THESIS REPORT

Master's Degree

Smart Structures and Wavelet Based System Identification

by R. Rezaiifar
Advisor: W.P. Dayawansa

M.S. 93-18

The Institute for Systems Research is supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), the University of Maryland, Harvard University, and Industry

Abstract

Title of Thesis: Smart Structures and Wavelet Based System Identification

Name of degree candidate: Ramin Rezaiifar

Degree and year: Master of Science, 1993

Thesis directed by: Associate Professor W. P. Dayawansa
Department of Electrical Engineering and
Institute for Systems Research

In this thesis we study the problem of modeling and control of vibrations of a flexible structure. Different approaches have been considered for fitting a finite dimensional model to the infinite dimensional system. In particular we used a discretization method to model the cantilever beam and using this model and taking into account the saturation of input signal, we have designed and analyzed a nonlinear controller . An observer based version of the controller has also been implemented and shown to be stable.

A simulation program is written for solving the governing partial differential equation for a cantilever beam and the effect of passive damping due to viscous and internal damping has been considered to get a more realistic simulation.

Finally, a general algorithm for nonparametric system identification is developed and implemented and is used to obtain an approximation to the transfer

function matrix of a smart structure. The results are compared with conventional methods.

Smart Structures and Wavelet Based System Identification

by

Ramin Rezaiifar

Thesis submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Master of Science
1993

Advisory Committee:

Associate Professor W. P. Dayawansa , Chairman/Advisor
Professor P.S. Krishnaprasad (Co-Advisor)
Professor André Tits

Acknowledgements

I would like to sincerely thank my advisors, Dr. W. P. Dayawansa and Dr. P. S. Krishnaprasad for their vision, guidance and patience during this project. I also wish to thank Dr. Y. C. Pati for his guidance especially in developing the third chapter of this work and Dr. Mark A. Austin for answering my questions in strength of materials and related subjects as well as all of the students, faculty, and staff of the Institute of Systems Research for their support and encouragement.

I am also very grateful for the comments and criticisms of Prof. A.L. Tits, the third member of committee.

This research was supported in part by the NSF grant# ECS 9096121, Engineering Research Center Program grant# CD8803012, and the ARO grant# DAAL03-92-G01201.

Contents

List of Tables

List of Figures

Smart Structures and Wavelet Based
System Identification

Ramin Rezaiifar

July 10, 1993

This comment page is not part of the dissertation.

Typeset by LaTeX using the `dissertation` style by Pablo A. Straub, University of Maryland.

Chapter 1

Introduction

Motivated by the rapid developments of problems of design of large flexible space structures, vibration suppression by active modal control has drawn much attention in recent years and has been the subject of much research.

Generally, large spacecraft/space structures have very poor passive damping due to the materials used and also the fact that there might be no air to induce viscous damping. Another characteristic of these space structures is that the natural frequency of the structure (at least the lowest) may fall into the bandwidth of the attitude control system resulting in large amplitude vibrations and fatigue in the whole structure.

Mechanical systems of this type are generally governed by a system of partial differential equations; thus, evolution equations of them may be modeled in infinite dimensional state space.

Despite the recent advances in the theory of distributed-parameter control, (see [19] and [14] for example), it is still a growing field. Also, usually distributed-parameter control requires some sort of distributed sensor which may be costly and difficult to implement. As a result, there is always a tendency toward the approximation of the infinite dimensional system with a finite dimen-

sional one to take advantage of the existing well-developed lumped-parameter control theory. Of course, this approach (truncation/approximation) may entail some performance trade-off.

There has been a significant effort in the field of distributed parameter control with specific application in active vibration damping of flexible structures, e.g. Balas [11] devised a control scheme for a certain class of flexible structures and Curtain and Glover [16] addresses the problem of designing a finite dimensional controller for the control of a distributed-parameter system.

The major goal in the first two chapters (chapters two and three) of this thesis is to exploit some kind of finite dimensional approximation method to obtain a model (an approximation) for a cantilever beam. In chapter two, a Galerkin approximation method is used to model the beam with several rigid rods that are compliantly linked together, and equations of motion are derived based on this model. The rest of the chapter is devoted to obtaining estimates for various parameters that are involved in the model. Once the system of ordinary differential equations has been derived, there is another step of simplification that is needed, namely linearization.

The detailed models of piezoceramic actuated beam can be quite involved and complicated (see [4] for example). Usually several aspects such as stiffness of the bonding material or the way in which the piezo ceramic is mounted (e.g. embedded or bonded) comes into the picture. For example, in a real world application like active vibration damping in a helicopter rotor [17], the helicopter blade cannot be precisely modeled by an Euler-Bernoulli beam and more complicated models should be used that in turn need the knowledge of many physical parameters that are not so easily obtainable.

In chapter three we developed a general non-parametric system identification

tool that uses the idea of rational wavelet decomposition. The matching pursuit algorithm of Mallat et. al. [18] has been applied to the problem of approximation of a given frequency response with a real rational transfer function. This method is then used to obtain a finite dimensional model of the experimental set up that consists of a flexible beam and piezoceramic actuators.

Chapter four addresses the problem of control. First, a classical LQR type controller is employed to control the vibration. This is later used as a basis for comparison with other methods. In a real control set-up, however, there is a limit on the amplitude of the voltage that can be applied safely to the actuating piezoceramics and this saturation limit on control signals may lead to instability or poor performance of the system if one does not take voltage limit into account in the design of the controller. Stability of systems with bounded controls is a topic studied in depth by Sussman et al. [21], Gutman and Hagander [13] and others. Gutman and Hagander proposed a control algorithm to accommodate bounded controls in [13]. In chapter four, a controller of this type has been designed and tested in a simulation program. This type of controller, as presented in [13] needs state variable information to implement the stabilization, a requirement that may not be feasible in most practical situations.

In chapter five, we provide a modification to the Gutman-Hagander algorithm to allow for the usage of estimated state variables obtained by a classical state observer. The proof of stability is also given in chapter five. Although the main goal in this type of controller is stability, an improvement in settling time of the response is observed in comparison to the LQR method.

Chapter six offers an algorithm (implicit finite difference) for finding the numerical solution to the PDE describing the behavior of a cantilever beam with piezo actuator material mounted on it and piezoceramics used as sensors. The

effect of air drag and internal damping is also included in the model. This can serve as a simulation tool to test the performance of control strategies. With the application of spectral analysis method and the simulation program at hand, the empirical frequency response of the cantilever beam with piezoceramic actuators and sensors is obtained and is later used for the purpose of system identification. The final section of chapter six has been devoted to the derivation of an analytical expression for the transfer function of a cantilever beam with bonded piezoceramics. The empirical frequency response obtained through spectral analysis is shown to corroborate the one obtained from the closed form representation of the transfer function.

Chapter seven contains results, some concluding remarks, and suggestions for future work in this area.

Chapter 2

Galerkin Model of a Flexible Cantilever Beam

First, equations of motion of an approximated finite dimensional model for a cantilever beam with embedded piezo actuators are obtained. The model consists of N rigid bodies that are linked together via torsional springs. We use piezoceramics as actuators which apply point moments at the joints. These actuators may have non-uniform spatial distribution. Based on the physical parameters of the beam, some expressions for the stiffness coefficients and moment generators that model the effect of piezoceramics has been obtained. Finally, simulations are also performed to verify the correctness of the model.

2.1 Derivation of Equations of Motion

A flexible beam is an infinite dimensional dynamical system. In order to use the classical control methods for damping the vibration of a flexible cantilever beam, we first need to approximate it with a finite dimensional model.

Consider the following model in which the beam is approximated with N

rigid links, connected to each other by torsional springs. Figure 2.1 shows our discretized model.

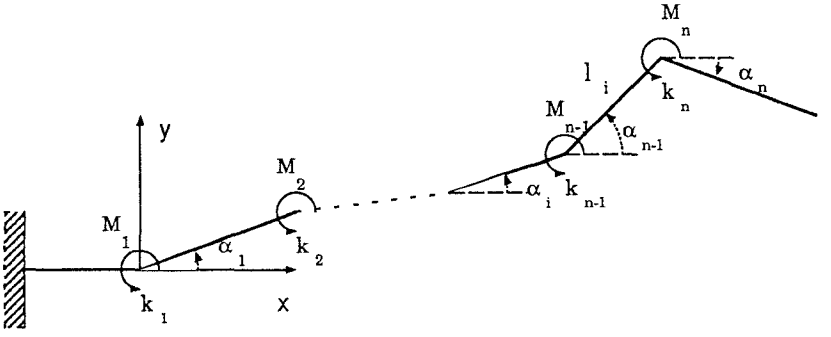

Figure 2.1: Discretized model of the cantilever beam

where,

k_i = Stiffness of the i^{th} spring,

M_i = Moment corresponding to the i^{th} actuator (piezoceramic),

l_i = Length of the i^{th} link,

(x_i, y_i) = Position of the center of mass of the i^{th} link,

α_i = Slope of the i^{th} link.

We use the Euler-Lagrange method to derive the equations of motion,

$$M_k - M_{k+1} = \frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\alpha}_k}\right) - \frac{\partial T}{\partial \alpha_k} + \frac{\partial U}{\partial \alpha_k}, \tag{2.1.1}$$

with $M_{n+1} = 0$. The potential energy, U, and kinetic energy, T, are obtained from the following relations,

$$T = \sum_{i=1}^{N} \frac{1}{2} m_i (\dot{x}_i^2 + \dot{y}_i^2) + \sum_{i=1}^{N} \frac{1}{2} I_i \dot{\alpha}_i^2, \tag{2.1.2}$$

$$U = \sum_{i=0}^{N} \frac{1}{2} k_{i+1} (\alpha_{i+1} - \alpha_i)^2, \tag{2.1.3}$$

$$\alpha_0 \;=\; 0, \quad \alpha_{N+1} = \alpha_N,$$

where,

m_i = Mass of the i^{th} link,

I_i = Moment of inertia of the i^{th} link with respect to its center of mass.

Now, we compute the various terms in the Euler-Lagrange equation,

$$\frac{\partial T}{\partial \dot{\alpha_k}} = \sum_{i=k}^{N} m_i (\sum_{j=1}^{i} \frac{\partial x_i}{\partial \alpha_j}\dot{\alpha_j}) \frac{\partial x_i}{\partial \alpha_k} + \sum_{i=k}^{N} m_i (\sum_{j=1}^{i} \frac{\partial y_i}{\partial \alpha_j}\dot{\alpha_j}) \frac{\partial y_i}{\partial \alpha_k} + I_k \dot{\alpha_k},$$

$$\begin{aligned}
\frac{d}{dt}(\frac{\partial T}{\partial \dot{\alpha_k}}) \;=\; & \sum_{i=1}^{N} [m_i (\sum_{j=1}^{i}\sum_{l=1}^{i} \frac{\partial^2 x_i}{\partial \alpha_j \partial \alpha_l}\dot{\alpha_j}\dot{\alpha_l} + \sum_{j=1}^{i} \frac{\partial x_i}{\partial \alpha_j}\ddot{\alpha_j}) \frac{\partial x_j}{\partial \alpha_k}] \\
& + \sum_{i=1}^{N} [m_i (\sum_{j=1}^{i}\sum_{l=1}^{i} \frac{\partial^2 y_i}{\partial \alpha_j \partial \alpha_l}\dot{\alpha_j}\dot{\alpha_l} + \sum_{j=1}^{i} \frac{\partial y_i}{\partial \alpha_j}\ddot{\alpha_j}) \frac{\partial y_j}{\partial \alpha_k}] \\
& + \sum_{i=1}^{N} [m_i (\sum_{j=1}^{i} \frac{\partial x_i}{\partial \alpha_j}\dot{\alpha_j}) \sum_{l=1}^{i} \frac{\partial^2 x_i}{\partial \alpha_K \partial \alpha_l}\dot{\alpha_l}] \\
& + \sum_{i=1}^{N} [m_i (\sum_{j=1}^{i} \frac{\partial y_i}{\partial \alpha_j}\dot{\alpha_j}) \sum_{l=1}^{i} \frac{\partial^2 y_i}{\partial \alpha_K \partial \alpha_l}\dot{\alpha_l}] + I_k \ddot{\alpha_k}, \qquad (2.1.4)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial U}{\partial \alpha_k} - \frac{\partial T}{\partial \alpha_k} \;=\; & k_k(\alpha_k - \alpha_{k-1}) - k_{k+1}(\alpha_{k+1} - \alpha_k) \\
& - \sum_{i=1}^{N} [m_i (\sum_{j=1}^{i} \frac{\partial x_i}{\partial \alpha_j}\dot{\alpha_j})(\sum_{l=1}^{i} \frac{\partial^2 x_i}{\partial \alpha_k \partial \alpha_l}\dot{\alpha_l})] \\
& - \sum_{i=1}^{N} [m_i (\sum_{j=1}^{i} \frac{\partial y_i}{\partial \alpha_j}\dot{\alpha_j})(\sum_{l=1}^{i} \frac{\partial^2 y_i}{\partial \alpha_k \partial \alpha_l}\dot{\alpha_l})]. \qquad (2.1.5)
\end{aligned}$$

Substituting equations (2.1.4) and (2.1.5) into the Euler-Lagrange equation (2.1.1 we get,

$$M_k - M_{k+1} \;=\; \sum_{i=1}^{N} [m_i (\sum_{j=1}^{i}\sum_{l=1}^{i} \frac{\partial^2 x_i}{\partial \alpha_j \partial \alpha_l}\dot{\alpha_j}\dot{\alpha_l} + \sum_{j=1}^{i} \frac{\partial x_i}{\partial \alpha_j}\ddot{\alpha_j}) \frac{\partial x_j}{\partial \alpha_k}]$$

$$+\sum_{i=1}^{N}[m_i(\sum_{j=1}^{i}\sum_{l=1}^{i}\frac{\partial^2 y_i}{\partial\alpha_j\partial\alpha_l}\dot{\alpha}_j\dot{\alpha}_l+\sum_{j=1}^{i}\frac{\partial y_i}{\partial\alpha_j}\ddot{\alpha}_j)\frac{\partial y_j}{\partial\alpha_k}]$$
$$+I_k\ddot{\alpha}_k+k_k(\alpha_k-\alpha_{k-1})-k_{k+1}(\alpha_{k+1}-\alpha_k). \qquad (2.1.6)$$

the position of the center of mass of the i^{th} link can be obtained from the following relations (assuming a uniform distribution of mass),

$$x_i = \sum_{j=1}^{i-1} l_i\cos\alpha_j+\frac{1}{2}l_i\cos\alpha_i \qquad (2.1.7)$$

$$y_i = \sum_{j=1}^{i-1} l_i\sin\alpha_j+\frac{1}{2}l_i\sin\alpha_i. \qquad (2.1.8)$$

To simplify the equations of motion (2.1.6) we need the partial derivatives of x_i and y_i with respect to α_k i.e.,

$$\frac{\partial x_i}{\partial\alpha_k} = \begin{cases} 0 & i<k \\ -\frac{l_i}{2}\sin\alpha_k & i=k \\ -l_k\sin\alpha_k & k<i \end{cases} \qquad (2.1.9)$$

$$\frac{\partial y_i}{\partial\alpha_k} = \begin{cases} 0 & i<k \\ \frac{l_i}{2}\cos\alpha_k & i=k \\ -l_k\cos\alpha_k & k<i \end{cases} \qquad (2.1.10)$$

$$\frac{\partial^2 x_i}{\partial\alpha_j^2} = \begin{cases} 0 & i<j \\ -\frac{l_i}{2}\cos\alpha_j & i=j \\ -l_j\cos\alpha_j & j<i \end{cases} \qquad (2.1.11)$$

$$\frac{\partial^2 y_i}{\partial \alpha_j^2} = \begin{cases} 0 & i < j \\ -\frac{l_i}{2}\sin\alpha_j & i = j \\ -l_j \sin\alpha_j & j < i \end{cases} . \tag{2.1.12}$$

Now, (2.1.6) reduces to,

$$\begin{aligned} M_k - M_{k+1} &= \sum_{i=k}^{N} m_i [\sum_{j=1}^{i} (\frac{\partial^2 x_i}{\partial \alpha_j^2}\dot{\alpha}_j^2 + \frac{\partial x_i}{\partial \alpha_j}\ddot{\alpha}_j)\frac{\partial x_i}{\partial \alpha_k}] \\ &\quad + \sum_{i=k}^{N} [m_i (\sum_{j=1}^{i} (\frac{\partial^2 y_i}{\partial \alpha_j^2}\dot{\alpha}_j^2 + \frac{\partial y_i}{\partial \alpha_j}\ddot{\alpha}_j)\frac{\partial y_i}{\partial \alpha_k}] \\ &\quad + I_k\ddot{\alpha}_k + k_k(\alpha_k - \alpha_{k-1}) - k_{k+1}(\alpha_{k+1} - \alpha_k) \end{aligned} \tag{2.1.13}$$

$$\begin{aligned} &= \sum_{i=k+1}^{N} m_i [\sum_{j=1}^{i-1} (\frac{\partial^2 x_i}{\partial \alpha_j^2}\dot{\alpha}_j^2 + \frac{\partial x_i}{\partial \alpha_j}\ddot{\alpha}_j)\frac{\partial x_i}{\partial \alpha_k}] \\ &\quad + m_k \sum_{j=1}^{k-1} (\frac{\partial^2 x_k}{\partial \alpha_j^2}\dot{\alpha}_j^2 + \frac{\partial x_k}{\partial \alpha_j}\ddot{\alpha}_j)\frac{\partial x_k}{\partial \alpha_k} \\ &\quad + \sum_{i=k+1}^{N} m_i [(\frac{\partial^2 x_i}{\partial \alpha_i^2}\dot{\alpha}_i^2 + \frac{\partial x_i}{\partial \alpha_i}\ddot{\alpha}_i)\frac{\partial x_i}{\partial \alpha_k}] \\ &\quad + m_k (\frac{\partial^2 x_k}{\partial \alpha_k^2}\dot{\alpha}_k^2 + \frac{\partial x_k}{\partial \alpha_k}\ddot{\alpha}_k)\frac{\partial x_k}{\partial \alpha_k} \\ &\quad + \sum_{i=k+1}^{N} m_i [\sum_{j=1}^{i-1} (\frac{\partial^2 y_i}{\partial \alpha_j^2}\dot{\alpha}_j^2 + \frac{\partial y_i}{\partial \alpha_j}\ddot{\alpha}_j)\frac{\partial y_i}{\partial \alpha_k}] \\ &\quad + m_k \sum_{j=1}^{k-1} (\frac{\partial^2 y_k}{\partial \alpha_j^2}\dot{\alpha}_j^2 + \frac{\partial y_k}{\partial \alpha_j}\ddot{\alpha}_j)\frac{\partial y_k}{\partial \alpha_k} \\ &\quad + \sum_{i=k+1}^{N} m_i [(\frac{\partial^2 y_i}{\partial \alpha_i^2}\dot{\alpha}_i^2 + \frac{\partial y_i}{\partial \alpha_i}\ddot{\alpha}_i)\frac{\partial y_i}{\partial \alpha_k}] \\ &\quad + m_k (\frac{\partial^2 y_k}{\partial \alpha_k^2}\dot{\alpha}_k^2 + \frac{\partial y_k}{\partial \alpha_k}\ddot{\alpha}_k)\frac{\partial y_k}{\partial \alpha_k} \\ &\quad + I_k\ddot{\alpha}_k + k_k(\alpha_k - \alpha_{k-1}) - k_{k+1}(\alpha_{k+1} - \alpha_k), \end{aligned} \tag{2.1.14}$$

$$
\begin{aligned}
M_k - M_{k-1} \;=\; & \sum_{i=k+1}^{N} m_i[\sum_{j=1}^{i-1}(-l_j\cos\alpha_j\dot{\alpha}_j^2 - l_j\sin\alpha_j\ddot{\alpha}_j)(-l_k\sin\alpha_k)] \\
& -l_k\sum_{j=1}^{k-1}(-l_j\cos\alpha_j\dot{\alpha}_j^2 - l_j\sin\alpha_j\ddot{\alpha}_j)(-\frac{l_k}{2}\sin\alpha_k) \\
& +\sum_{i=k+1}^{N} m_i[(-\frac{l_k}{2}\cos\alpha_i\dot{\alpha}_i^2 - \frac{l_i}{2}\sin\alpha_i\ddot{\alpha}_i)(-l_k\sin\alpha_k)] \\
& +m_k(\frac{-l_j}{2}\cos\alpha_k\dot{\alpha}_k^2 - \frac{l_k}{2}\sin\alpha_k\ddot{\alpha}_k)(-\frac{l_k}{2}\sin\alpha_k) \\
& \sum_{i=k+1}^{N} m_i[\sum_{j=1}^{i-1}(-l_j\sin\alpha_j\dot{\alpha}_j^2 + l_j\cos\alpha_j\ddot{\alpha}_j)(+l_k\cos\alpha_k)] \\
& -l_k\sum_{j=1}^{k-1}(-l_j\sin\alpha_j\dot{\alpha}_j^2 + l_j\cos\alpha_j\ddot{\alpha}_j)(\frac{l_k}{2}\cos\alpha_k) \\
& +\sum_{i=k+1}^{N} m_i[(-\frac{l_k}{2}\sin\alpha_i\dot{\alpha}_i^2 + \frac{l_i}{2}\cos\alpha_i\ddot{\alpha}_i)(l_k\cos\alpha_K)] \\
& +m_k(\frac{-l_j}{2}\sin\alpha_k\dot{\alpha}_k^2 + \frac{l_k}{2}\cos\alpha_k\ddot{\alpha}_k)(\frac{l_k}{2}\cos\alpha_k) \\
& +I_k\ddot{\alpha}_k + k_k(\alpha_k - \alpha_{k-1}) - k_{k+1}(\alpha_{k+1} - \alpha_k). \qquad (2.1.15)
\end{aligned}
$$

