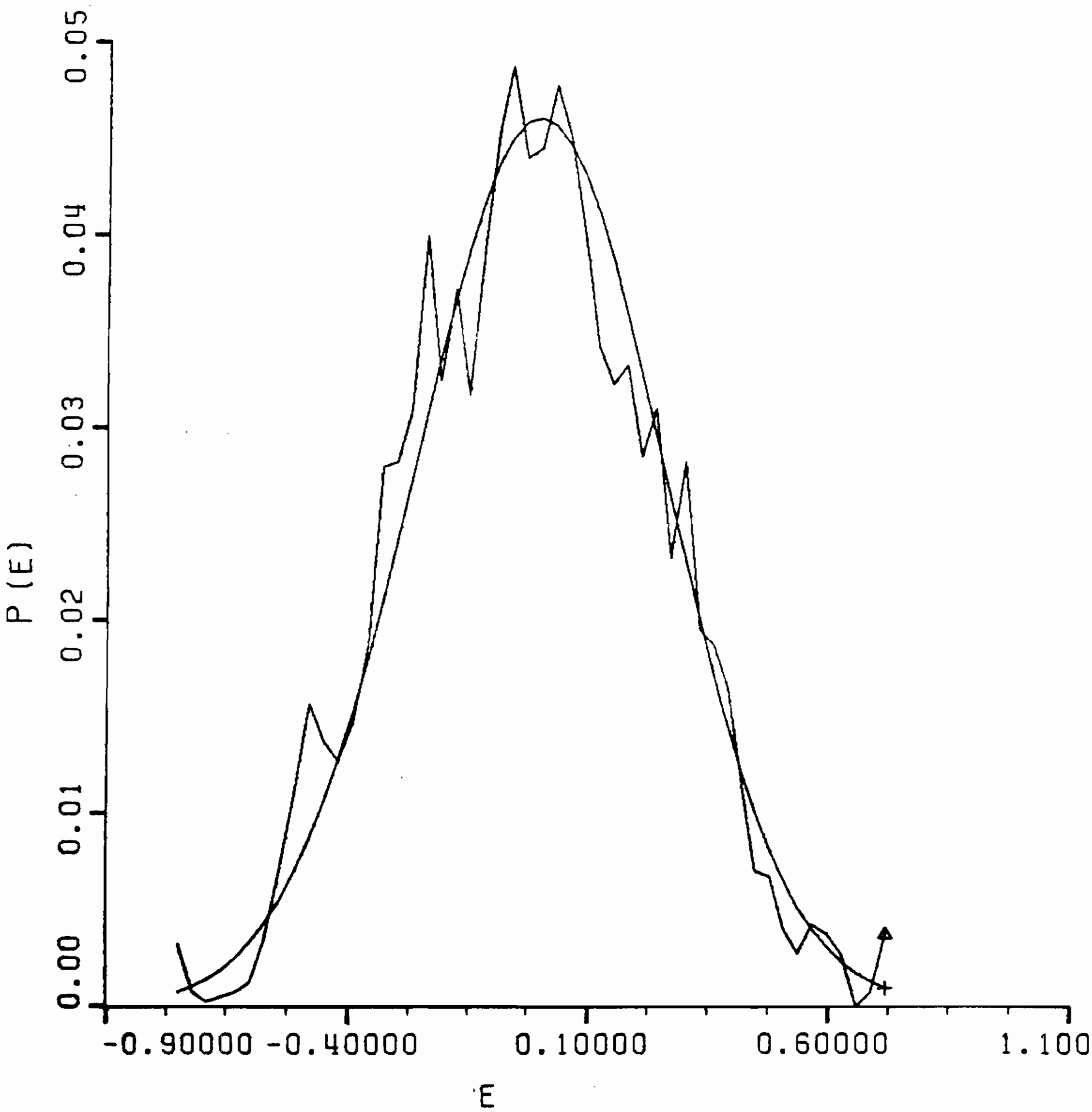

Figure 23. Probability distribution of the error in the estimated state.

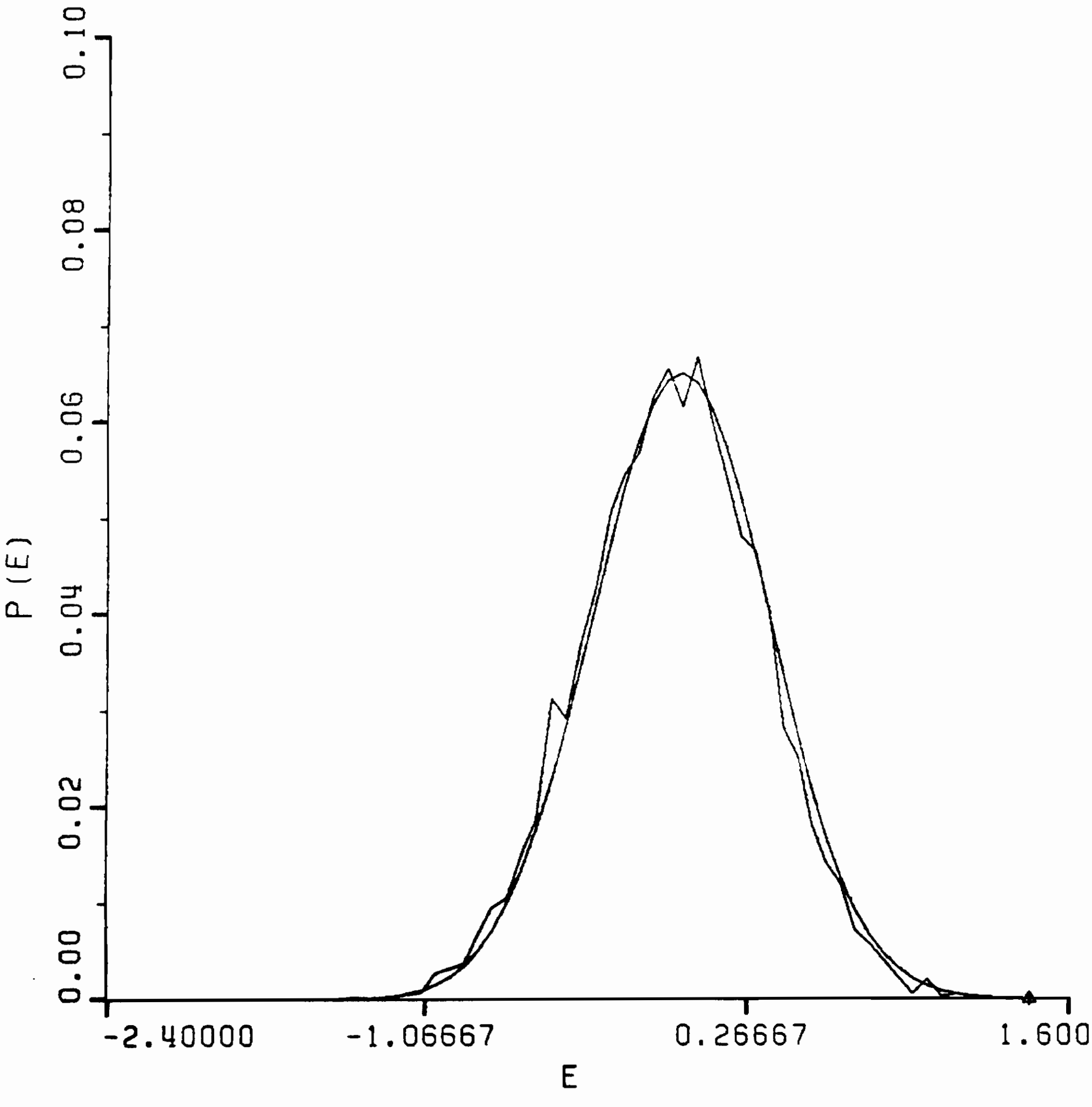

Figure 24. Probability distribution of the (error + gaussian noise).