After some cancellations, the above equation reduces to,

$$
\begin{aligned}
M_k - M_{k+1} \;=\; & \sum_{i=k+1}^{N} m_i[\sum_{j=1}^{i-1} l_j l_k(\sin(\alpha_k - \alpha_j)\dot{\alpha}_j^2 + \cos(\alpha_k - \alpha_j)\ddot{\alpha}_j)] \\
& +\frac{m_k}{2}[\sum_{j=1}^{k-1} l_j l_k(\sin(\alpha_k - \alpha_j)\dot{\alpha}_j^2 + \cos(\alpha_k - \alpha_j)\ddot{\alpha}_j)] \\
& +\sum_{i=k+1}^{N}\frac{m_i}{2}[l_i l_k(\sin(\alpha_k - \alpha_i)\dot{\alpha}_i^2 + \cos(\alpha_k - \alpha_i)\ddot{\alpha}_i)] \\
& +m_k\frac{r_k^2}{4}\ddot{\alpha}_k + I_k\ddot{\alpha}_k + k_k(\alpha_k - \alpha_{k-1}) - k_{k+1}(\alpha_{k+1} - \alpha_k), \\
& \qquad k \;=\; 1, \cdots, N. \qquad (2.1.16)
\end{aligned}
$$

These, are the equations of motion for the discretized model.

In order to apply classical control strategies we will have to linearize them about the stationary equilibrium point $\alpha_k = \dot{\alpha_k} = 0, \quad k = 1, \cdots, N$.

The linearized equations are:

$$\begin{aligned} M_k - M_{k+1} &= \sum_{i=k+1}^{N} \rho l_i \sum_{j=1}^{i-1} l_i l_K \ddot{\alpha_j} + \frac{1}{2}\rho l_k \sum_{j=1}^{k-1} l_j l_k \ddot{\alpha_j} + \frac{1}{2}\sum_{i=k+1}^{N} \rho l_i^2 l_k \ddot{\alpha_i} \\ &+ \frac{1}{3}\rho l_k^3 \ddot{\alpha_k} + k_k(\alpha_k - \alpha_{k-1}) - k_{k+1}(\alpha_{k+1} - \alpha_k), \end{aligned}$$

where ρ is the mass density per unit of length of the beam and we made the following substitutions for the moment of inertia, I_k, and mass, m_k, of the k^{th} link respectively,

$$\begin{aligned} I_k &= \frac{1}{12}\rho l_k^3, \\ m_i &= \rho l_i. \end{aligned}$$

2.2 An Estimate For the Stiffness Constant of the k^{th} Link

Up to this point we derived the equations of motion for our model. The next step is to compute the various constants that appear in (2.1.17) . One of these constants is the stiffness of the k^{th} link. We have to decide on some criterion for computing these constants. One reasonable criterion might be to compute k_i such that the deflection of the tip of the beam at the steady state due to a force F applied at its free end is the same for both the model and the actual flexible beam.

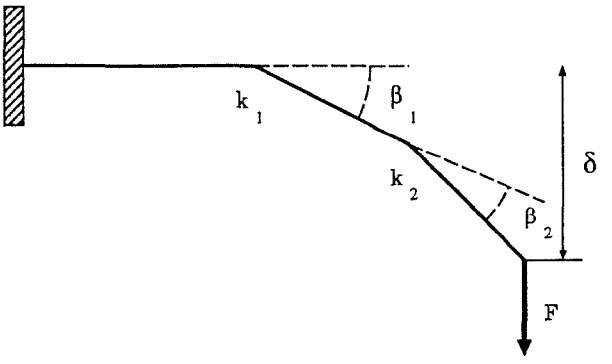

Figure 2.2: Deflection of the beam due to a force F applied at its free end

The deflection of a cantilever beam due to a force F at its free end is [5],

$$\delta = \frac{FL^3}{3EI}, \tag{2.2.1}$$

where, L is the length of the beam and E is the modulus of elasticity. For our N-link model and for small β_i in the static situation we can write,

$$\delta = \sum_{i=1}^{N}(\sum_{j=1}^{i}\beta_j)l_i, \tag{2.2.2}$$

$$k_i\beta_i = F\sum_{j=i}^{N} l_j. \tag{2.2.3}$$

Eliminating β_i from the above two equations and using (2.2.1), we get,

$$\sum_{i=1}^{N}\sum_{j=1}^{i}\sum_{m=j}^{N}\frac{l_m l_i}{k_j} = \frac{L^3}{3EI}. \tag{2.2.4}$$

We also assume that the compliance of the torsional spring corresponding to each link is proportional to its length. i.e.,

$$k_i l_i = c \qquad i = 1, \cdots, N, \tag{2.2.5}$$

where, c is a constant. Combining (2.2.4) and (2.2.5) and then solving for c we get,

$$\sum_{i=1}^{N}\sum_{j=1}^{i}\sum_{m=j}^{N}\frac{l_m l_i l_j}{c} = \frac{L^3}{3EI},$$

$$c = \frac{3EI}{L^3}\sum_{i=1}^{N}\sum_{j=1}^{i}\sum_{m=j}^{N} l_m l_i l_j. \tag{2.2.6}$$

Alternatively, one may consider a different criterion for estimating the stiffness coefficients: find k_i such that the potential energy stored in the model is the same as that of the flexible beam. The potential energy stored in torsional springs is,

$$U = \sum_{i=1}^{N}\frac{1}{2}k_i\beta_i^2. \tag{2.2.7}$$

The potential energy stored in the flexible beam is obtained from the following equation:

$$U = \frac{1}{2}\int_0^L EI(\frac{\partial^2 w}{\partial x^2})^2 dx. \tag{2.2.8}$$

Again, it can be shown that the transversal deflection of the flexible beam due to a force F applied at its free end is [5],

$$w(x) = (2L^3 - 3L^2x + x^3)\frac{F}{6EI} \tag{2.2.9}$$

where x is the distance from the free end. Substituting for $w(x)$ from (2.2.9) into (2.2.8), results:

$$U = \tfrac{1}{2}\textstyle\int_0^L EI\frac{F^2}{36(EI)^2}[\frac{\partial^2}{\partial x^2}(2L^3 - 3L^2x + x^3)]^2,$$

$$= \tfrac{F^2L^3}{6EI}.$$

Now we can equate (2.2.10) and (2.2.7) to get,

$$\frac{F^2L^3}{6EI} = \sum_{i=1}^{N} \frac{1}{2} k_i \beta_i^2. \tag{2.2.10}$$

Substitute for β_i from (2.2.3) into the right hand side of the above equation:

$$\frac{F^2L^3}{6EI} = \frac{1}{2} \sum_{i=1}^{N} k_i [\frac{F}{k_i} \sum_{j=i}^{N} l_j]^2, \tag{2.2.11}$$

and finally we can eliminate k_i from the above equation and (2.2.5 to obtain the constant c ,

$$\frac{L^3}{3EI} = \sum_{i=1}^{N} \frac{l_i}{c} (\sum_{j=i}^{N} l_j)^2.$$

Thus,

$$c = \frac{3EI}{L^3} \sum_{i=1}^{N} l_i (\sum_{j=i}^{N} l_j)^2 \tag{2.2.12}$$

Remark: It can be easily shown that the constant c goes to the value EI as the number of rigid links, N goes to infinity while the length L remains constant.

2.3 Relation Between M_i and the Voltage Applied to the Piezoceramic

We try to find an expression for M_i in terms of V_i , the voltage that applied to the i^{th} piezoceramic. We compute M_i such that at the static situation it results in the same amount of deflection at the tip of each link as the amount of

deflection caused by applying the voltage V_i to the i^{th} segment of the piezo material. In the steady state, the following relation holds for the actual flexible beam,

$$\frac{\partial^2 w}{\partial x^2} EI = c_p V \tag{2.3.1}$$

where, w is the deflection and c_P is a constant relating the voltage applied to the piezo material to the moment that it produces [20]. Integrating the above equation with respect to x two times we get,

$$\begin{aligned} \delta &\equiv w(l_i), \\ &= \frac{c_p}{EI} \int_0^{l_i} (\int_0^{\sigma} V(\tau) d\tau) d\sigma, \\ &= \frac{c_p}{2EI} l_i^2 V_i. \end{aligned} \tag{2.3.2}$$

For the discretized model we have,

$$M_i = k_i \frac{c_p l_i V_i}{2EI}, \tag{2.3.3}$$

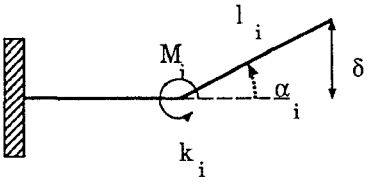

Figure 2.3: Deflection resulting from the moment M_i

where, k_i has already been computed.

2.4 Initial State Computation

Suppose our model for the cantilever beam is subjected to a force F at its free end. In order to perform some simulations we need the values of $\alpha_i, \quad i = 1, \cdots, N$ as the initial state for the dynamical equations of motion. From (2.2.1) and (2.2.3) we have:

$$\beta_i = \frac{F}{k_i} \sum_{j=1}^{N} l_j, \tag{2.4.1}$$

$$\delta = \frac{FL^3}{3EI}, \tag{2.4.2}$$

which give us β_i as a function of δ (deflection at the tip),

$$\beta_i = \frac{1}{k_i} \frac{3EI\delta}{L^3} \sum_{j=i}^{N} l_j \quad i = 1, \cdots, N, \tag{2.4.3}$$

and α_i can be computed using the following relation,

$$\begin{aligned} \alpha_i &= \textstyle\sum_{j=1}^{i} \beta_i, \\ &= \frac{3EI\delta}{L^3} \textstyle\sum_{m=1}^{i} \frac{1}{k_m} \sum_{j=m}^{N} l_j \qquad i = 1, \cdots, N. \end{aligned}$$

2.5 Simulation

Based on equation (2.1.17) the program 'EqOfMotion.m' has been written to derive the matrices A and B in the state space representation,

$$\dot{x} = Ax + Bu, \tag{2.5.1}$$

which we need for simulation and control. *EqOfMotion.m* is a *Mathematica* macro. As input, this macro accepts $n \equiv$ number of links and physical constants

of the beam and in return it gives the numerical matrices A, B, and x_0. The output of the program is stored on a text file named *AandB.m* which is in a format that is readable by *MATLAB*. Given those two matrices a simulation has been performed that shows the free oscillation of the beam when it is released from the initial rest with $\delta = 4\ cm$.

The results are shown on figure 2.4 and figure 2.5 for $N = 3$ and $N = 4$ respectively. The upper graphs are the slopes of each rigid link versus time and the lower ones shows the time derivative of the slope of the rigid links. Each curve corresponds to one link.

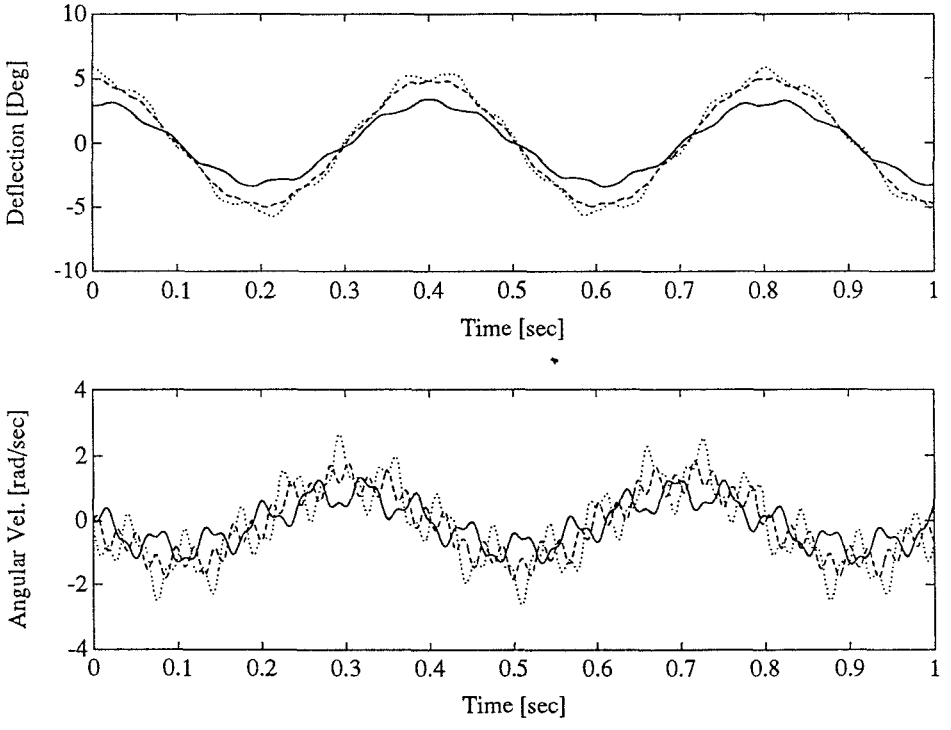

Figure 2.4: Free oscillation of the beam $N = 3$

One may compare the lowest mode of oscillation of the Galerkin model obtained above (which is easily readable from the deflection curves) and compare it with the one that can be obtained analytically by solving the Euler-Bernoulli

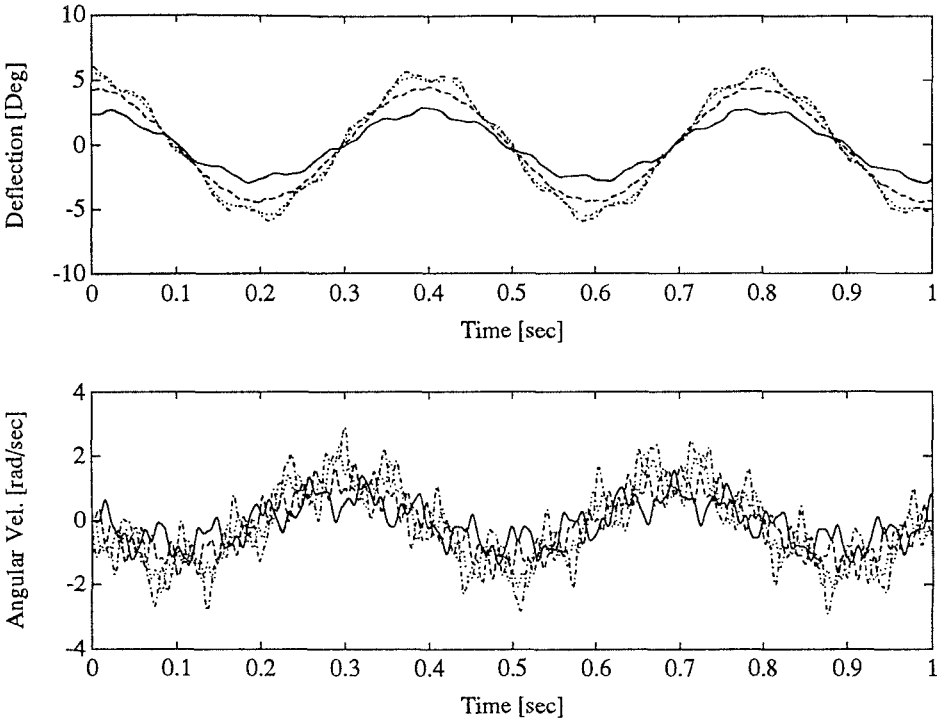

Figure 2.5: Free oscillation of the beam $N = 4$

equation to verify the correctness of the model and simulation. In the next chapter we follow a different algorithm to get a finite dimensional model for the flexible beam.

Chapter 3

Matching Pursuit with Rational Wavelets for Nonparametric Estimation of Stable Systems

The matching pursuit algorithm developed by S. Mallat and Z. Zhang [18] has been exploited to obtain a rational transfer function approximation to a set of empirical data obtained from say, an experiment, or even to reduce the order of a transfer function which is real-rational already. In this chapter we follow closely the work done by Y. C. Pati in his Ph.D. thesis [15], which basically addresses the same problem but with a different solution. Using this method, a priori knowledge of time-frequency localization properties of the given transfer function enables us to obtain a "good" model for the physical system under consideration. One advantage of this algorithm is its simplicity which in turn makes it fast (e.g. compared to the least squares method discussed in [15]). As a practical application of this, we considered the problem of finding a finite dimensional approximation to the transfer function of flexible beam with

[0]This work has been done under the supervision of Dr. Y. C. Pati.

piezoceramic actuators. The performance of the matching pursuit algorithm is compared to the Laguerre method and also the one developed in [15].