CHAPTER IV

ESTIMATOR APPLICATION IN FEEDBACK CONTROL

4.1 Introduction

In control system theory, it is generally assumed that all state variables of a system can be measured directly. A desired system response can be obtained using proper feedback of the measured state variables. But what if not all system state variables are directly measurable, i.e. What if the system is partially observable? For a linear system, using kalman filter, the missing state variables can be calculated theoretically and appropriate feedback strategies can be adopted for desired system response. In case of non-linear stochastic systems, the theoretical approach may not always work. In this chapter, the estimator developed using Jump Matrix Technique, will be used to estimate the state of the system and using appropriate feedback control, the system response will be modified. The first case is that of the relay control system discussed in the previous chapter.

4.2 Feedback Control in Relay Control System.

The relay control system discussed in Chapter III had a second order low pass filter in the forward path. If the saturating amplifier was replaced by a linear amplifier then the whole system would have been a unity gain feedback control system, with a second order low pass filter in the forward path and the system transfer function would be given as :

$$H(s) = \frac{\frac{K}{s(s+1)}}{1+\frac{K}{s(s+1)}} = \frac{K}{s^2+s+K} \qquad [4.1]$$

where 'K' represents the gain of the linear amplifier.

The idea was to remove the unity feedback and then use a particular linear combination of the system state variables as feedback, to obtain a system transfer function given below :

$$H(s) = \frac{K}{s^2+4s+8} \qquad [4.2]$$

The system setup is shown in Fig. 25. The feedback gains a_1 and a_2 were obtained as 0.5 and 3.0 respectively for a amplifier gain of 10.0 (i.e. K = 10.0). The slope was chosen to be 10.0 so that at a later stage the linear amplifier could be replaced by the saturating amplifer considered earlier in relay control system in Chapter III.

The next step in the feedback control problem was to use linear combination of the estimated values of the state variables instead of actual state variables as the feed-

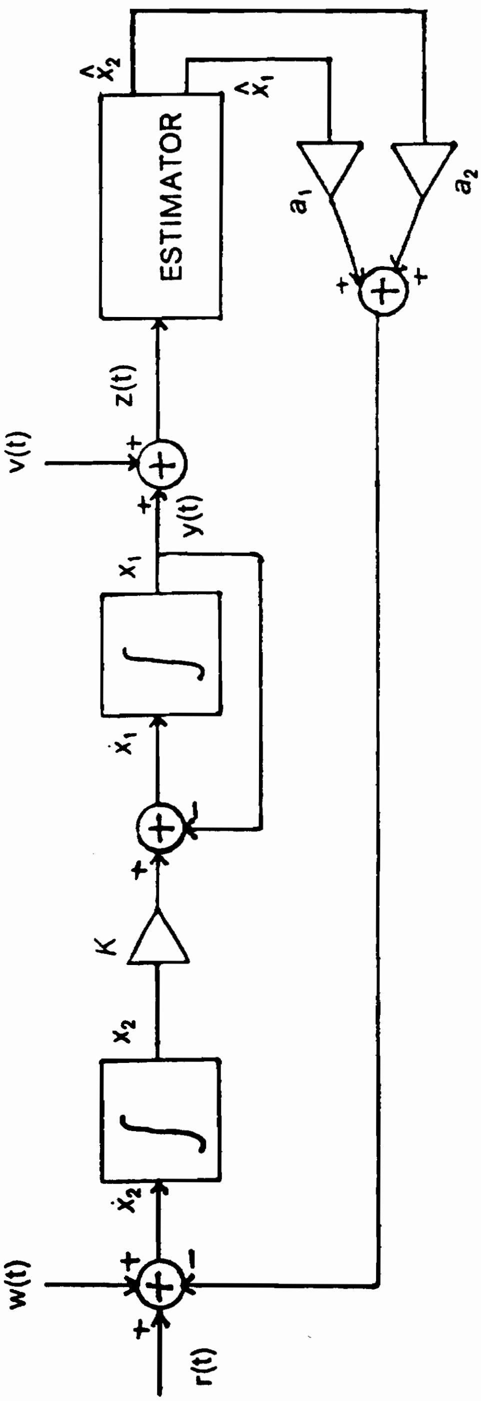

Figure 25. Linear relay control system.

back. That is to use ($a_1\hat{x}_1 + a_2\hat{x}_2$) as the negative feedback instead of ($a_1x_1 + a_2x_2$) and then to observe the system behaviour.

The feedback signal ($a_1\hat{x}_1 + a_2\hat{x}_2$) is considered as a deterministic input to the system and the control system can now be modelled as

$$x_1 = x \qquad \dot{x}_1 = Kx_2 - x_1$$

$$x_2 = \dot{x}_1 \qquad \dot{x}_2 = -a_1\hat{x}_1 - a_2\hat{x}_2 + r(t) + w(t)$$

Let $r(t) - a_1\hat{x}_1 - a_2\hat{x}_2 = u(t)$ The state equations can now be written in the matrix form as

$$\dot{X} = \begin{bmatrix} -1 & K \\ 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w(t) \qquad [4.3]$$

The system defined above was simulated and then, using a kalman filter, the linear combination of the system state variables was fed back as a deterministic input. The discrete time version of the state equations can be written in the closed form as :

$$X_{k+1} = \phi_{OL}X_k + \psi w_k + \Gamma u_k \qquad [4.4]$$

$$= \phi_{OL} + \psi w_k + \Gamma(r - a\hat{x}_k)$$

$$= (\phi_{OL} - \Gamma a)x_k + \psi w_k + \Gamma r$$

where 'a' is the feedback vector. The last step follows from the fact that as $k \to \infty$, $\hat{x}_k \to x_k$.

Defining $\phi_{OL} - \Gamma a = \phi_{CL}$ the above equation can now be written as

$$X_{k+1} = \phi_{CL}X_k + \psi w_k + \Gamma r_k$$

For the calculation of Kalman gains at each iterative step, the first step is the calculation of M_{k+1} , using the equation $M_{k+1} = \phi P_k \phi^T + \psi Q \psi^T$. The problem now is that which ϕ is to be used for the calculation of M_{k+1}? The solution to the problem can be found on the basis of desired system response. If the system which is modelled is an open loop system, then ϕ_{OL} should be used, but if the system was designed for a closed loop response, as in this case, the ϕ_{CL} should be used. All along this chapter both alternatives of using ϕ will be used. The system response, when ϕ_{OL} is used, is more noisy. This can be explained by the fact that the model in this case represents an open loop system which is inherently more noisy. Figs.28, 29, depict the system response, measurement process and the estimated state using ϕ_{CL} and ϕ_{OL} respectively. The system noise variance σ_w^2 and measurement noise σ_v^2 were set for 0.1 and 0.01 respectively with the sampling time period T equal to 0.01 secs.