3.1 Introduction

The nonparametric estimation problem has been addressed in many ways and different methods have been devised to solve the problem. Obviously one always seeks a low order approximation while achieving a certain degree of accuracy. It is also desirable to obtain an approximation that can be easily realized. We use the algorithm developed by S. Mallat and Z. Zhang [18] that decomposes any signal into a linear combination of waveforms that belong to a redundant dictionary of functions.

Clearly, the choice of the dictionary has a great deal to do with how compact the decomposition is going to be. For the system identification problem one of the requirements is the real-rationality of the approximation (a real-rational function is a rational one with real coefficients). At least this is important when we want to use the method to represent an infinite dimensional system with a finite dimensional one.

We also may use our a priori knowledge of time-frequency localization properties of the signal to select the elements of the dictionary in an efficient fashion.

Motivated by these two, rational wavelets seem to be a good choice for the purpose of decomposition of a transfer function because:

- 1. They are rational.
- 2. They capture the localization properties of the signal.

- 3. It is easy to realize a wavelet decomposition of the signal (elements of the dictionary are just translation and dilation of the mother wavelet.)).

Of course, the fact that the approximation given by this method is *real*-rational is something that must be proven and we will show that in section 3.

One thing that should be pointed out is that although a poor choice will affect the compactness and the degree of accuracy, the elements of the dictionary can be virtually any set of functions and there are no constraints such as orthogonality on them. The only requirement is that they must be unit normed.

3.2 Preliminaries

Before introducing the algorithm, we need to develop some notations which are almost standard but yet, for the sake of completeness, worth mentioning. The inner product of two signals f, and g in $\mathrm{L}^2(R)$ is defined to be

$$\langle f,\ g\rangle\ =\ \int_{-\infty}^{+\infty} f(t)\overline{g}(t)dt,$$

which leads to the usual L^2 norm,

$$\|f\|^2\ =\ \int_{-\infty}^{+\infty} \|f(t)\|^2 dt.$$

If $\Phi(\omega)$ is the analyzing wavelet $\Phi_{m,n}(s)$ is defined by,

$$\Phi_{m,n}(s)\ =\ a_0^{m/2}\Phi(a_0^m s - inb_0), \tag{3.2.1}$$

where a_0 and b_0 are dilation and translation step sizes, respectively and m and n are called dilation and translation levels. Note that it is necessary to put the

imaginary number i in (3.2.1) for n to be the translation index. To see this, let us substitute $i\omega$ for s in (3.2.1),

$$\begin{aligned}\Phi_{m,n}(i\omega) &= a_0^{m/2}\Phi(a_0^m i\omega - inb_0) \\ &= a_0^{m/2}\Phi(ia_0^m(\omega - a_0^{-m}nb_0)).\end{aligned}$$

3.3 The Algorithm

First, we describe the algorithm in its general form and its application to rational transfer functions follows as a special case. In the initialization part of the algorithm, a set of translation and dilation levels has to be chosen that is referred to as dictionary. This is the set from which the algorithm select the translation and dilation levels. The apriori knowledge of time-frequency localization of the signal should help to choose this set. Here the objective is to approximate f, an element of the Hilbert space, by a linear combination of elements of the dictionary.

- *Step 0:* Select the analyzing wavelet Φ with norm equal to one.
- *Step 1:* Choose the finite set $\mathbf{D}$ of translation and dilation levels.
- *Step 2:* Set

$$R^0 f = f; \qquad i = 1.$$

- *Step 3:* Find indices $(m_i, n_i) \in \mathbf{D}$ such that,

$$|\langle R^{i-1}f,\ \Phi_{m_i,n_i}\rangle| \geq |\langle R^{i-1}f,\ \Phi_{m,n}\rangle| \qquad (3.3.1)$$
$$\forall (m,n) \in \mathbf{D}.$$

- *Step 4:* Compute the approximation at level i via,

$$f_i = \sum_{k=0}^{i} \langle R^{k-1} f,\ \Phi_{m_k,n_k} \rangle \Phi_{m_k,n_k}. \tag{3.3.2}$$

If satisfactory, stop. Otherwise continue.

- *Step 5:* Compute,

$$R^i f = R^{i-1} f - \langle R^{i-1} f,\ \Phi_{m_i,n_i} \rangle \Phi_{m_i,n_i},$$

increment i, and go to step 3.

3.4 The Theory Behind The Algorithm

First note that the residuals satisfy the following recursive equation,

$$R^n f = \langle R^n f,\ \Phi_{\gamma_n} \rangle, \Phi_{\gamma_n} + R^{n+1} f. \tag{3.4.1}$$

where, Φ_{γ_n} is the selected wavelet at the n^{th} iteration. Evaluating the inner product on both sides with Φ_{γ_n} and noting that Φ_{γ_n} is normed one, proves that $R^{n+1} f$ is orthogonal to Φ_{γ_n} which in turn enables us to write,

$$\|R_n f\|^2 = |\langle R^n f,\ \Phi_{\gamma_n} \rangle|^2 + \|R^{n+1} f\|^2. \tag{3.4.2}$$

This shows that the residual $R^n f$ is monotonically decreasing. In fact the following theorem states that the series:

$$f = \sum_{n=0}^{\infty} \langle R^n f,\ \Phi_{\gamma_n} \rangle \Phi_{\gamma_n} \tag{3.4.3}$$

converges to the projection of f into the subspace spanned by the dictionary.

Theorem [18]: Let $\mathbf{H}$ be a Hilbert space and $f \in \mathbf{H}$. The residual $R^m f$ defined by the recursion (3.4.1) satisfies

$$\lim_{m\to\infty} \|R^m f - \mathbf{P}_W f\| = 0. \tag{3.4.4}$$

Hence

$$\mathbf{P}_V f \;=\; \sum_{n=0}^{\infty} \langle R^n f,\ \Phi_{\gamma_n}\rangle \Phi_{\gamma_n}, \tag{3.4.5}$$

and

$$\|\mathbf{P}_V f\|^2 \;=\; \sum_{n=0}^{\infty} |\langle R^n f,\ \Phi_{\gamma_n}\rangle|^2. \tag{3.4.6}$$

Where, V is the space spanned by the vectors in D, W is its orthogonal complement in $\mathbf{H}$, and $\mathbf{P}_V$ and $\mathbf{P}_W$ are orthogonal projections of f into V and W respectively. When V =$\mathbf{H}$, we have $\mathbf{P}_V f = f$ and the matching pursuit recovers f.

But, if we want to use this method to approximate a stable transfer function F with real coefficients there is one more thing that needs to be shown. That is, we have to show that the projection of F into the space V is a real rational function. Formally, the space of transfer functions of stable systems is the Hardy space $\mathrm{H}^2(\Pi^+))$ where Π^+ is the right half plane . Also, $\mathrm{RH}^2(\Pi^+)$ is a subset of $\mathrm{H}^2(\Pi^+))$ that consists of those elements that are rational functions in s with real coefficients. So, we need to prove that $\mathbf{P}_V f = f \in \mathrm{RH}^2(\Pi^+)$.

Rationality is not a problem because the analyzing wavelet is rational to begin with (we choose it to be this way). But note that $\Phi_{m,n}(s) = a_0^{m/2}\Phi(a_0^m s -$

inb_0) is not the Laplace transform of a real valued function (unless $n = 0$). It can be easily shown that [15] $G^{m,n}(s)$ defined as,

$$G^{m,n}(s) \;=\; \alpha\Phi_{m,n}(s) + \overline{\alpha}\Phi_{m,-n}(s)$$

is a real rational function in $\mathrm{H}^2(\Pi^+)$.

Thus, to guarantee the real rationality, for every wavelet $\Phi_{m,n}$ in the space V, corresponding wavelet with the translation index $-n$ should be present in V as well. In fact by doing this, the matching pursuit process picks the coefficients corresponding to wavelets $\Phi_{m,n}$ and $\Phi_{m,-n}$ complex conjugate of each other to make $\mathbf{P}_V f$ real rational.

Hereafter, we use $\Phi(.)$ instead of $\Phi(i.)$ to simplify the notation. Recall that if the inverse Fourier transform of $F(\omega)$ is a real valued function then,

$$F(-\omega) \;=\; \overline{F(\omega)}. \tag{3.4.7}$$

We will use this in the proof of the following proposition.

Proposition 1 The following relation holds,

$$\langle F,\ \Phi_{m,-n}\rangle \;=\; \overline{\langle F,\ \Phi_{m,n}\rangle}. \tag{3.4.8}$$

Proof:

$$\begin{aligned}
\langle F,\ \Phi_{m,n}\rangle &= \int_{-\infty}^{+\infty} F(\omega)\overline{a_0^{m/2}\Phi(a_0^m\omega - nb_0)} \\
&= \int_{-\infty}^{+\infty} F(\omega)a_0^{m/2}\Phi(nb_0 - a_0^m\omega) \\
&= (F * \Phi_{m,0})(nb_0/a_0^m) \\
&= \mathcal{F}[f(t)\phi_{m,0}(t)]|_{\omega=nb_0a_0^{-m}}.
\end{aligned}$$

Similarly,

$$\langle F,\ \Phi_{m,-n}\rangle\ =\ \mathcal{F}[f(t)\phi_{m,0}(t)]]|_{\omega=-nb_0a_0^{-m}}. \tag{3.4.9}$$

Since $f(t)$ and $\phi_{m,0}(t)$ are both real we have,

$$\mathcal{F}[f(t)\phi_{m,0}(t)](-\omega)\ =\ \overline{\mathcal{F}[f(t)\phi_{m,0}(t)](\omega)},$$

and this proves the statement of the proposition.□

The next proposition is what we need to justify that the approximation given by the matching pursuit algorithm converges to a real rational function.

Proposition 2 Let V be such that $\Phi_{m,n} \in V$ implies that $\Phi_{m,-n} \in V$ for all $m, n \in D$, then $\mathbf{P}_V f$ is real rational.

Proof: Since $\mathbf{P}_V f$ is the orthogonal projection of f into V we have,

$$\langle f - \mathbf{P}_V f,\ g\rangle\ =\ 0, \qquad \forall g \in V.$$

In particular the following equalities hold:

$$\langle f(\omega),\ \Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)\rangle = \langle \mathbf{P}_V f(\omega),\ \Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)\rangle \tag{3.4.10}$$

$$\langle f(\omega),\ \Phi_{m,n}(\omega) - \Phi_{m,-n}(\omega)\rangle = \langle \mathbf{P}_V f(\omega),\ \Phi_{m,n}(\omega) - \Phi_{m,-n}(\omega)\rangle \tag{3.4.11}$$

From (3.4.7) we see that $\Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)$ is a real rational function and by (3.4.7) we can write,

$$\Phi_{m,n}(-\omega) + \Phi_{m,-n}(-\omega)\ =\ \overline{\Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)}. \tag{3.4.12}$$

Replace ω by $-\omega$ in (3.4.11) and use (3.4.12) and the fact that $f(\omega)$ has a real valued weighting pattern (therefore $f(-\omega) = \overline{f(\omega)}$), to get

$$\begin{aligned}\langle \mathbf{P}_V f(-\omega),\ \Phi_{m,n}(-\omega) + \Phi_{m,-n}(-\omega)\rangle &= \langle f(-\omega),\ \Phi_{m,n}(-\omega) + \Phi_{m,-n}(-\omega)\rangle \\ &= \langle \overline{f(\omega)},\ \overline{\Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)}\rangle \\ &= \overline{\langle f(\omega),\ \Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)\rangle} \\ &= \langle \overline{\mathbf{P}_V f(\omega)},\ \overline{\Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)}\rangle.\end{aligned} \tag{3.4.13}$$

Therefore,

$$\langle \mathbf{P}_V f(-\omega) - \overline{\mathbf{P}_V f(\omega)},\ \overline{\Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)}\rangle = 0, \tag{3.4.14}$$

or, after taking complex conjugate,

$$\langle \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega),\ \Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)\rangle = 0 \qquad \forall\ (m,n) \in \mathbf{D}. \tag{3.4.15}$$

It can be shown that the following holds:

$$\Phi_{m,n}(-\omega) - \Phi_{m,-n}(-\omega) = -\overline{\Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)}. \tag{3.4.16}$$

Starting with (3.4.11) and taking the same steps as we took for (3.4.11), we obtain

$$\langle \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega),\ \Phi_{m,n}(\omega) - \Phi_{m,-n}(\omega)\rangle = 0, \qquad \forall\ (m,n) \in \mathbf{D}. \tag{3.4.17}$$

Add equations (3.4.16) and (3.4.18) up to get

$$\langle \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega),\ \Phi_{m,n}(\omega) \rangle \ = \ 0, \qquad \forall\ (m,n) \in \ \mathbf{D}. \tag{3.4.18}$$

Equations (3.4.16) and (3.4.18) can also be rewritten as

$$\langle \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega),\ \overline{\Phi_{m,n}(-\omega) + \Phi_{m,-n}(-\omega)} \rangle = 0, \qquad \forall\ (m,n) \in \ \mathbf{D}, \tag{3.4.19}$$

$$\langle \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega),\ \overline{\Phi_{m,n}(-\omega) - \Phi_{m,-n}(-\omega)} \rangle = 0, \qquad \forall\ (m,n) \in \ \mathbf{D}, \tag{3.4.20}$$

which are obtained by taking the complex conjugate once and using real rationality of $\Phi_{m,n}(\omega) + \Phi_{m,-n}(\omega)$ and equation (3.4.16). Again, add equations (3.4.20) and (3.4.21) up to obtain

$$\langle \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega),\ \overline{\Phi_{m,n}(-\omega)} \rangle \ = \ 0, \qquad \forall\ (m,n) \in \ \mathbf{D}. \tag{3.4.21}$$

Notice that $\mathbf{P}_V f$ and $\overline{\mathbf{P}_V f(-\omega)}$ can be expressed as a linear combination of vectors in V. i.e.

$$\begin{aligned} \mathbf{P}_V f(\omega) \quad &= \textstyle\sum_{i,j \in I} \Phi_{i,j}(\omega), \\ \overline{\mathbf{P}_V f(-\omega)} \quad &= \textstyle\sum_{i,j \in I} \overline{\Phi_{i,j}(-\omega)} \qquad I \subset D. \end{aligned} \tag{3.4.22}$$

From equations (3.4.22), (3.4.19), and (3.4.22) we conclude that,

$$\langle \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega),\ \overline{\mathbf{P}_V f(-\omega)} - \mathbf{P}_V f(\omega) \rangle \ = \ 0. \qquad (3.4.23)$$

Hence,

$$\overline{\mathbf{P}_V f(-\omega)} \ = \ \mathbf{P}_V f(\omega) \quad \textit{almost everywhere.} \qquad (3.4.24)$$

which together with continuity of $\mathbf{P}_V f(\omega)$ results that the equality holds everywhere. This is the necessary and sufficient condition for $\mathbf{P}_V f$ having a real-valued weighting pattern.□

3.5 Computational Aspects

There are a few points worth noting that make the algorithm more efficient as far as storage and speed is concerned. First note that in order to choose the indices (m_i, n_i) in the third step, and also to compute the approximation at step 4, we only need the inner product of the residuals with the wavelets (and not the residuals). Thus, in step 5 one may directly compute the inner products instead. i.e.

$$\langle R^i f,\ \Phi_{m,n} \rangle \ = \ \langle R^{i-1} f, \Phi_{m,n} \rangle - \langle R^{i-1} f,\ \Phi_{m_i,n_i} \rangle \langle \Phi_{m_i,n_i},\ \Phi_{m,n} \rangle. \qquad (3.5.1)$$

Note that by doing this we actually do not have to do any integration for computing the inner products (other than just one set of integration to get $\langle f,\ \Phi_{m,n} \rangle$). As far as storage is concerned, note that at each step we only have to keep one set of inner products of the residuals with the wavelets and as we proceed, we can throw away the previous ones.

The third point is that the correlation matrix (with the elements $\langle \Phi_{k,l}, \Phi_{m,n} \rangle$) can be computed a priori and in fact this can be done analytically:

$$\begin{aligned} \langle \Phi_{m_1,n_1}(\omega),\ \Phi_{m_2,n_2}(\omega) \rangle &= \int \Phi_{m_1,n_1}(\omega) a_0^{m_2/2} \overline{\Phi(\omega a_0^{m_2} - n_2 b_0)} d\omega \\ &= a_0^{m_2/2} \int \Phi_{m_1,n_1}(\omega) \Phi(n_2 b_0 - \omega_0^{m_2}) d\omega \\ &= (\Phi_{m_1,n_1} * \Phi_{m_2,n_2})|_{\omega = \frac{n_2 b_0}{a_0^{m_2}}} \\ &= \mathcal{F}(\phi_{m_1,n_1}(t) \phi_{m_2,0}(t))|_{\omega = \frac{n_2 b_0}{a_0^{m_2}}}. \end{aligned} \tag{3.5.2}$$

Also, $\phi_{m,n}(t)$ can be written in terms of $\phi(t)$ as,

$$\phi_{m,n}(t) = a_0^{-m/2} e^{\frac{nb_0}{a_0^M} it} \phi(\frac{t}{a_0^m}). \tag{3.5.3}$$

Substitute for $\phi_{m,n}(t)$ from (3.5.3) into (3.5.2) to get,

$$\langle \Phi_{m_1,n_1}(\omega),\ \Phi_{m_2,n_2}(\omega) \rangle = a_0^{\frac{-(m_1+m_2)}{2}} \mathcal{F}(e^{\frac{n_1 b_0}{a_0^{m_1}} it} \phi(\frac{t}{a_0^{m_1}}) \phi(\frac{t}{a_0^{m_2}}))|_{\omega = \frac{n_2 b_0}{a_0^{m_2}} - \frac{n_1 b_0}{a_0^{m_1}}} \tag{3.5.4}$$

3.6 Results

The matching pursuit scheme has been used to obtain the approximation to several frequency responses and in each case we compared its performance with the Laguerre method and wavelet decomposition using least square [15]. As an example consider the second order system with delay which has been examined in [15]. The transfer function for this system is

$$H(s) = \frac{4.94 e^{-2s}}{s^2 + 1.25s + 0.406}$$

Figure (3.1) shows the magnitude of the wavelet coefficients obtained from the matching pursuit algorithm. The analyzing wavelet is again given in Eq.(6.8.1). As it can be seen in the figure, very few coefficients have a significant magnitude and this is a measure of how well the frequency response/impulse response is

localized in time-frequency domain. This localization property, in turn, enables us to get a fairly low degree for the approximation.