Using the values of a_1 and a_2 calculated earlier for the linear case, the nonlinear saturating amplifier is introduced in the forward path instead of the linear amplifier of gain K. The new system is shown in Fig.25. The system can be modelled as follows :

Let $\quad x_1 = y \qquad \dot{x}_1 = x_2 - x_1$

$$x_2 = f(x_3) \qquad \dot{x}_2 = 0$$

$$\dot{x}_3 = r(t) + w(t) - a_1\hat{x}_1 - a_2\hat{x}_3 = u(t) + w(t)$$

Setting up the state equations in a matrix form

$$\dot{X} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t) \qquad kT^+ < t < (k+1)T^- [4.5a]$$

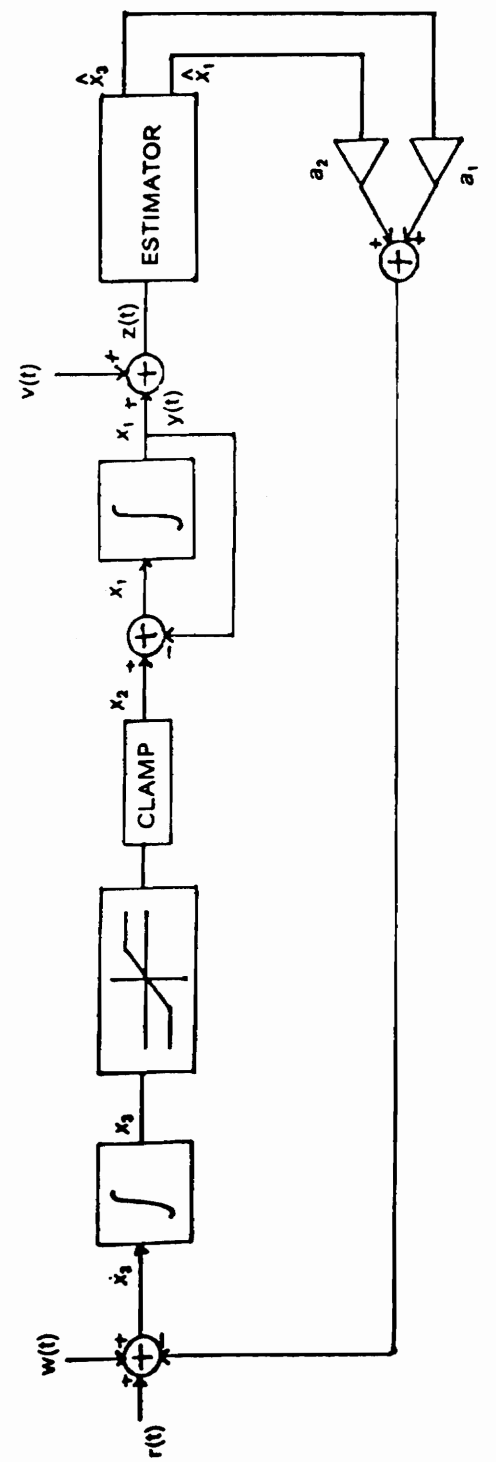

Figure 26. Feedback System for Relay Control System.

$$X(kT^{+}) = \begin{bmatrix} 1 & 0 & 0 \\ & f(x_3) & \\ 0 & 1 & 0 \end{bmatrix} X(kT^{-}) \qquad [4.5b]$$

where f(.) refers to the saturating amplifier discussed in Section 3.4.

Figures 30 and 31 depict the system state, measurement and the estimated state for the two cases of usage of ϕ_{CL} and ϕ_{OL} respectively. The system parameters were the same as in the linear case. The saturating effect of the amplifier cuts down the system noise, and the output in this case is less noisy than in the previous case of the linear amplifier. On observing the system response obtained using actual values of the state variables, there is no difference between the two responses. The only difference is that the rise time of the response using actual state variables is slightly less than in the previous case using the estimated state as feedback. The reason being that when the actual state is used for feedback the effect is more direct and leads to faster system response, while in the case of estimated state used as feedback, the measurement noise and the delay in estimator lead to a higher rise time. Overall the system performance is very satisfactory. To check the validity of the estimator usage for feedback control, a second system is considered in the next section.

4.3 Feedback Control in Nonlinear Spring

The first example discussed in Chapter III was that of a nonlinear spring. Because of the well documented behaviour of the system, the same was taken up for further investigation. Here the basic equation defining the system response was the same (equation (3.1)). The development of the feedback control problem is as follows :

$$\ddot{x} + \mu\dot{x} + \omega_o^2 x + hx^3 = w(t)$$

choosing $\quad x_1 = x \qquad \dot{x}_1 = x_2$

$$x_2 = \dot{x}_1 \qquad \dot{x}_2 = -\,\omega_o^2 x_1 - \mu x_2 - hx_1^3 + w(t)$$

$$x_3 = hx_1^3 \qquad \dot{x}_3 = 0$$

Instead of using the actual state variables the estimated values will be used for calculation of $\dot{x}_2$ i.e.

$$\dot{x}_2 = -\,\mu\hat{x}_2 - \omega_o^2\hat{x}_1 - x_3 + w(t)$$

The system configuration can now be defined as in Fig.27. The state equations written in the matrix form are

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} (\omega_o^2\hat{x}_1 + \mu\hat{x}_2) + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} w(t) \qquad [4.6a]$$

$$X(kT^+) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & hx_1^3 & \end{bmatrix} X(kT^-) \qquad [4.6b]$$

The system defined above was simulated using T = 0.01 secs, $\sigma_v^2 = 4.0, \sigma_w^2 = 1.0$ and $h = 1.0$ (hard spring). Figs.32, 33, represent the system response, measurement and estimated state for the two cases of ϕ_{CL} and ϕ_{OL} usage for calulation of Kalman gains. The results are as expected, namely that in the open loop

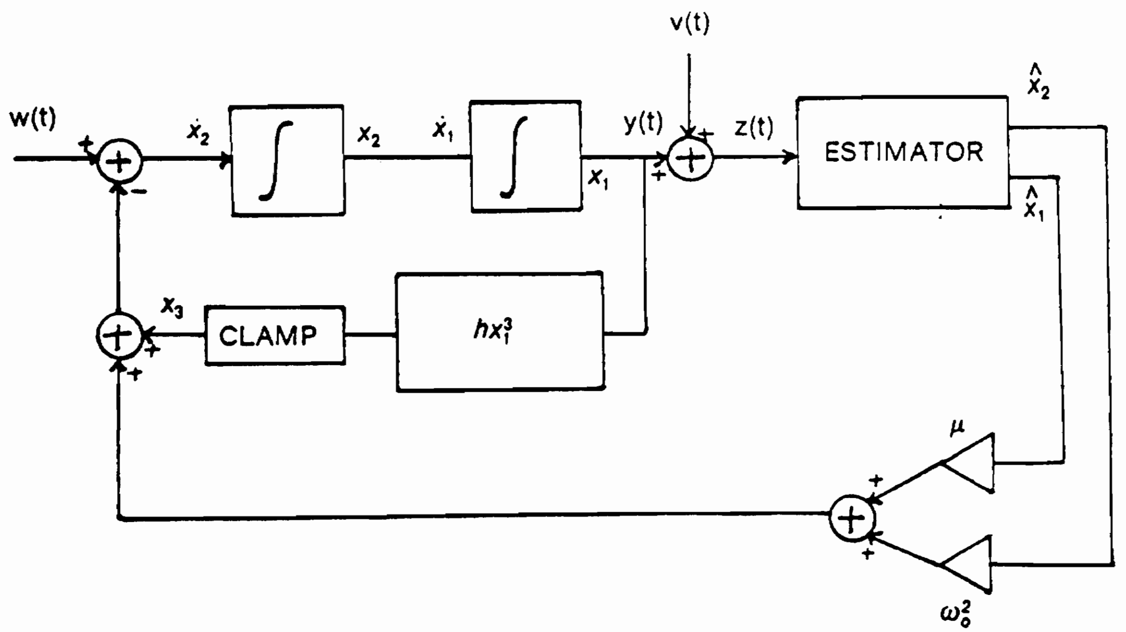

Figure 27. Feedback Control of Nonlinear Spring.

case the response is more noisy. A modification was made in the system nonlinearity. The existing nonlinearity was modified and made more complex by changing the spring constant 'h'. The transformed nonlinearity is given as follows :

$$x_3 = \begin{bmatrix} 1.0 & x_1 > 1.0 \\ x_1^3 & |x_1| < 1.0 \\ -1.0 & x_1 < -1.0 \end{bmatrix}$$

The values of μ and ω_o^2 were kept the same as in previous case. The system model was same as discussed previously, except for the remodelled nonlinearity. The system simulation was carried out for the same system parameters as in the previous case. Figs.34, 35 represent the system response, measurement and estimated system response for the two cases of ϕ_{CL} and ϕ_{OL} respectively.