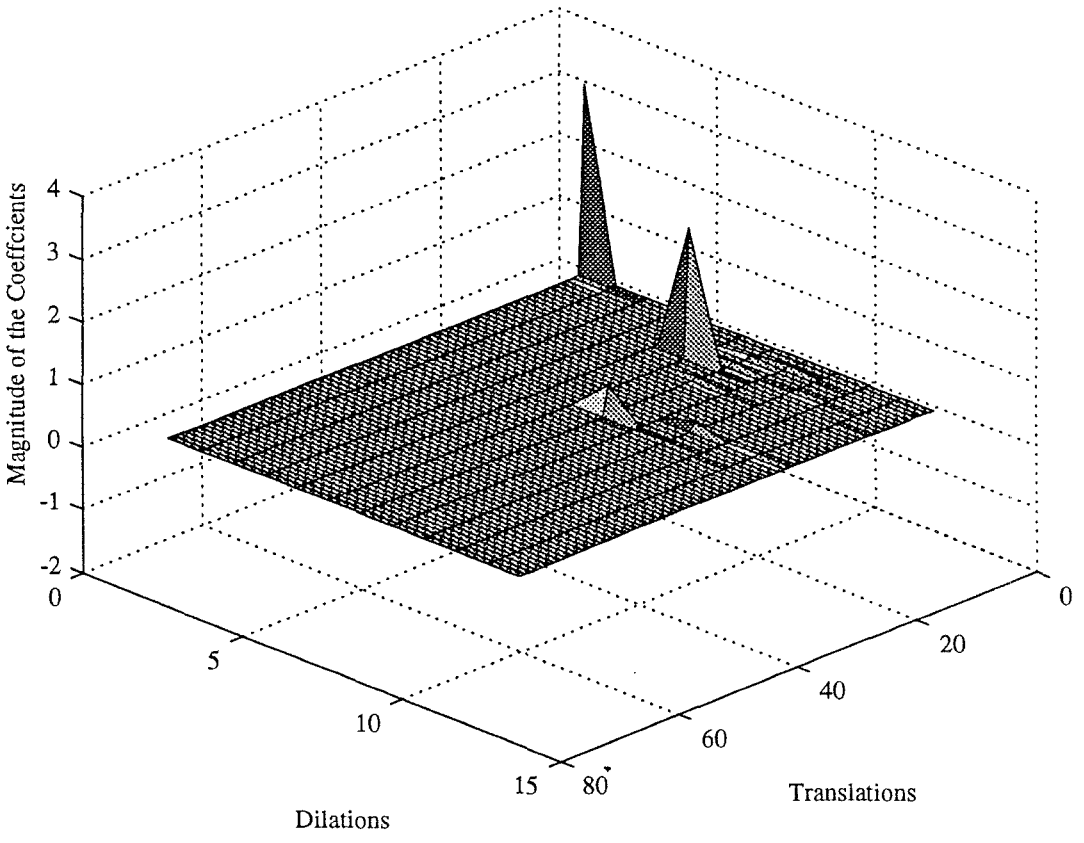

Figure 3.1: Magnitude of the wavelet coefficients (delay system)

The approximation results together with the true values of the frequency response are shown in figure (3.2). The model order is 16 here and the number of wavelets that has been used is only 5 (each wavelet with nonzero translation increase the order by four and those with zero translation only increase the model order by two).

For the purpose of comparison, the normalized error (in time domain) versus

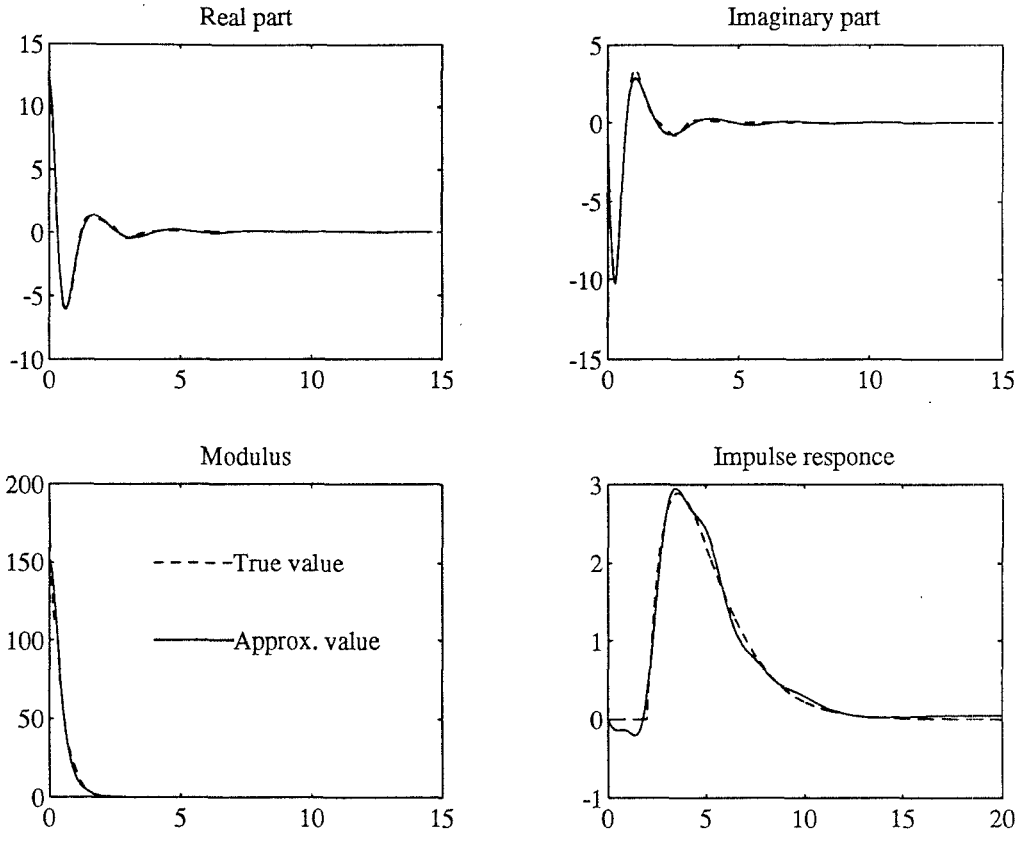

Figure 3.2: Approximation to the delay system with model order 16

the model order is plotted in figure (3.3). It can be seen from figure (3.3) that the matching pursuit algorithm outperforms both methods in a relatively wide range of model order and it is only with high order models that the least square method beats the matching pursuit. In this figure, the Laguerre data is taken from [15]

In figure (3.4) is again the magnitude of the wavelet coefficients for the cochlear filter (it is a filter in the human ear that separates different frequencies in the receiving wave). The same set of graphs are shown for in figure (3.5). As before, the time-frequency localization properties are clear from figure (3.4).

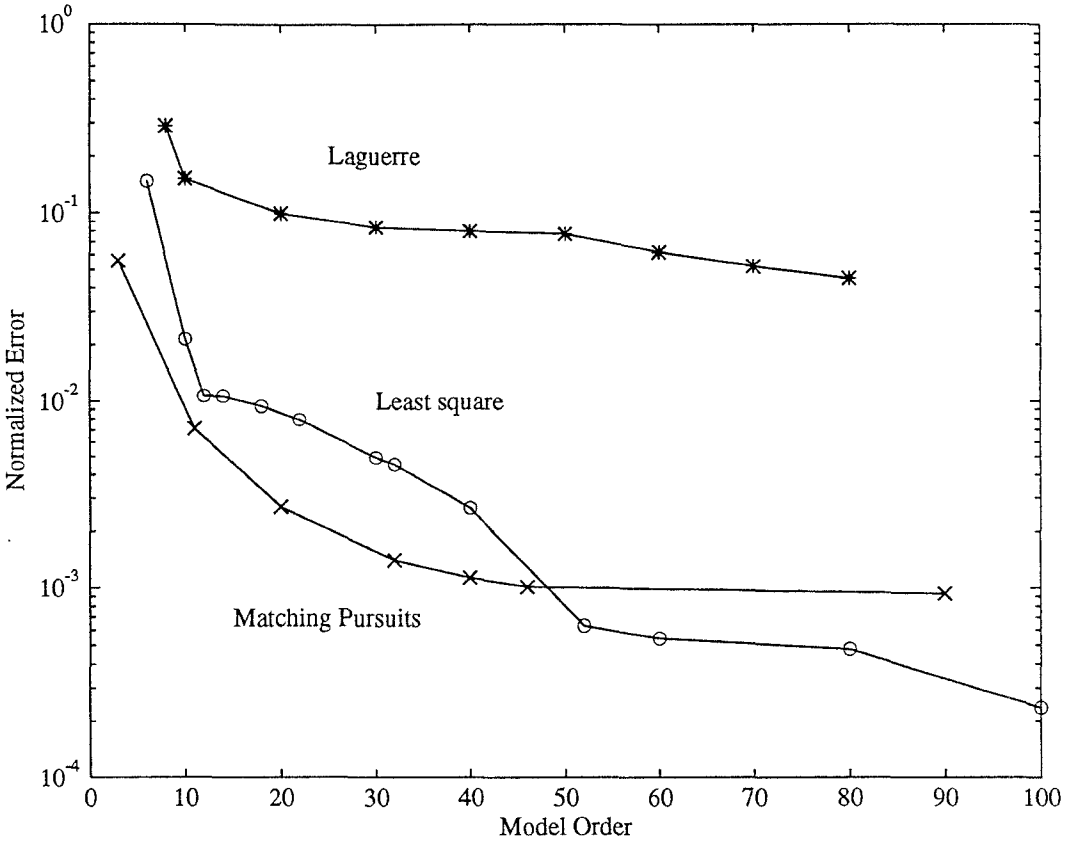

Figure 3.3: Normalized time domain approximation error versus model order for different algorithms (delay system) (Laguerre data from [14])

3.7 Application

As a practical application we consider the problem of approximation of the transfer function relating the input voltage to the actuating piezo ceramic to the output voltage of the sensors (which are piezo ceramics again). The setup for this experiment is shown in figure (6.7).

The empirical transfer function has been obtained through simulation and application of spectral analysis.

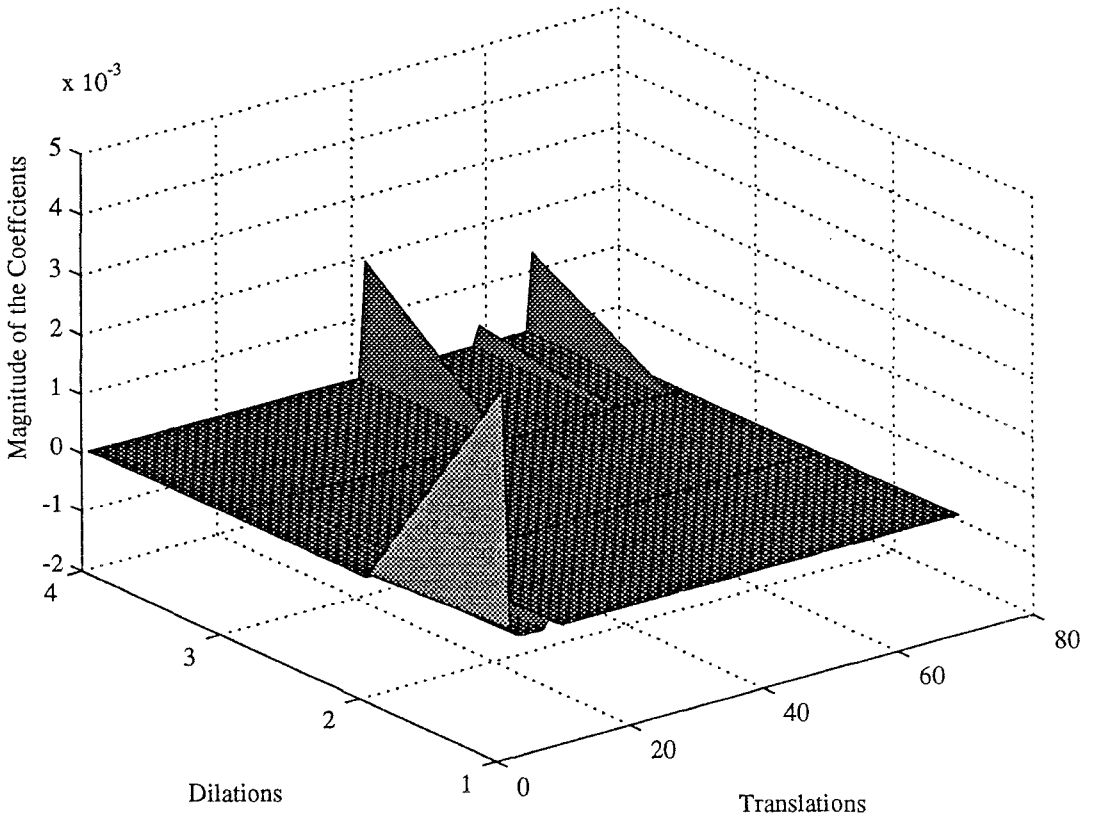

Figure 3.4: Magnitudes of wavelet coefficients (cochlear filter)

Due to the presence of resonances the frequency response turns out to be fairly well localized in the frequency domain, which is a desirable factor when we want to apply the matching pursuit algorithm.

Figure (3.7) shows the result of an approximation of degree 28. The dashed line represents the empirical frequency response, and the solid line is the the one obtained by the degree 28 approximation. Note that the governing model for the problem is actually a PDE and thus it is not a finite dimensional system. The analyzing wavelet is again the one given in Eq.(6.8.1).

It is also interesting to get a high order approximation and observe that

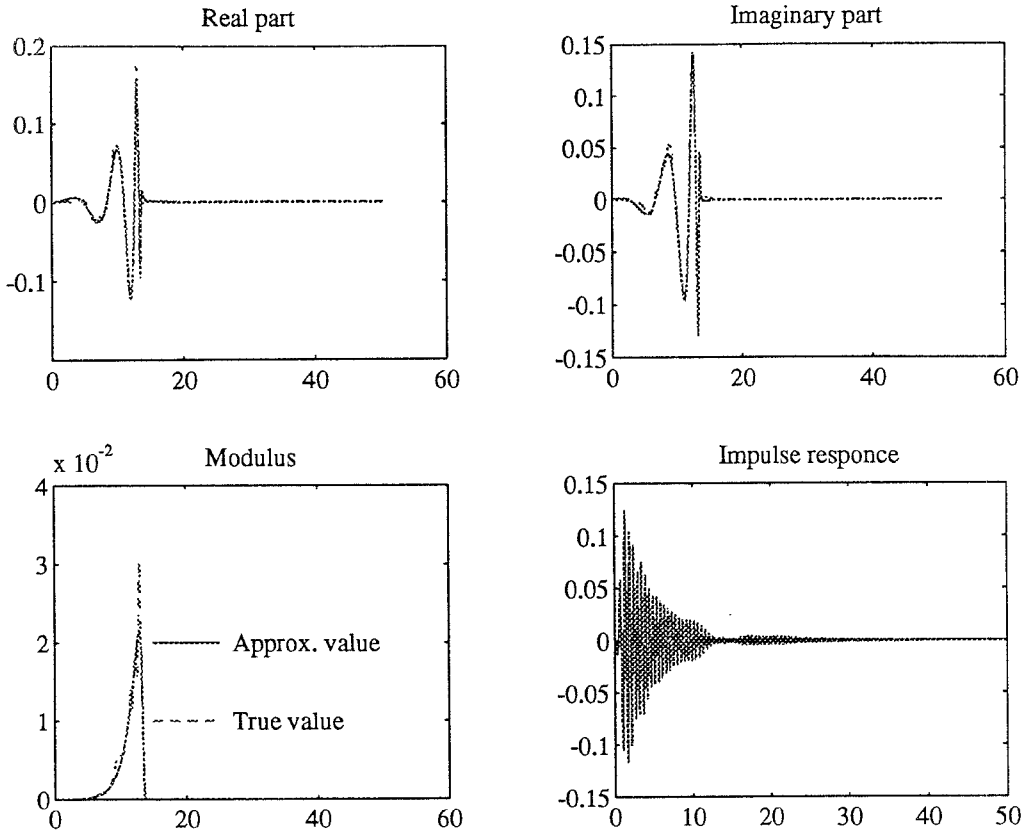

Figure 3.5: Approximation to the cochlear filter with model order 16.

only a few of the wavelet coefficients are of significant magnitude. Figure (3.8) reflects this fact.

3.8 The Experiment

In addition to previous applications we arranged an experimental setup consisting of a flexible cantilever Aluminum beam, piezo material, spectrum analyzer, and a power amplifier together with two power supplies. The objective here, as in its simulated version, is to find a finite dimensional model that approx-

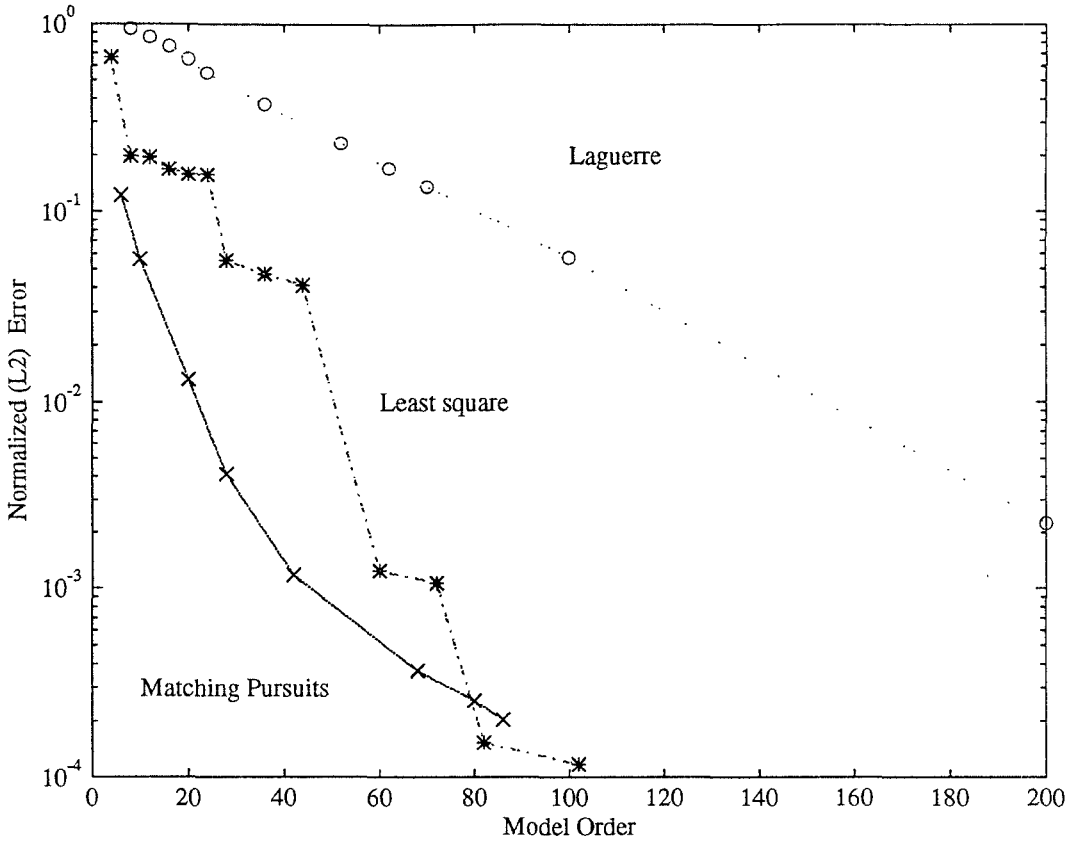

Figure 3.6: Normalized time domain approximation error versus time for different algorithms (cochlear filter) (Laguerre+Least square from [14]).

imates the transfer function from input voltage to the power amplifier to the output voltage measured by the piezo ceramics. A schematic diagram of the experimental setup is shown in fig.(3.9) .