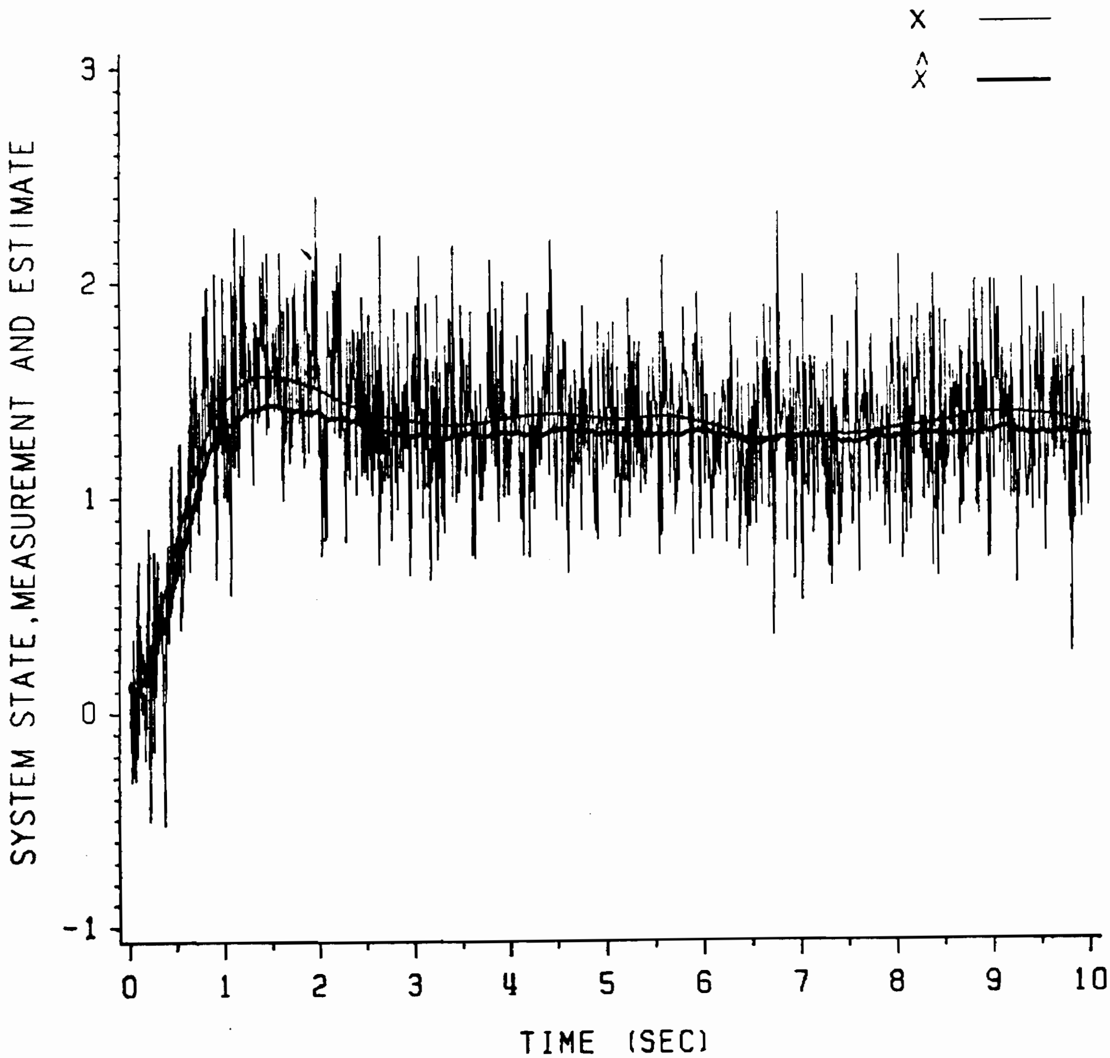

Figure 28. Linear Relay control system response using Feedback Control (ϕ_{CL} case).

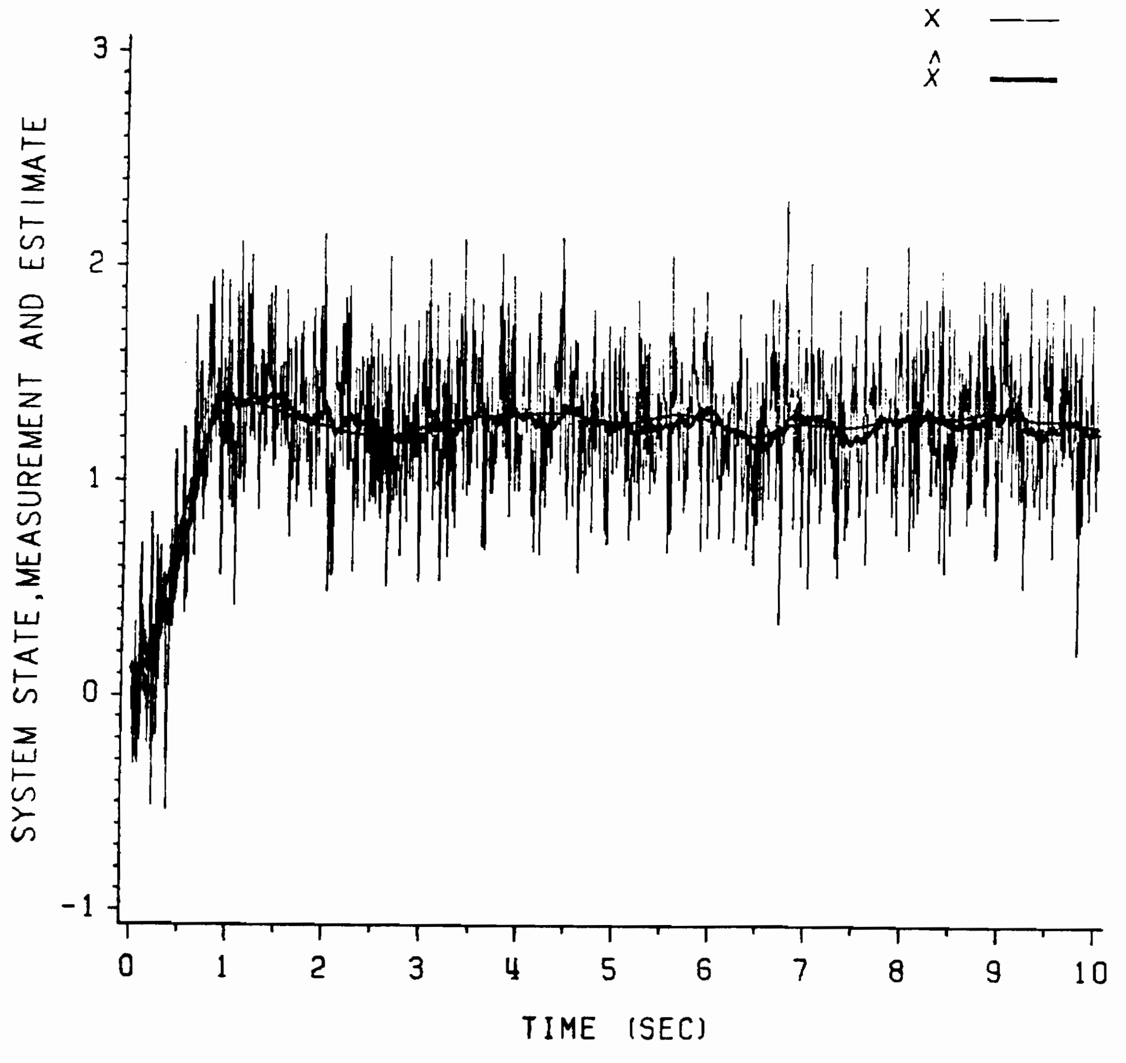

Figure 29. Linear Relay control system response using Feedback Control (ϕ_{OL} case).