The spectrum analyzer (HP 3566A) [6] applies a swept sine wave to the input of the power amplifier and senses the voltage generated by the sensor. The gain of the power amplifier is about 43 and it has a fairly large bandwidth [3]. The frequency response obtained in this way is actually the one from the input of the amplifier to the sensor voltage. The output voltage of the spectrum analyzer is

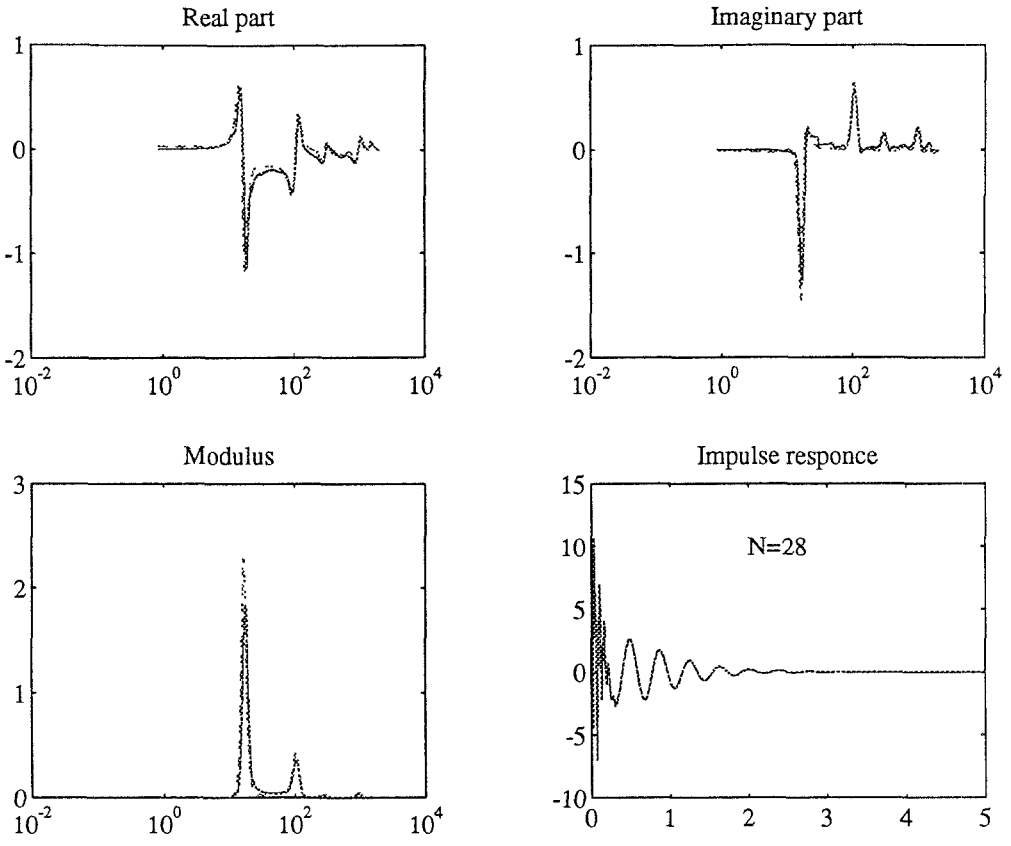

Figure 3.7: Approximation of order 28 to the beam transfer function

adjustable and has been set to get an output voltage of about 120 $volts\ pk-pk$ at the output of the amplifier. At frequencies below $80Hz$ the output is distorted and it is so small that the $60Hz$ noise of the line is more significant, so we start from $80Hz$. The Bode plot obtained from the experiment is shown in fig.(3.11). The peaks in the magnitude plot represent the resonance frequencies.

Having the empirical data at hand, the next step is to feed it to the program and get the approximation. Here there are a few parameters that we should choose before running the program. First, we must choose the translation step-size b_0. There is a limit on how large we can choose this parameter and it is

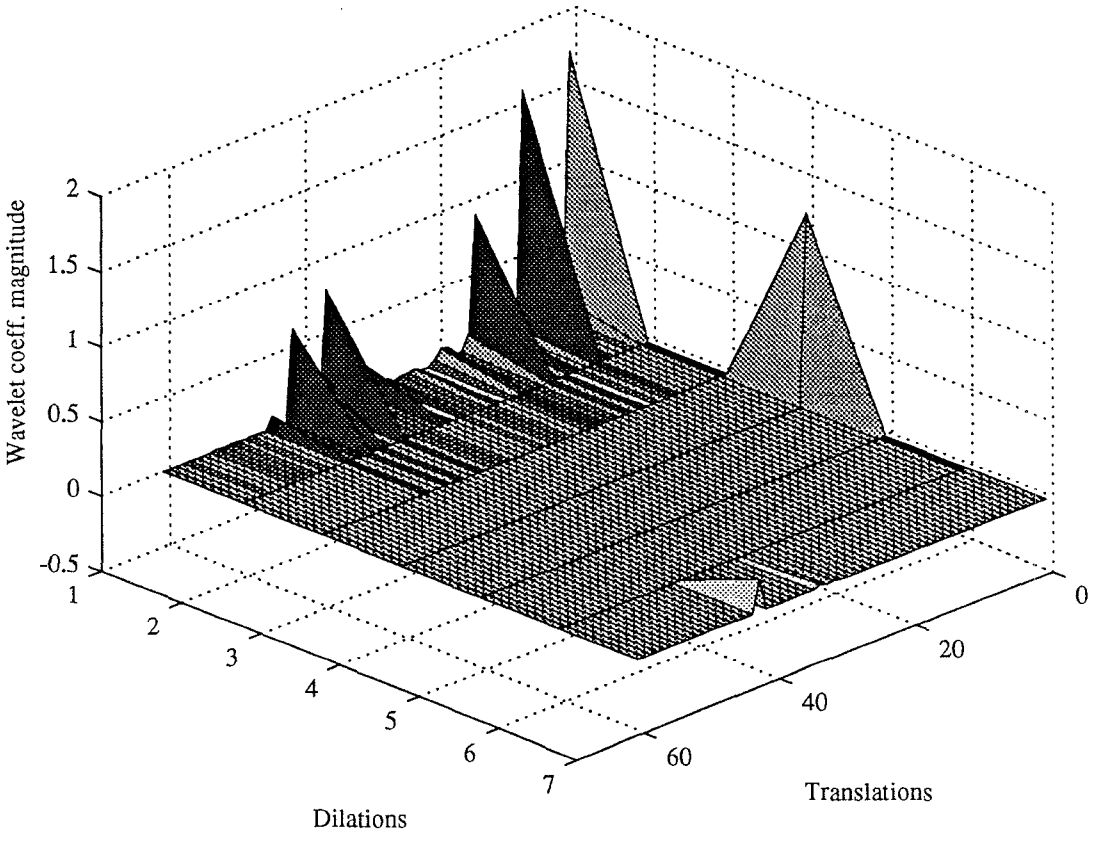

Figure 3.8: Magnitudes of wavelet coefficients (experimental set up)

determined by a limit beyond which the set of wavelets cease to be a frame (see [15] for a definition of frames). As the analyzing wavelet again we chose the second order system,

$$\Psi(s) = \frac{1}{(s+\gamma)^2+\xi},$$

with the parameters as in Table (3.1). The upper limit for b_0 to preserve the frame property is in this case is about 17 [15]. In Table (3.2) we have listed the selected elements of the dictionary for an approximation of order 46. Note that each wavelet with nonzero translation adds up four to the order (because we have to include the negative translation too) and those with zero translation

ξ	γ	a_0	b_0
1.0	5.0	2.0	16.37

Table 3.1: Parameters of the analyzing wavelet and translation-dilation step sizes

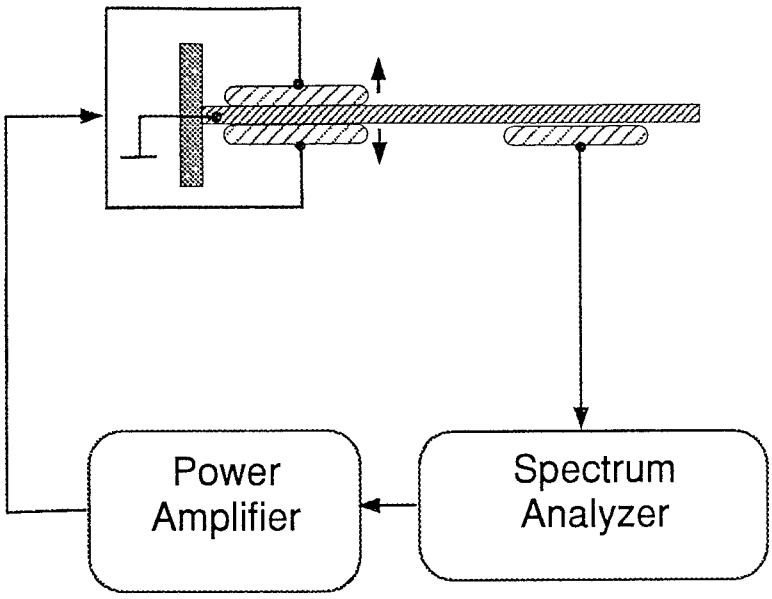

Figure 3.9: Schematic diagram of the experiment set up

just add two. The approximation has the following form:

$$\begin{aligned}\widehat{f}(s) = \; & \textstyle\sum_{n\neq 0}\sum_m (C_{mn}a_0^{m/2}g(a_0^{m/2}s - inb_0) + \overline{C_{mn}}a_0^{m/2}g(a_0^{m/2}s + inb_0)) + \\ & \textstyle\sum_m C_{m0}a_0^{m/2}g(a_0^{m/2}s).\end{aligned} \tag{3.8.1}$$

The frequency response of the approximation corresponding to Table(3.2), along with the empirical frequency response is plotted in Fig.(3.12). Fig.(3.14) shows the magnitude of the wavelet coefficients for a high order approximation.

3.9 Conclusion

The matching pursuit method is a simple yet efficient algorithm to compactly represent a waveform/signal. Here, with a little modification, we applied this method to the problem of obtaining an approximation to a given frequency response. Through comparison, it has been shown that this method out performs the classical Laguerre method for system identification and gives a lower normalized error for the same degree of the approximation. There is a natural

Dilation level	Translation level	C_{mn}
1	59	$-0.69032973051 - 3.84848356247i$
0	52	$3.12591791153 + 2.19522929192i$
0	9	$0.40908780694 + 3.03336858749i$
0	29	$2.59771537781 + 0.25139620900i$
-1	15	$-3.00720357895 - 7.58607721329i$
-1	7	$2.81444954872 - 0.27529984713i$
-1	20	$1.73732197285 - 2.21613240242i$
-2	13	$4.71194553375 + 23.73843002319i$
-3	5	$1.66028666496 - 9.23096656799i$
-4	3	$-3.34253716469 - 0.55811583996i$
-4	2	$0.47969451547 - 6.92282009125i$
-5	2	$5.80866384506 + 0.55891269445i$
-6	1	$-2.26374673843 - 2.98484420776i$

Table 3.2: Selected elements of the dictionary and their coefficients

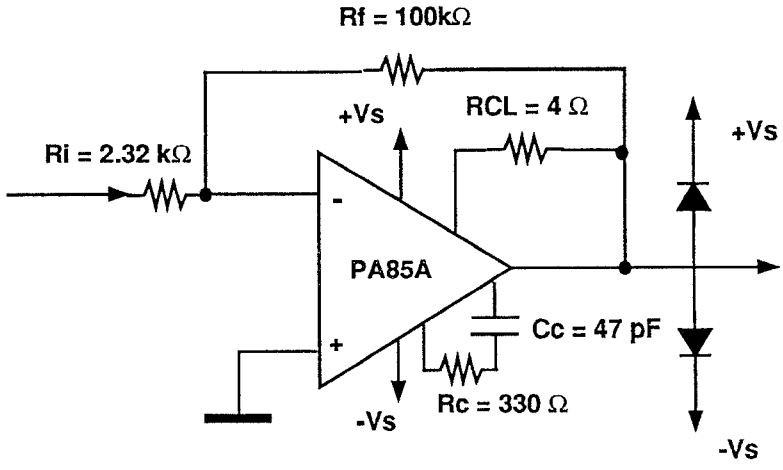

Figure 3.10: Piezo-electric transducer drive circuit

question here that one may ask about this method. i.e. since the approximation actually converges to the projection of the signal into the space spanned by the waveforms in the dictionary, why don't we simply compute the projection using pseudo inverse and in this way get the answer in one shot? There are two reasons that we do not want to do this; Firstly, computation of the pseudo inverse is very expensive and requires lots of operations specially when we are dealing with extremely redundant dictionary. The second reason, which is really more important, is that our objective is to pick those waveforms in the dictionary that are more important in the sense that they can represent the structure of the signal best. In this way we have taken advantage of the redundancy of the set of waveforms and at the same time if we want to only choose, say n number of waveforms to represent the signal we know when to stop where as in the case of pseudo inverse there is no clear way to do this.