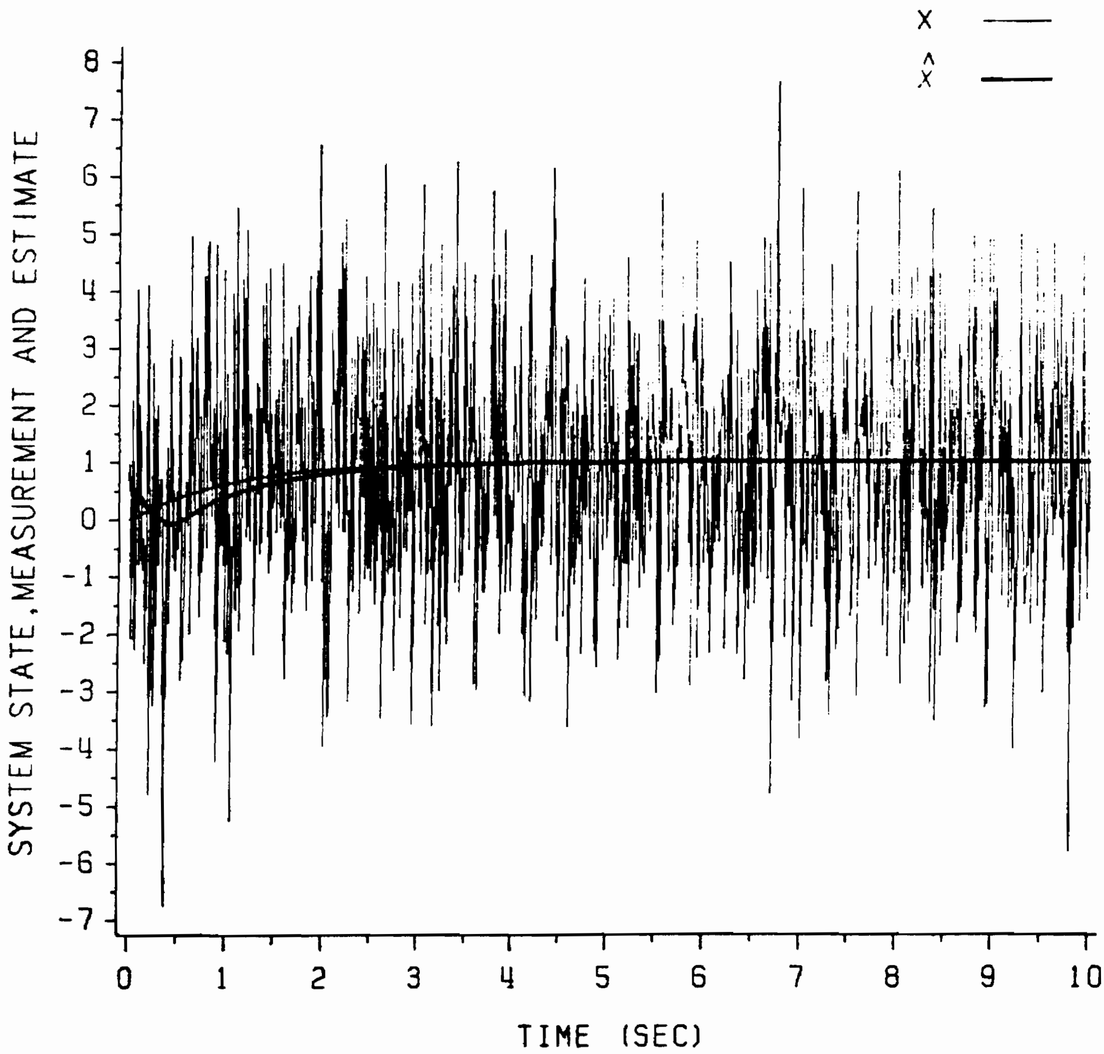

Figure 30. Relay control system response using Feedback Control (ϕ_{CL} case).

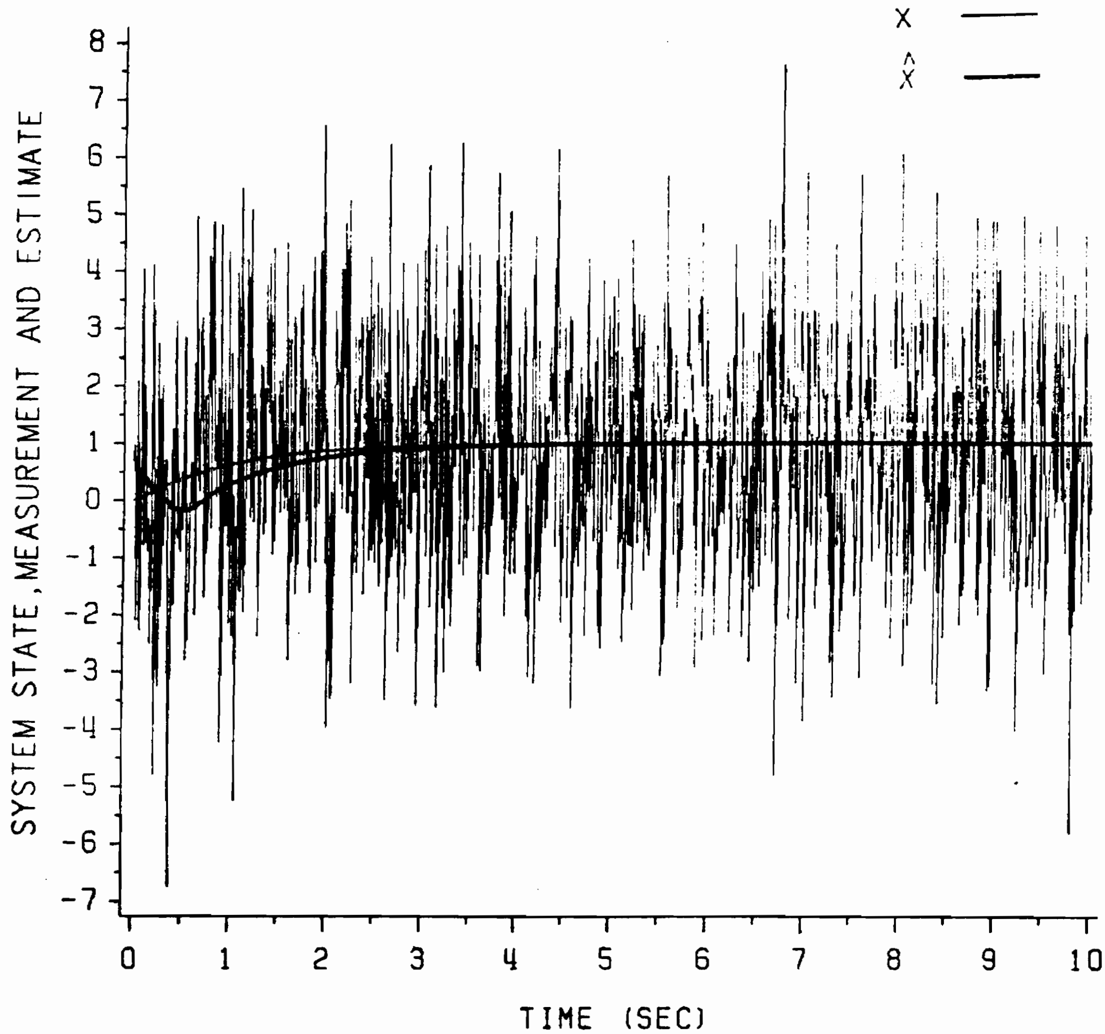

Figure 31. **Relay control system response using Feedback Control (ϕ_{OL} case).**

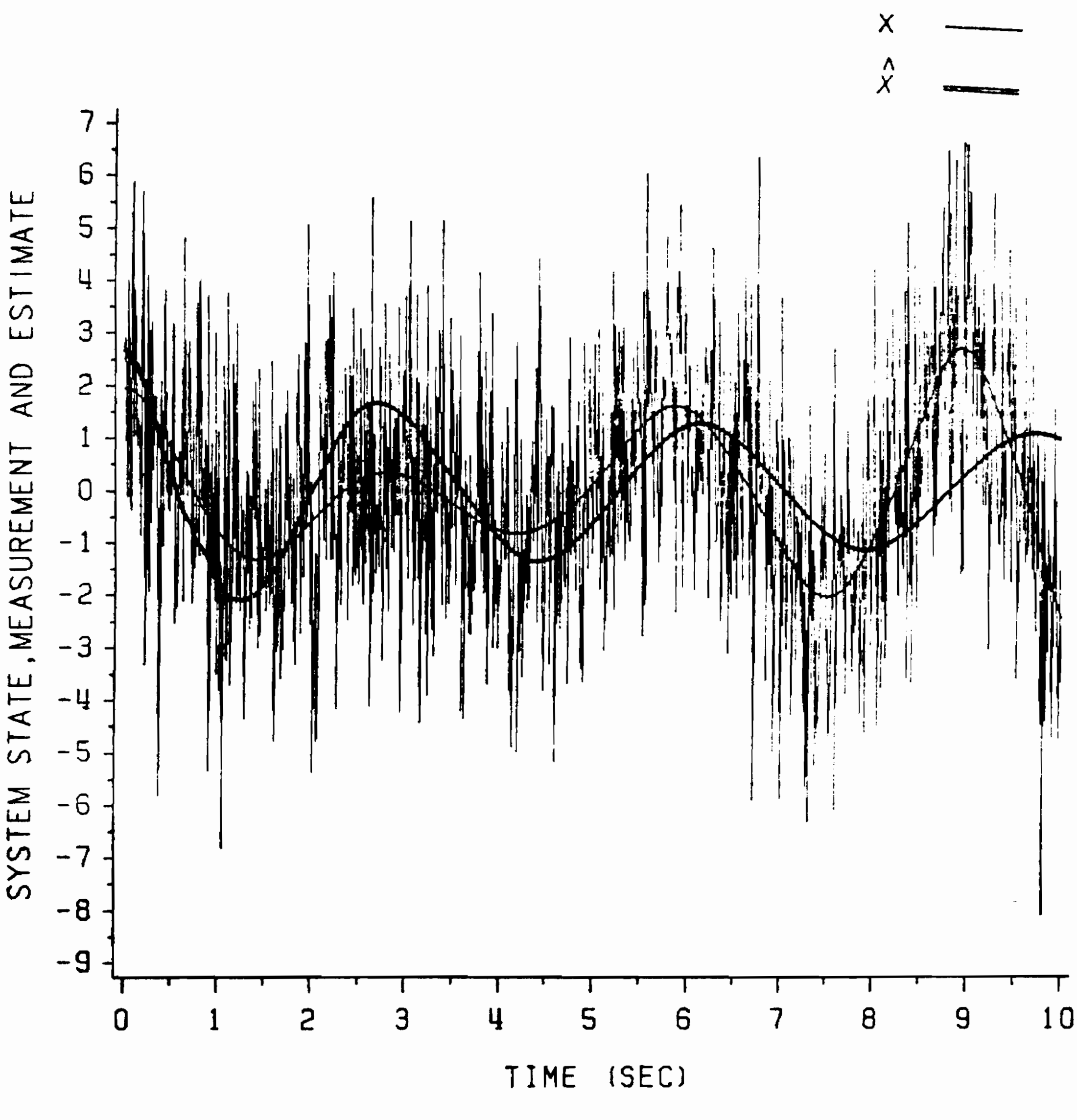

Figure 32. Nonlinear Spring Response using Feedback Control. (ϕ_{CL} case).

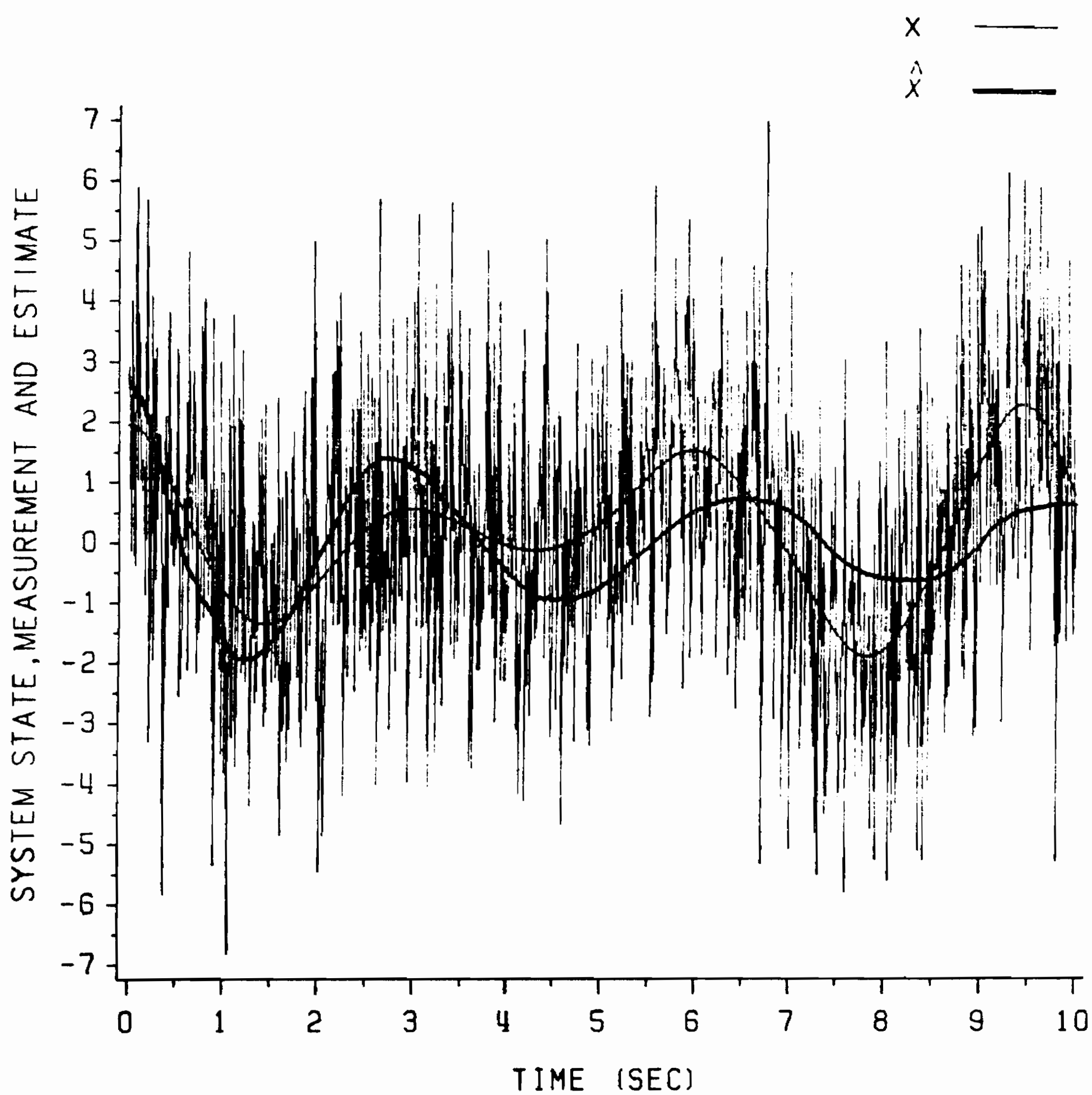

Figure 33. Nonlinear Spring Response using Feedback Control. (ϕ_{OL} case).