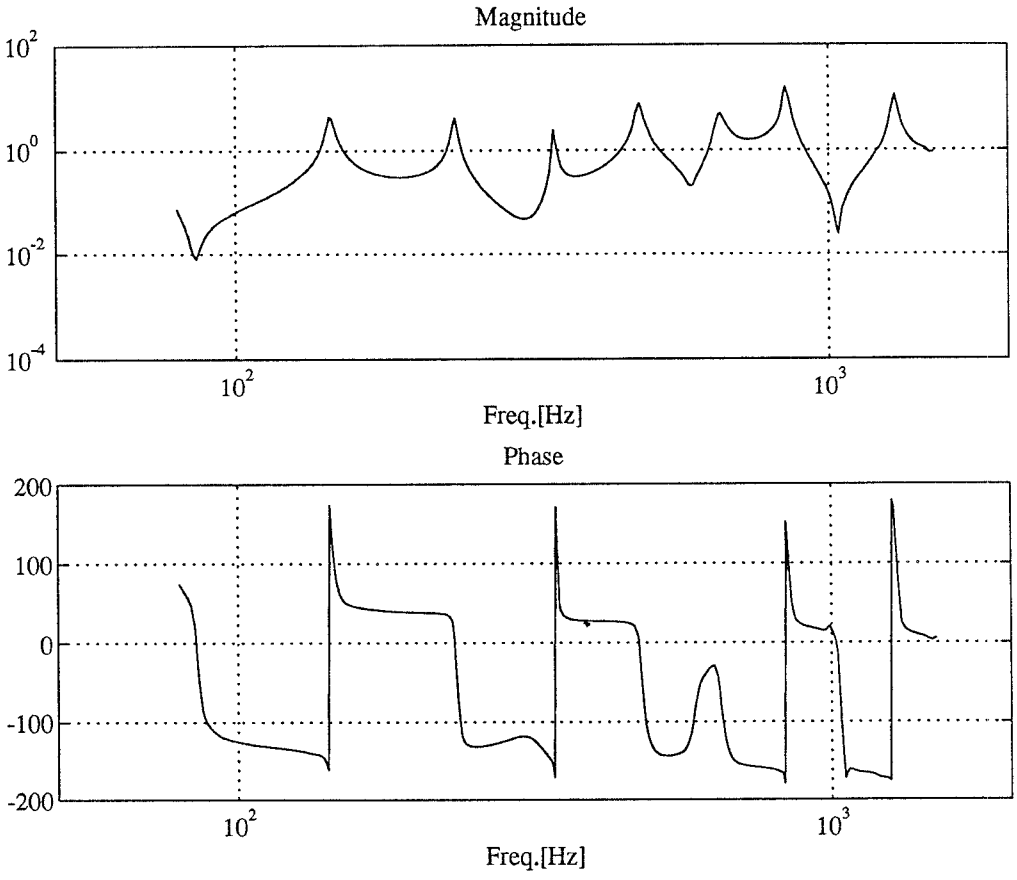

Figure 3.11: Frequency response obtained from the experiment

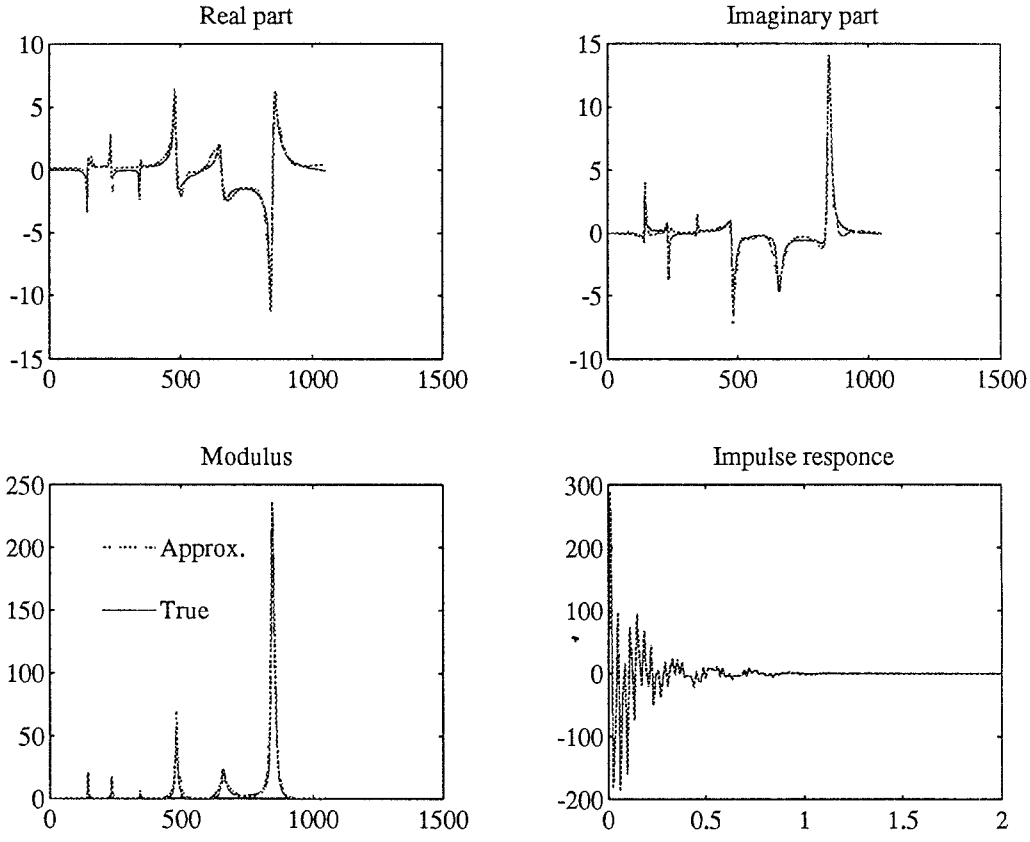

Figure 3.12: Approximation to the experimental freq. response of order 46

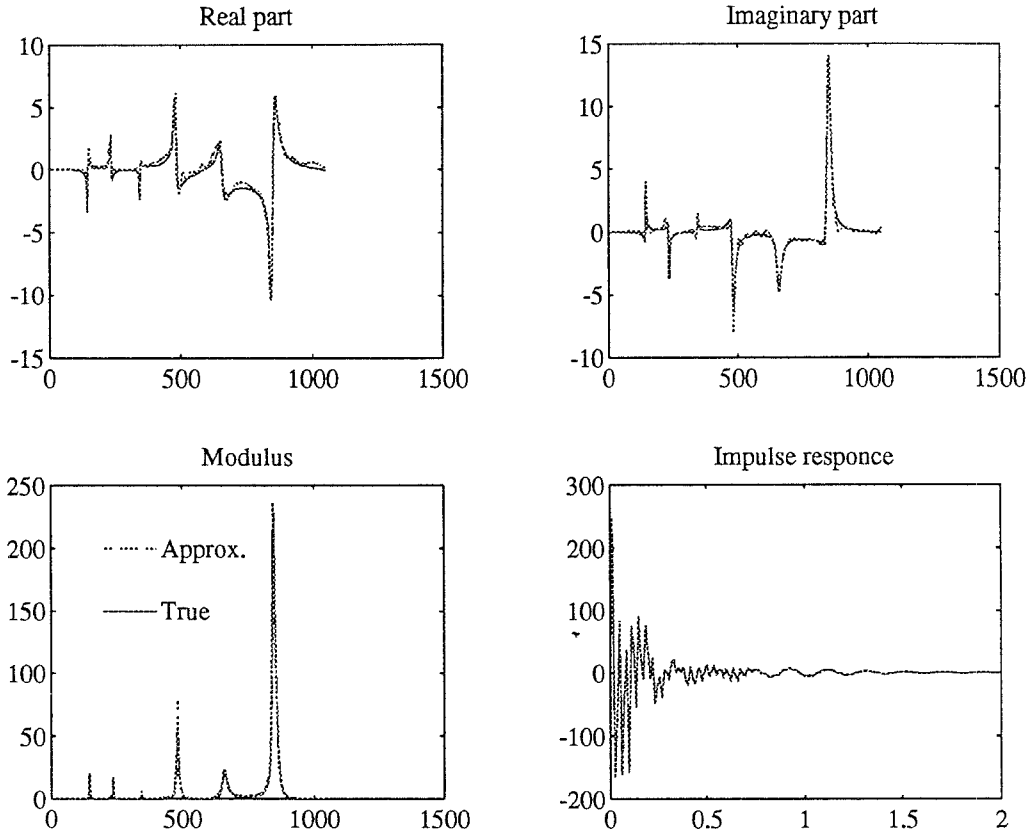

Figure 3.13: High order approximation to the experimental freq. response

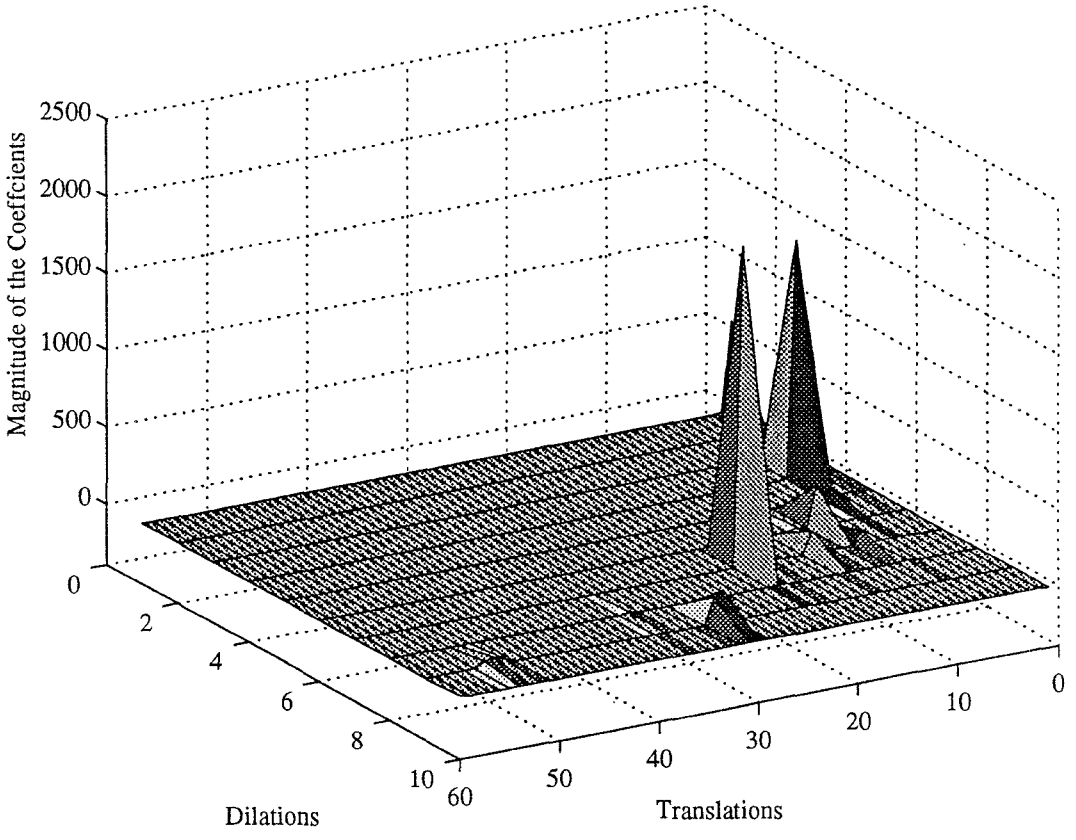

Figure 3.14: Magnitude of the wavelet coefficients corresponding to the experimental transfer function

Chapter 4

Design of Controller for the Constrained System

Like most of the systems in the real world, there is a limit on the amplitude of the input that can be safely applied to the actuators. In the case of flexible beam, these limits are imposed partly by the saturation limit on power amplifiers driving the piezoceramics and partly by the break down voltage of the piezo material itself. One approach to controller design would be to define a cost function and solve a constrained optimization problem to compute the optimum input. Although this gives the "optimal" solution, it has the disadvantage that both computation and storage of the optimum input is expensive. As an alternative procedure we consider the method proposed by Gutman and Hagander [13] and formulate the vibration suppression problem into a form suitable for application of this control strategy. The results of this method are then compared with the LQR method.

4.1 Application of the LQR method

As mentioned before, we assume that there is a constraint on the magnitude of the maximum and minimum voltages that can be applied to the input of the power amplifiers driving the piezoceramic actuators. As a basis for comparison first we use the LQR method for obtaining the suitable stabilizing feedback gain L. The penalty on inputs has been increased until the magnitude of inputs satisfy the constraints at all times. The results of the simulation are shown on figure 4.1. Here, we used the linearized Galerkin model obtained in chapter 2.

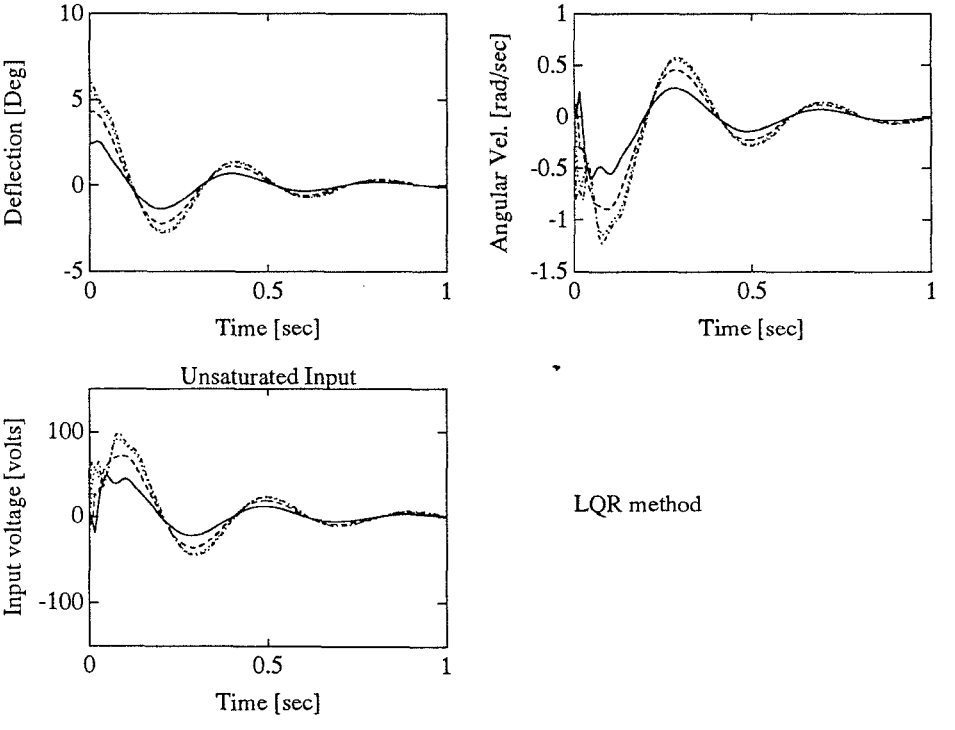

Figure 4.1: Controlled vibration of the beam (Linear Quadratic Regulator)

Shown in figure 4.2 and figure 4.3 are the results of the simulation for the cases that a disturbance torque in the form of a sinusoid at the frequency of the largest and the smallest modes is applied at the first joint, respectively.

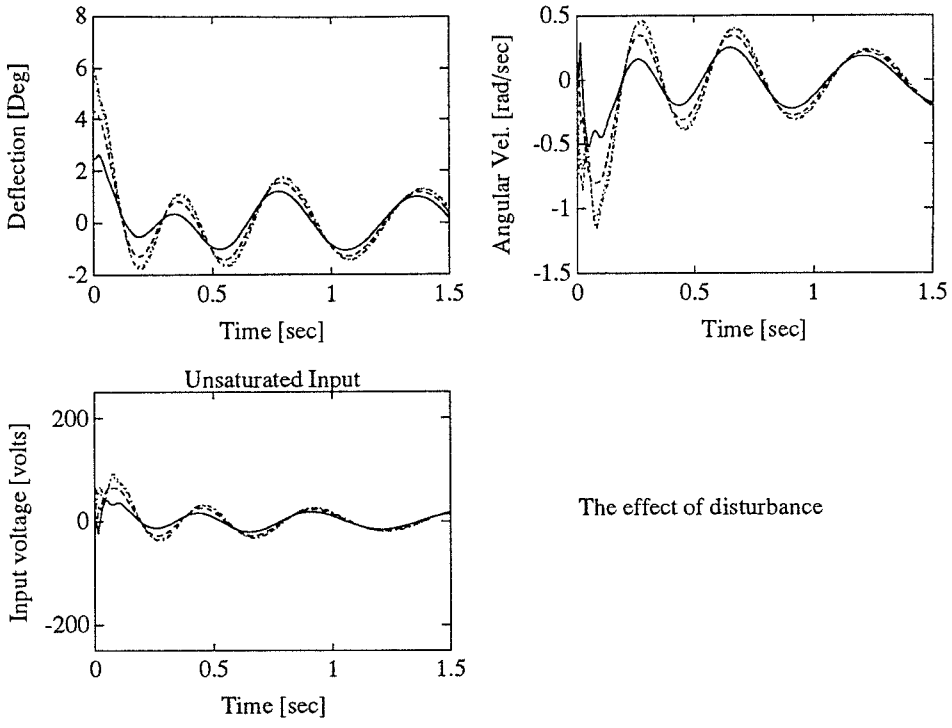

Figure 4.2: Controlled vibration of the beam under the disturbance torque (Linear Quadratic Regulator)

Now let us put a saturation block at the input terminal of the system and observe the effects of saturation with the LQR as the controller as we increase the amount of the initial deflection of the free end of the beam. Simulation results are shown in figure 4.4.

4.2 The Gutman-Hagander Method

In this section, we use a control strategy devised by Gutman and Hagander [13] for controlling systems subject to input constraints. Although this method is not optimal, it guarantees the asymptotic stability. Hereafter, in this chapter we will use the same notations and symbols as in [13] and the reader is referred

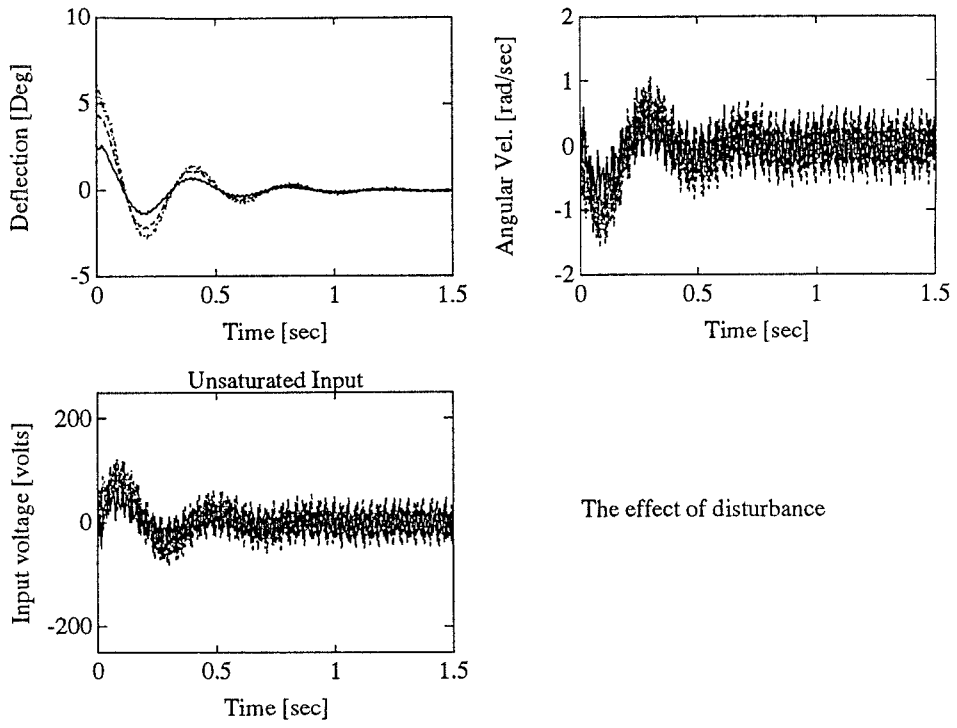

Figure 4.3: Controlled vibration of the beam under the disturbance torque (Linear Quadratic Regulator)

to that paper for a detailed description of the method and definition of various symbols that we are going to use. The general idea is to first stabilize the system by a low-gain linear state feedback. Then, quadratic Lyapunov function is found, on the basis of which another linear state feedback is computed. The two controls are added and passed through the saturation element Figure (4.5) shows the block diagram of such a controller.

The algorithm consists of five steps:

- *Step 1:* Determine the set of initial conditions, D. This is the set from which we want the controller steer the state of the system to zero. One obtains an estimate for this set by a knowledge of the physics of the system.

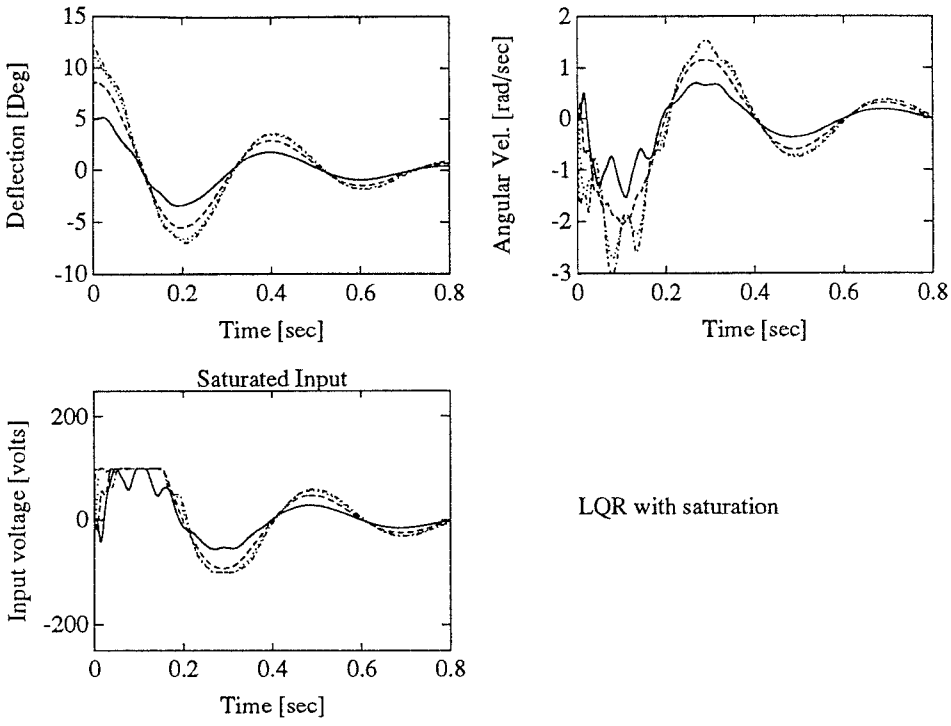

Figure 4.4: The effect of saturation on the LQR controller

- *Step 2:* Find the low-gain stabilizing feedback gain by solving an LQR problem. Increase the control penalty until the inputs satisfy the constraint for x starting in D.

- *Step 3:* Find the positive definite matrix P by solving the following Lyapunov equation,

$$PA_c + A_c^T P = -Q, \tag{4.2.1}$$

where, Q is a positive definite matrix that can be considered as a control parameter and $A_c \equiv A + BL^T$. Since A_c is stable this equation has a unique positive definite solution.

- *Step 4:* Let E be the set of state variables such that the control $u = L^T x$

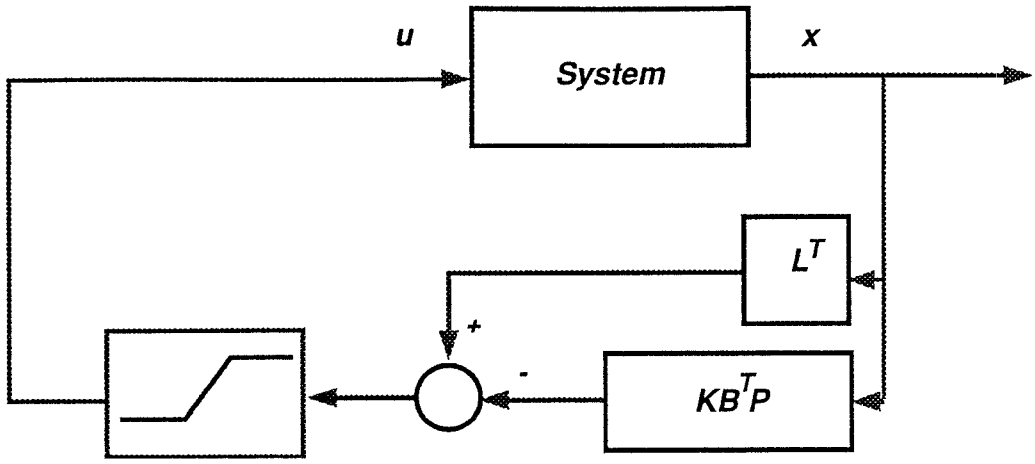

Figure 4.5: Block diagram of a Gutman-Hagander type controller.

satisfies the constraint. Check to see if the following inequality holds,

$$\sup_{x \in D} x^T P x \leq \min_{x \in \delta E} x^T P x. \tag{4.2.2}$$

If not either go back to step 3 and choose another matrix Q, or change the feedback gain L, or choose a more restrictive set of initial conditions D.