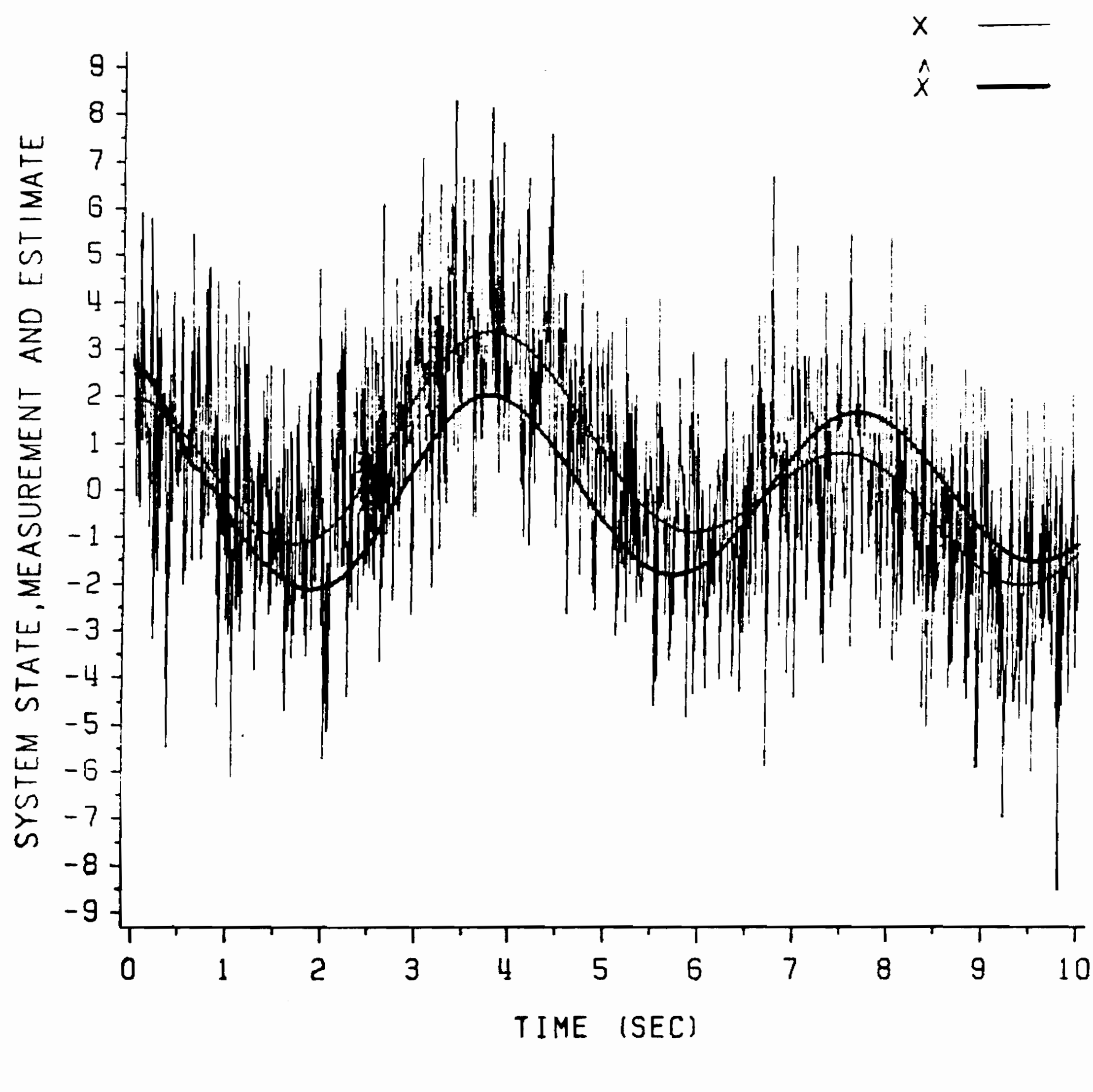

Figure 34. Modified Nonlinear Spring Response using Feedback Control. (ϕ_{CL} case).

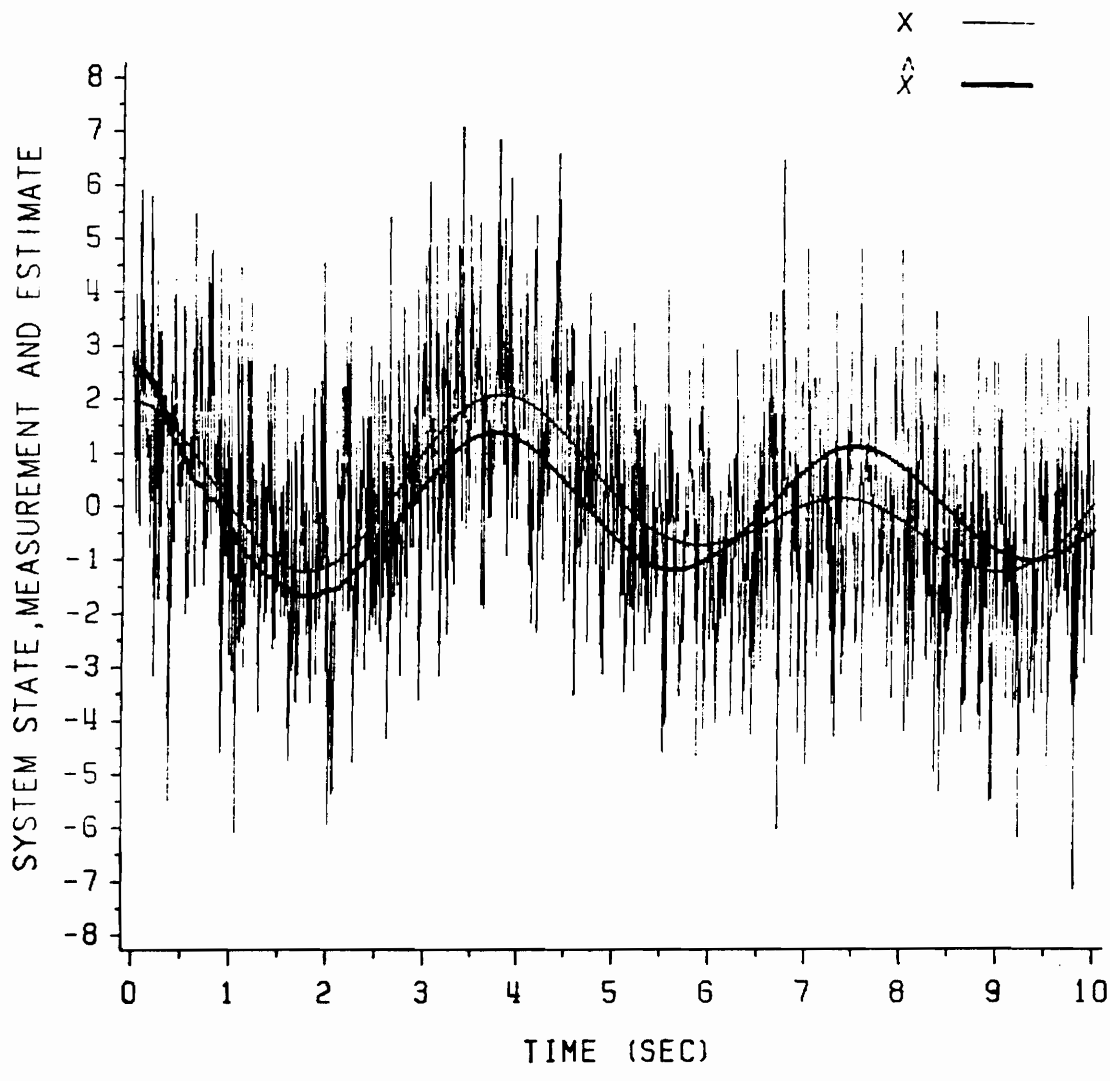

Figure 35. Modified Nonlinear Spring Response using Feedback Control. (ϕ_{OL} case).