- *Step 5:* Set up the control u and tune the parameter K, which is a diagonal positive definite matrix,

$$u = sat[(L^T - KB^T P)x]. \tag{4.2.3}$$

In the following subsections, we formulate the problem of vibration damping such that fits the Gutman-Hagander (G-H) algorithm.

Identifying the Initial Condition Set D

Assume that the objective is to damp the vibration of the beam resulting from deflecting the tip of the beam and then releasing it. Simulation results of free oscillation can be used to obtain the region from which we want to steer the

state to zero. i.e. by looking at the maximum and minimum values that each state variable takes during free oscillation, we can get an estimate of the range of values on which state variables take values. This, of course, gives a conservative and rather large estimate for the set D because the extremum of all state variables do not occur at the same instant of time. Thus, we get inequalities like the following,

$$m_i \leq x_i \leq M_i \qquad i = 1, \cdots, N. \tag{4.2.4}$$

Constraints on Inputs

As mentioned before, there is a limit on the voltage that can be applied to the piezo material as an actuator. This limit is due to two factors: first, the breakdown voltage of the piezoceramic and second, the limit imposed by the saturation voltage of the power amplifier. So, there are constraints of the following form on inputs,

$$V_{min} \leq V_i \leq V_{max} \qquad i = 1, \cdots, N, \tag{4.2.5}$$

where, V_i is the input voltage to the power amplifier.

The Stabilizing Feedback Gain 'L'

As mentioned in Step 2, the objective is to find the feedback gain matrix $L = [l_1 \mid l_2 \mid \cdots \mid l_m]$ such that the input $V = L^T x$ stabilizes the system (asymptotically) while the inputs V_i satisfy inequalities (4.2.5) for all $x \in D$ (this guarantees that $D \subset E$ (see [13])). Given the feedback gain L the maximum of $l_i^T x$ is achieved when x is computed from the following rule,

$$x_{j\ max} = \begin{cases} M_j & l_{ji} \geq 0 \\ m_j & l_{ji} < 0 \end{cases}$$
$$j = 1, \cdots, 2N. \tag{4.2.6}$$

Similarly, the minimum of $l_i^T x$ is achieved for

$$x_{j\ min} = \begin{cases} m_j & l_{ji} \geq 0 \\ M_j & l_{ji} < 0 \end{cases}$$
$$j = 1, \cdots, 2N. \tag{4.2.7}$$

One way to choose the feedback gain L is to use the LQR method and increase the penalty on input until inequalities (4.2.5) are satisfied with V_i replaced by $l_i^T x$, for all $x \in D$. To check if the given feedback gain is acceptable, the above rules for $x_{j\ min}$ and $x_{j\ max}$ may be used.

Feasibility Check

In this step we check to see if the following inequality holds for a given choice of P, L, and D [13]:

$$\sup_{x \in D} x^T P x \leq \min_{x \in \delta E} x^T P x. \tag{4.2.8}$$

At the left hand side of the above inequality, the supremum can be achieved only at the vertices of the polytope (a convex hull of finitely many points) D. The reason for this is that P can be diagonalized with an orthogonal transformation T. Then in the new coordinate system, the left hand side of (4.2.8) can be rewritten as, $\sup_{z \in T(D)} \sum_i z_i^2 \lambda_i$ ($\lambda_i > 0$). The result follows by noting that first, the transformation induced by T is actually a rotation and vertices

are transformed to vertices and second, the maximum of the above expression occurs at a vertex of the transformed poly tope. Therefore, $\sup_{x\in D} x^T P x$ can be computed after 2^{2N} checks (i.e. evaluating $x^T P x$ on the vertices of the region D and taking the maximum value).
Also, the left hand side of the inequality (4.2.8) is equal to [13],

$$\min_{x\in\delta E} x^T P x \;=\; \min_i[\min\{\frac{V_{min}^2}{l_i^T P^{-1} l_i},\; \frac{V_{Max}^2}{l_i^T P^{-1} l_i}\}]. \tag{4.2.9}$$

Again, if the inequality (4.2.8) is not satisfied we have two options: first, change Q to get another P and second, increase the input penalty to obtain another feedback gain L and proceed as before.

Gutman and Hagander showed that [13] the input,

$$V \;=\; sat[(L^T - kB^T P)x], \tag{4.2.10}$$

stabilizes the system for all positive definite diagonal matrices k. In order to choose the tuning matrix k we plot the eigenvalues of the closed loop system (without the saturation) as k changes (here we choose $k = cI$ and we vary c to obtain the root locus). The tuning parameter k has been chosen so that the dominant eigenvalue (the one that is the closest to the imaginary axis) be as far as possible from the imaginary axis. Figure 4.6 shows a portion of the root locus.

The results of simulation for this particular choice of tuning parameters are shown in figure 4.7.

In order to compare the performance of the LQR method and the Gutman-Hagander method, a disturbance in the form of a sinusoid is applied at the first

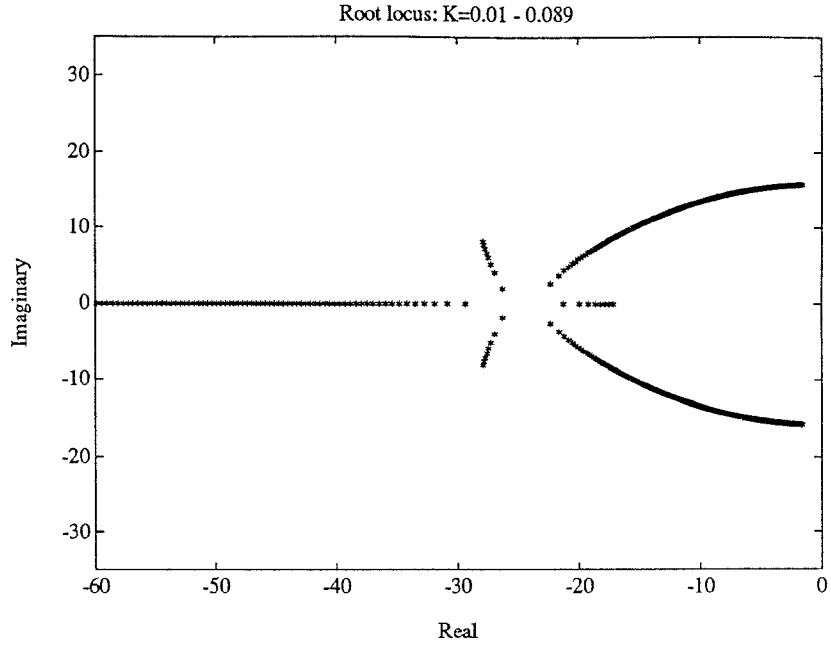

Figure 4.6: Location of the eigenvalues of the closed loop system as the tuning parameter k varies

joint. In figure 4.8 the frequency of this sinusoid is chosen to be the lowest natural mode of the beam. Figure 4.9 shows the results when the frequency of the disturbance is the highest natural mode of the beam.

Up to this point we studied the behavior of the linearized system. To observe the effect of non-linearities(which come into the picture when we use the non-linear equations of motion, (2.1.16), instead of the linearized ones). In order to perform the simulation, the non-linear equations of motion has been transformed in the following form :

$$Q\ddot{\alpha} = f(\alpha, \dot{\alpha}, M), \tag{4.2.11}$$

where Q is a symmetric matrix

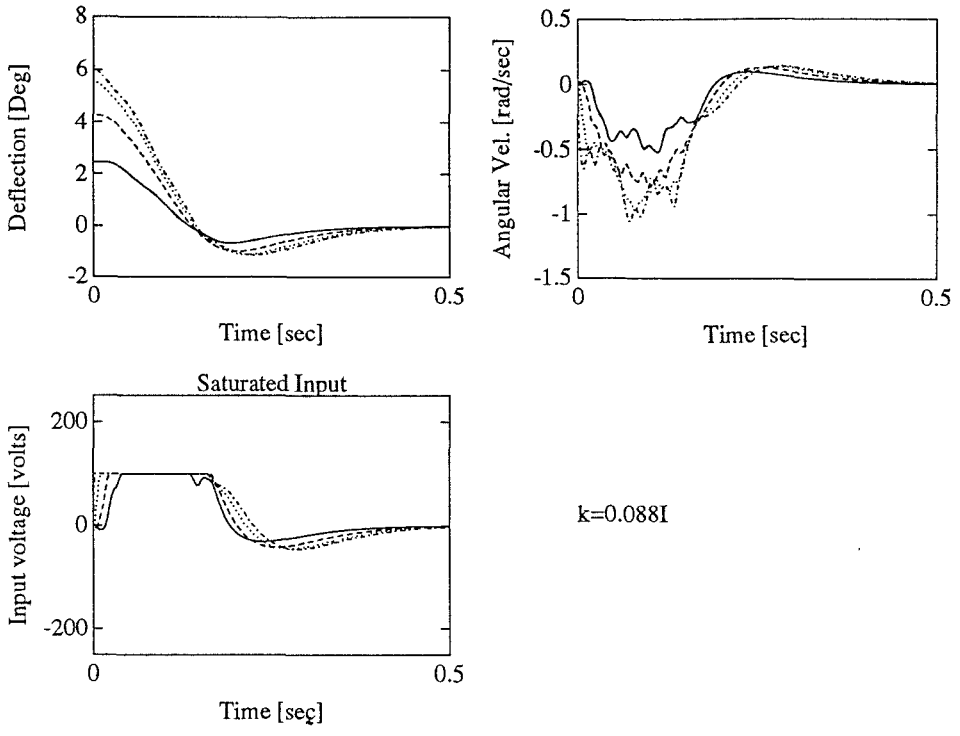

Figure 4.7: Controlled vibration of the beam (Gutman-Hagander method)

Two simulations has been performed. In the first simulation we want to show the effect of non-linearities in the free oscillation of a four link model. One may compare figure 2.5 and figure 4.10 .

In the second simulation the Gutman-Hagander control is applied to the linear and nonlinear systems. Although in this simulation we consider large deflections of the beam (which is unrealistic), again comparison of figure 4.11 and 4.12 shows that the responses are very similar and the controller work well even for the nonlinear system.

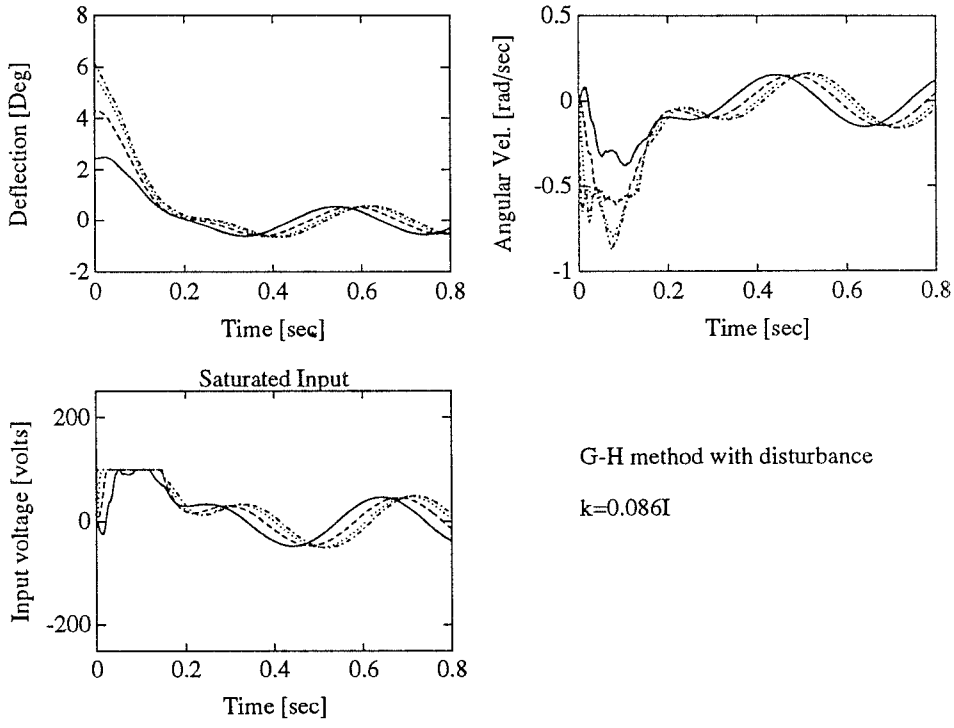

Figure 4.8: Performance of Gutman-Hagander controller under external disturbance (low frequency disturbance)

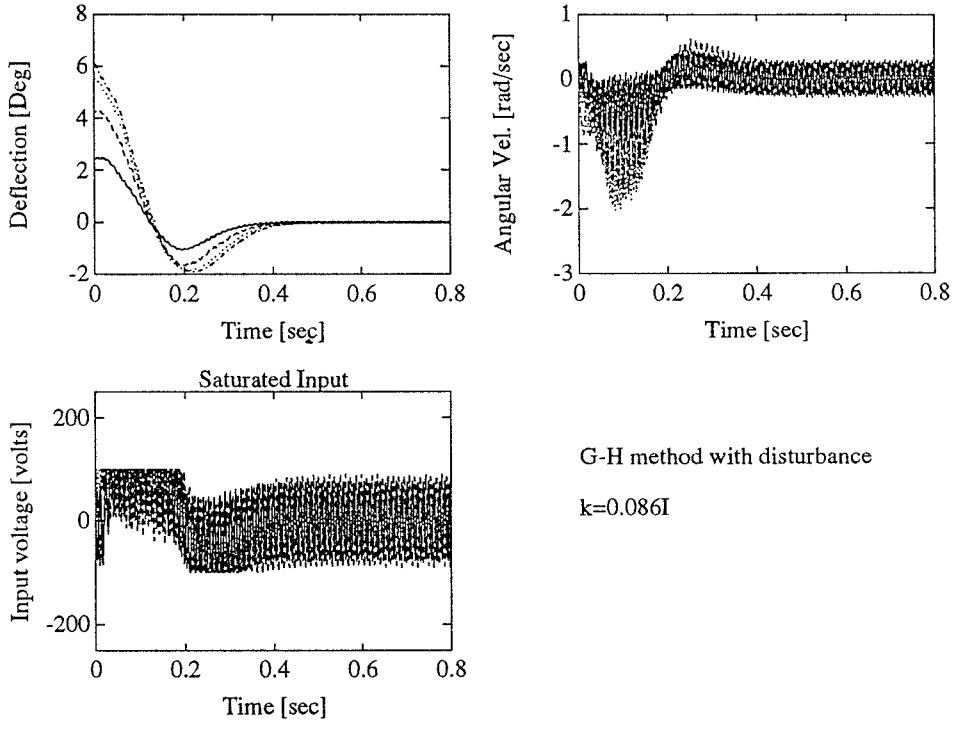

Figure 4.9: Performance of Gutman-Hagander controller under external disturbance (high frequency disturbance)

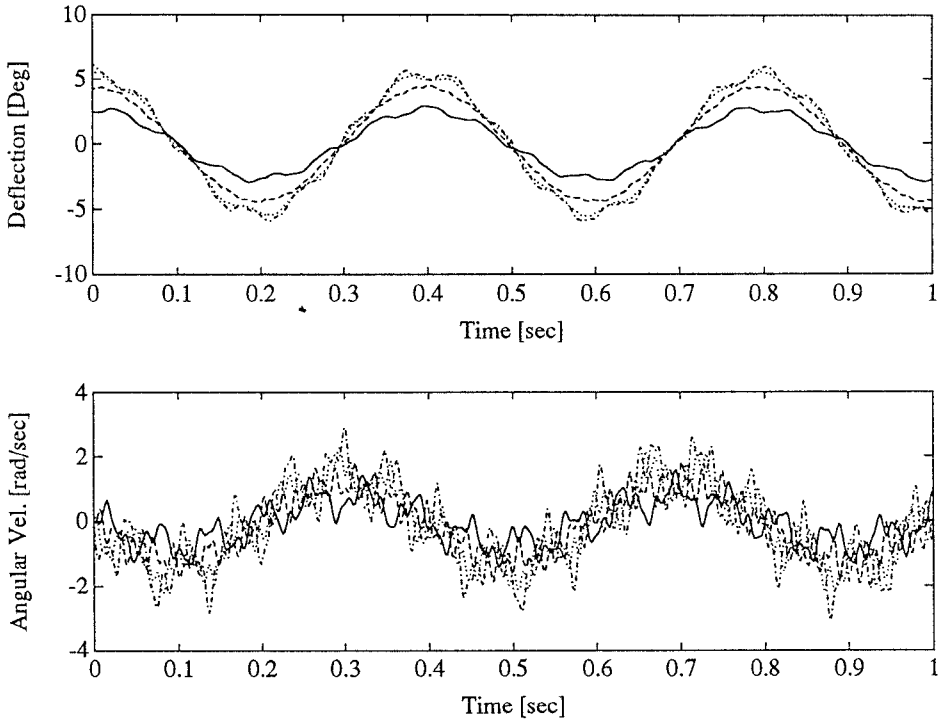

Figure 4.10: Free Oscillation (Nonlinear Model)

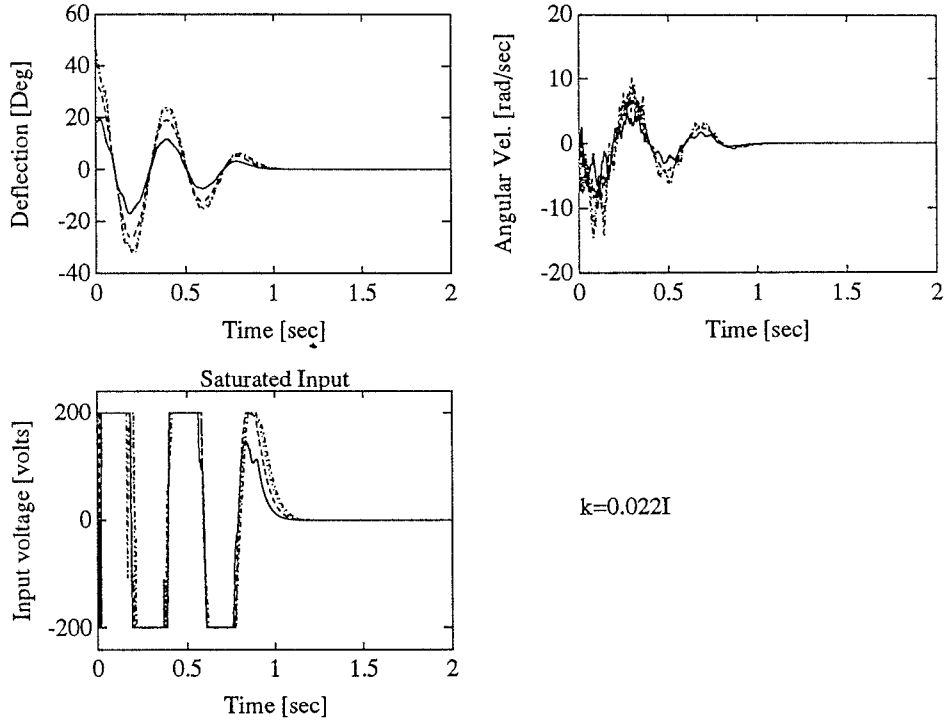

Figure 4.11: Controlled vibration of the beam (Large deflection, Linear Model, G-H method)