CHAPTER V

CONCLUSIONS

The purpose of this thesis was to further investigate the estimation algorithm developed using the Jump Matrix Technique for Nonlinear systems. The basic idea of the technique was to decompose a nonlinear system into linear and nonlinear subsystems. The linear system propagates during the sampling interval as any ordinary linear time invariant system. The system variables were updated at each sampling instant. The linear variables were continuous during the transition at the sampling instant. However, for the case of the nonlinear subsystem state variables, there was an instantaneous "jump" in the value. This "jump" in the system state variables introduces the effect of the nonlinearity in the system and hence the name of the technique. The estimator algorithm is developed on lines of Kalman filter for the linear system case. In the development of the estimator, one basic assumption regarding the relationship of the error in estimated system state before and after the nonlinear jump update is made. This assumption of a timevarying gain matrix a_{k+1} being the proportionality factor between the two errors, is nothing but in effect the quasi-linearization of the nonlinearity. This idea of quasi-linearization of the nonlinearity has been used extensively in related research work, but only in a mathematical sense. In this thesis, the idea was to develop the estimator for ease in system simulation and estimation, and not to get bogged down by mathematical rigor and lose the practical feel of system behavior.

Chapter II dealt with the development of the estimator using the Jump Matrix Technique. On the basis of the assumptions made regarding the time varying gain matrix a_{k+1} the estimator development is mathematically rigorous. Three different ways of calculation of a_{k+1} were discussed. The first assumption of setting a_{k+1} as identity matrix not only reduces computational time but also in effect reduces the quasi-inearization of the nonlinearity to nothing but simple linearization, and hence was not taken up for further investigation. The other two alternatives require the gain to be calculated online at each iterative step and computationally take up comparable time. The significant difference between the two approaches (Online gain approach and the Jacobian approach) as seen from the results obtained in Chapter III was that of less noisy system response in the first case. This can be attributed to the fact that in the Online gain approach, the ratio of the differences is used and thus cancellation of the system noise is reflected in the Kalman gains calculated later. In Chapter III the three different system responses were estimated using the Jump Matrix technique. The effect of low Kalman gains on the estimator behavior presented some interesting results and can be investigated further. A possible solution can be to set some kind of a lower bound on the values of Kalman gain and when the gains went below it, raise them to the value of the lower bound.

Chapter IV dealt with the application of the estimator in Feedback control of nonlinear systems. The idea of controlling the system response using estimated state variables has been worked on before but using the same technique for case of nonlinear systems has lot of potential for future work. The change in system response for the case of spring with nonlinear hardness signifies the potential of the estimator developed using the Jump Matrix Technique.

In summary, the results clearly demonstrate the potential of the estimator developed using the Jump Matrix Technique. Further research could be carried to out to come up with some mathematical validity of the assumption regarding the time varying matrix or, if possible, come up with some other form of the estimator which doesn't use that particular assumption.

CHAPTER VI

BIBLIOGRAPHY

1. Baumann W. T., Rugh W. J., "Feedback control of Nonlinear Systems by Extended Linearization", *IEEE Transactions on Automatic Control* Vol AC-31 No. 1 , Jan 1986, pp 40-46.

2. Beaman J. J., Hedrick J. K., "An Extended Statistical Linearization Technique " *Winter Annual Meeting of the American Soceity of Mechanical Engineers.* New York, NY, Dec 1979.

3. Crandall S. H., "On Statistical Linearization for Nonlinear Oscillators", *Winter Annual Meeting of the American Soceity of Mechanical Engineers.* New York, NY, Dec 1979.

4. Cunningham, Walter J., "Introduction to Nonlinear Analysis", Mcgraw Hill - NY 1958.

5. Godiwala P. M., Moose R. L., "Digital Simulation of Nonlinear systems with deterministic or stochastic inputs", Unpublished paper.

6. Hedrick J. K., Arslan A. V., "Nonlinear analysis of Rail Vehicle Forced Lateral Response and Stability Analysis." *Winter Annual Meeting of the American Soceity of Mechanical Engineers.* New York, NY, Dec 1979.

7. Kuo B. C., "Discrete Data Control Systems", Prentice Hall, Inc. Englewood Cliffs, NJ, 1970.

8. Lauzon T. A., Moose R. L., "An Application of digital simulation to Nonlinear estimation", Project Report, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

9. McGarty T. P., "Stochastic Systems and State Estimation", John Wiley and Sons, 1974.

10. Meditch J. S., "Stochastic Optimal Linear Estimation and Control", Mcgraw Hill, 1969.

11. Papoulis A., "Probability, Random Variables and Stochastic Processes". Mcgraw Hill Book Co., 1965.

12. Ramnath R. V., Paynter H. M., "Scaling Transformations in Nonlinear Systems", *Winter Annual Meeting of the American Soceity of Mechanical Engineers.* New York, NY, Dec 1979.

13. Sontag E. D., "Nonlinear Regulation: The piecewise Linear Approach." *IEEE Transactions on Automatic Control* Vol AC-26 No. 22, April 1981, pp 346-358.

14. Vanlandingham H. F., "Introduction to Digital Control Systems", Macmillan Publishing Company, New York, NY 1985.

15. Vanlandingham H. F., Moose R. L., Lee E. C. W., "Digital Simultaion of Nonlinear Stochastic Systems", Unpublished paper.

16. Yavin Y., "Numerical studies in Nonlinear filtering", Springer Verlag, New York 1985.

17. Yavin Y., "Feedback Strategies for partially observable Stochastic Systems", Springer Verlag, New York, 1983.

Appendix A

KALMAN FILTER EQUATIONS

The discrete time representation of a linear time invariant system is given as :

$$X_{k+1} = \phi X_k + \Gamma U_k + \psi w_k$$

where

X	Vector of state variables.
ϕ	State transition matrix.
U	vector of deterministic inputs.
Γ	Transition matrix corresponding to U.
w	Zero mean Gaussian noise process.
ψ	Input transition matrix.
k	Discrete time parameter.

The measurement process is given as

$$z_{k+1} = hX_k + v_{k+1}$$

where

z	measurement.
h	measurement vector.
v	Zero mean gaussian noise process.

The Kalman gain vector calculation :

$$M_{k+1} = \phi P_k \phi^T + \psi Q \psi^T$$

$$K_{k+1} = M_{k+1} h^T (h M_{k+1} h^T + R)^{-1}$$

$$P_{k+1} = (I - K_{k+1} h) M_{k+1}$$

The estimate equation

$$\hat{X}_{k+1} = \phi \hat{X}_k + \Gamma U_k + K_{k+1} \left[z_{k+1} - h(\phi \hat{X}_k + \Gamma U_k) \right]$$

where

K	Kalman gain vector.
P_k	$E[(X_k - \hat{X}_k)(X_k - \hat{X}_k)^T]$
Q	$E[w.w^T]$.
R	$E[v.v^T]$.

Vita

Jastej S. Dhingra was born in Durgapur, India on May 31, 1965. After twelve years of schooling, he joined the Indian Institute of Technology, Kanpur, for B. Tech. in Electrical Engineering in 1982. He joined Virginia Tech. for a M .S. in Electrical Engineering with specialization in stochatic signal processing. He has been a Graduate Teaching Assistant for the past two years and has recently completed the requirements for the M. S. degree and is going ahead for a Ph. D. in Electrical Engineering starting Fall 88.

He is a member of IEEE and Phi Kappa Phi.

J.S. Dhingra