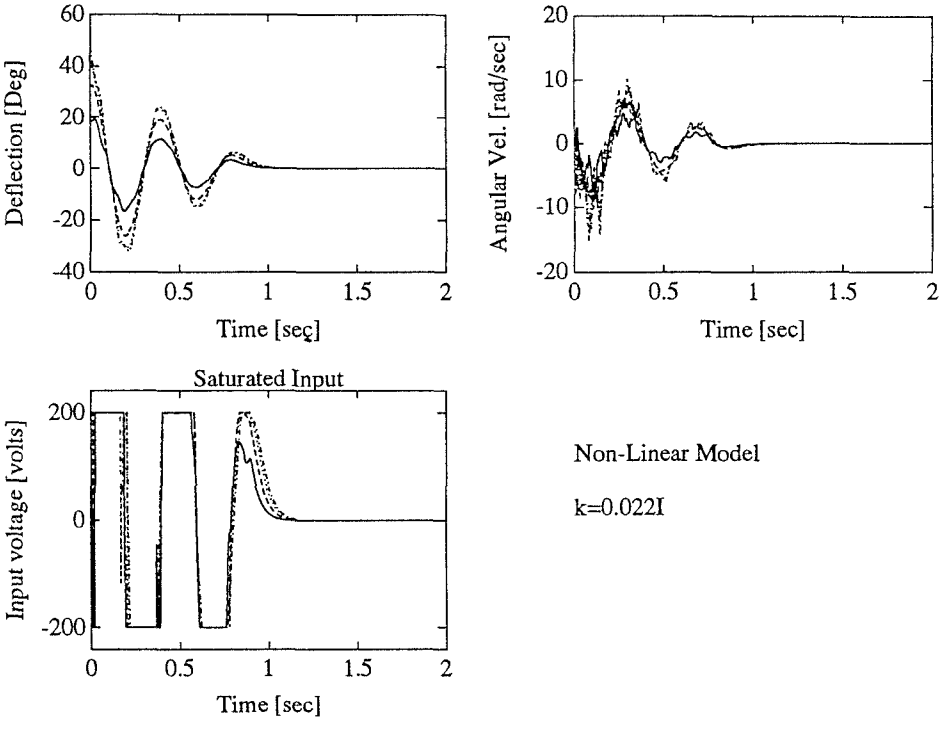

Figure 4.12: Controlled vibration of the beam (Large deflection, Nonlinear Model, G-H method)

Chapter 5

Observer Based Gutman-Hagander Type Controller

In this Chapter we prove that under certain conditions, a Gutman-Hagander type controller with a classical state observer can be used for LTI systems with constraints on the magnitude of input to make them stable.

5.1 Introduction

The problem of stabilizing LTI (linear time invariant) systems with constraints on the magnitude of the input has been considered by Gutman-Hagander [13]. The method requires the knowledge of all the state variables. In many practical situations it is not possible or it is too expensive to measure all the state variables. So, one may try to use the estimated values of the state variables and then exploit the Gutman-Hagander (G-H) type controller. We prove that it is possible provided that some conditions are satisfied. In Section 5.2 we obtain a proper Lyapunov function to prove the stability of the method. In Section 5.3 we propose an algorithm (which is very similar to the one proposed by Gutman and Hagander) to choose the control parameters. Next, in order to

test the performance of the control strategy we apply this control law to the problem of damping the vibration of a cantilever beam.

5.2 Control Strategy

Consider a controllable and observable linear time invariant plant represented as:

$$\begin{cases} \dot{x} &= Ax + Bu, \\ y &= Cx. \end{cases} \tag{5.2.1}$$

The control strategy due to Gutman and Hagander proposed the following control law as stabilizing input:

$$u = sat[(L^T - KB^T P)x], \tag{5.2.2}$$

where K is a positive definite diagonal matrix. In what follows we prove that the above control law still makes the system stable, even if one uses the estimated state variable, $\hat{x}$, computed by an state observer instead of the actual sate, x. i.e.

$$u = sat[(L^T - KB^T P)\hat{x}]. \tag{5.2.3}$$

Let e be the estimation error (i.e. $e \equiv x - \hat{x}$). It is well known that the dynamics of the error is described as:

$$\dot{e} = (A - HC)e, \tag{5.2.4}$$

where, H is the observer gain.

Let P , G be the positive definite solutions to the algebraic Lyapunov equations :

$$PA_c + A_c^T P = -Q, \tag{5.2.5}$$

$$GA_{ob} + A_{ob}^T G = -M. \tag{5.2.6}$$

Where $A_{ob} \equiv A - DC$, $A_c \equiv A + L^T B$, and M and Q are symmetric positive definite matrices M (these positive definite solutions exist because A_c and A_{ob} are stable). Suppose R is a positive definite diagonal matrix and apply the following input to the plant 5.2.1:

$$u = (L^T - RB^T P)\hat{x} \tag{5.2.7}$$

The claim is that $V(x,t)$ defined below is a Lyapunov function for the closed loop system (including the observer).

$$V = x^T P x + \alpha e^T G e \tag{5.2.8}$$

for some positive constant α which is to be chosen to make the derivative of V negative. One can show [13] that ,

$$sat[(L^T - KB^T P)\hat{x}] = (L^T - RB^T P)x, \tag{5.2.9}$$

for $0 < R < K$, provided that the condition $D \subseteq E$ holds (see section 4.2 for definitions of D and E). Taking the derivative with respect to time of eq. 5.2.8 and substituting for $u(t)$ from eq. 5.2.7 we get:

$$\begin{aligned}
\dot{V} &= (Ax+Bu)^T Px + x^T P(Ax+Bu) + \alpha e^T A_{ob}^T Ge + \alpha e^T G A_{ob} e \\
&= x^T(PA_c + A_c^T P)x - x^T PRB^T Px - x^T PBL^T e \\
&\quad + x^T PBRB^T Pe - e^T LB^T Px - x^T PBRPx \\
&\quad + e^T PBRB^T Px + \alpha e^T A_{ob}^T Ge + \alpha e^T G A_{ob} e \\
&= -x^T Qx - 2x^T (B^T P)^T R(B^T P)x \\
&\quad - 2e^T(LB^T P - PBRB^T P)x - \alpha e^T Me.
\end{aligned} \tag{5.2.10}$$

To simplify the above equation define S and U as,

$$\begin{aligned}
S &\equiv Q + 2(B^T P)^T R(B^T P), \qquad S > 0, \\
U &\equiv 2LB^T P - 2PBRB^T P.
\end{aligned} \tag{5.2.11}$$

Rewrite 5.2.10 in terms of S and U to get:

$$\dot{V} = -x^T Sx - e^T Ux - \alpha e^T Me. \tag{5.2.12}$$

The idea is to choose α large enough such that $\dot{V}$ becomes negative. Since matrices M and S are symmetric and positive definite these inequalities hold:

$$\lambda_M^{min}\|e\|^2 \leq e^T Me \leq \lambda_M^{max}\|e\|^2, \tag{5.2.13}$$

$$\lambda_S^{min}\|e\|^2 \leq e^T Se \leq \lambda_S^{max}\|e\|^2, \tag{5.2.14}$$

$$e^T Ux \leq |e^T Ux| \leq \|U\|\|x\|\|e\|. \tag{5.2.15}$$

Thus,

$$\begin{aligned}\dot{V} &\leq -\lambda_S^{min}\|x\|^2 - \alpha\lambda_M^{min}\|e\|^2 + \|U\|\|x\|\|e\| \\ &= -2\alpha_S^2\|x\|^2 - (\tfrac{\|U\|}{2\alpha_S})^2\|e\|^2 + \|U\|\|e\|\|x\| + (\tfrac{\|U\|}{2\alpha_S})^2\|e\|^2 - \alpha\lambda_M^{min}\|e\|^2,\end{aligned}$$

where, $2\alpha_S^2 \equiv \lambda_S^{min}$.

Subsequently,

$$\begin{aligned}\dot{V} &\leq -\alpha_S^2\|x\|^2 - (\alpha_S\|x\| - \tfrac{\|U\|}{2\alpha_S}\|e\|)^2 - (\alpha - \tfrac{\|U\|^2}{4\alpha_S^2})\|e\|^2 \\ &\leq -\alpha_S^2\|x\|^2 - (\alpha - \tfrac{\|U\|^2}{4\alpha_S^2})\|e\|^2.\end{aligned} \tag{5.2.16}$$

To make sure that $\dot{V}$ remains negative choose α such that,

$$\begin{aligned}\alpha &\geq \tfrac{\|U\|^2}{4\alpha_S^2} \\ &= \tfrac{\|U\|^2}{2\lambda_S}.\end{aligned} \tag{5.2.17}$$

Then we have,

$$\dot{V} \leq -\alpha_S^2\|x\|^2. \tag{5.2.18}$$

5.3 Stability Under Variable Matrix R

In the previous section we proved the stability of the plant when the control 5.2.7 is used. To relate this control to the control with saturation given by Equation (5.2.3) (see the inequality (5.2.9)), we need the following proposition.

Proposition: The input 5.2.7 with the condition $0 \leq R \leq K$ makes the plant 5.2.1 asymptotically stable.

Proof:

Now, both λ_S^{min} and $\|U\|$ in 5.2.17 are variables. So, we try to obtain an upper bound for $\|U\|$ and a lower bound for λ_S^{min}. From 5.2.11 we can write:

$$\begin{aligned} \|U\| &= \|2LB^TP - 2PBRB^TP\| \\ &\leq \|2LB^TP\| + 2\|PB\|^2\|R\| \\ &\leq \|2LB^TP\| + 2\|PB\|^2\|K\|, \end{aligned} \tag{5.3.1}$$

the last inequality follows from the assumption that $R \leq K$.

The next step would be to find an upper bound for λ_S^{min}:

$$\begin{aligned} \lambda_S^{min} &= \inf_{\|x\|=1} x^TSx \\ &= \inf_{\|x\|=1}(x^TQx + x^TNx). \end{aligned} \tag{5.3.2}$$

where $N \equiv 2(B^TP)^TR(B^TP)$. We also have:

$$\begin{aligned} \lambda_Q^{min} &\leq x^TQx \\ &\leq x^TQx + x^TNx. \end{aligned} \tag{5.3.3}$$

The above inequality shows that λ_Q^{min} is a lower bound for the set defined by $\left\{x^TQx + x^TNx \mid \|x\| = 1\right\}$ we get:

$$\lambda_S^{min} \geq \lambda_Q^{min}. \tag{5.3.4}$$

Therefore, the inequality 5.2.17 is satisfied if the following holds:

$$\alpha \geq \frac{2\|LB^TP\| + 2\|PB\|^2\|K\|}{\lambda_Q^{min}}. \tag{5.3.5}$$

This implies the negativity of $\dot{V}$ and asymptotic stability of the plant represented by 5.2.1 with the control given by Equation 5.2.3.

5.4 Controller Design Procedure

The controller design algorithm is essentially the same as the one proposed by Gutman-Hagander. The only difference is that, now we have to redefine the sets D , E , etc. introduced in [13]. Lets define the augmented state variable, $\widetilde{x}$,and the matrix $\widetilde{P}$ as,

$$\widetilde{x} = \begin{bmatrix} x \\ e \end{bmatrix}, \tag{5.4.1}$$

$$\widetilde{P} = \begin{bmatrix} P & 0 \\ 0 & \alpha G \end{bmatrix}. \tag{5.4.2}$$

Now, V can be written in terms of the new state variable:

$$V = \widetilde{x}^T\widetilde{P}\widetilde{x}. \tag{5.4.3}$$

To obtain the various control parameters we follow these steps:

Step 1: Find the set of initial conditions, $\widetilde{D}$, as restrictive as possible. This set must be large enough to include all the possible initial values of the augmented state variable $\widetilde{x}$ that may occur in practice.

Step 2: Choose the stabilizing feedback gain $\widetilde{L}$ in a way that:

$$g \leq \widetilde{L}^T \widetilde{x} \leq h \qquad \forall \widetilde{x} \in \widetilde{D}, \tag{5.4.4}$$

where, $\widetilde{L} \equiv [I \quad -I]^T L$ and L is a stabilizing feedback gain for the plant 5.2.1 and h, $g \in R^m$ are the upper bound and the lower bound on the input respectively.

Step 3: Choose the diagonal matrix K with positive elements and solve the Lyapunov equation 5.2.5 for some positive definite matrix Q. Then choose a positive constant α according to the inequality 5.3.5.

Step 4: Choose some positive definite matrix M and solve the Lyapunov equation 5.2.6 to get the matrix G. Construct the matrix $\widetilde{P}$ according to 5.4.2 and change the control parameters until the following inequality holds:

$$\sup_{\widetilde{x} \in \widetilde{D}} \widetilde{x}^T \widetilde{P} \widetilde{x} \leq \min_{\widetilde{x} \in \delta \widetilde{E}} \widetilde{x}^T \widetilde{P} \widetilde{x}. \tag{5.4.5}$$

Step 5: Compute the estimated state, $\hat{x}$, using a state observer and apply the stabilizing input 5.2.7 to the plant.

5.5 Simulation

To justify the second step, note that by the separation theorem, if $u = L^T x$ stabilizes the plant 5.2.1, the input $u = L^T \hat{x}$ stabilizes the whole system including the observer (provided that A_{ob} is stable too). Thus, the procedure for computing an acceptable $\widetilde{L}^T$ is to find stabilizing L, say by LQ method, and then define $\widetilde{L}$ as above. Note that we can write:

$$\widetilde{L}^T \widetilde{x} = L^T \hat{x}, \tag{5.5.1}$$

this means that:

$$\hat{x} \in \hat{E} \Leftrightarrow \tilde{x} \in \tilde{E}, \tag{5.5.2}$$

with the obvious definitions for $\hat{E}$ and $\tilde{E}$.

The implication 5.5.2 is important because in the proof of the stability we rely on the fact that $\hat{x}$ remains in the set $\hat{E}$ and by 5.5.2 we know that this is guaranteed provided that the following holds:

$$\widetilde{D} \subseteq \tilde{\Omega} \subseteq \tilde{E}. \tag{5.5.3}$$

Again, we can increase the input penalty so that $\tilde{L}$ satisfy 5.4.4. To see how efficient the proposed control strategy is, we apply the method to the problem of damping the vibration of a cantilever beam. We assume that the deflection angles are directly measurable and we design an observer to construct the other state variables (time derivative of the deflection angles).

5.5.1 Observability

Consider the state variable, x defined as the vector of alternating deflection angles and their respective corresponding time derivative:

$$x = \begin{bmatrix} \alpha_1 \\ \dot{\alpha}_1 \\ \vdots \\ \alpha_i \\ \dot{\alpha}_i \\ \vdots \\ \alpha_n \\ \dot{\alpha}_n \end{bmatrix}. \tag{5.5.4}$$

It can be shown that the A and C matrices describing the cantilever beam are of the form:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ * & 0 & * & 0 & * & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ * & 0 & * & 0 & * & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ * & 0 & * & 0 & * & 0 \end{bmatrix}, \tag{5.5.5}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{5.5.6}$$

here, for the sake of simplicity, we showed A and C for $n = 3$ ($*$ stands for nonzero elements).

Therefore, the observability matrix has the following form:

$$\mathcal{O} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{5.5.7}$$

which is full rank and consequently the pair (C, A) is observable.

5.5.2 Simulation Results

The observer settling time is chosen to be much faster than that of the closed loop system. Note that the faster you make the observer the larger the estimation error would be and the latter cause a larger initial condition set $\widetilde{D}$. Thus in a sense there is a trade off between the size of the set $\widetilde{D}$ and the speed of the observer. Figure 5.1 shows the estimation error of the observer for a particular choice of the parameters:

Now, we can set up the controller-observer according to the scheme shown in figure 5.2.

Although the analysis has been performed for the linear systems, simulation results show that the response of the linear and nonlinear systems are very similar.

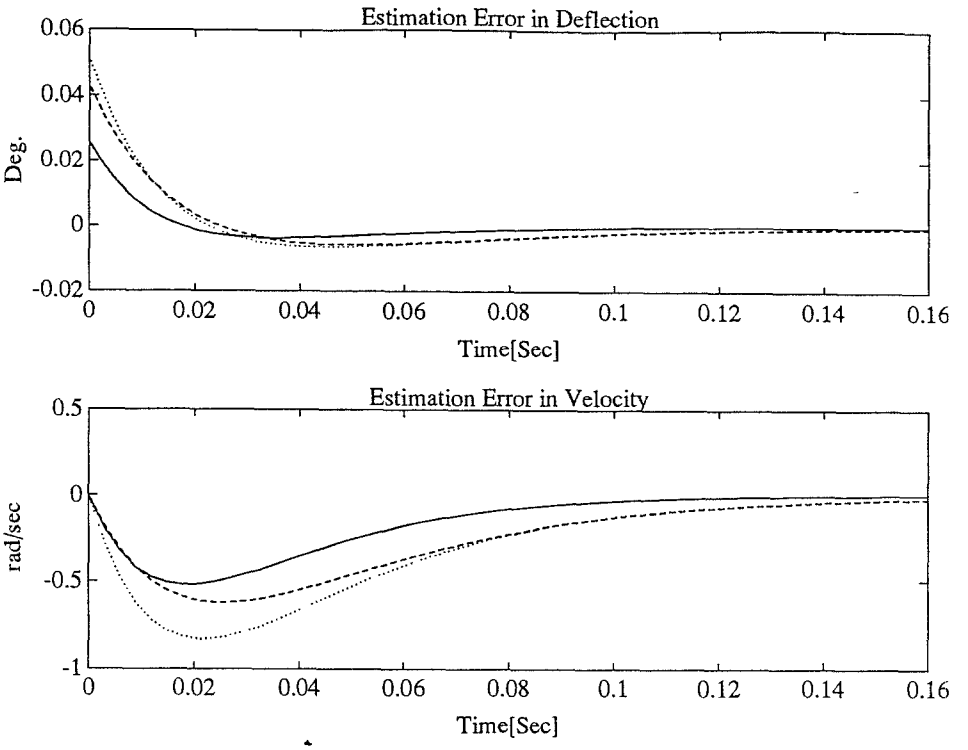

Figure 5.1: Estimation error in deflection and velocity

Shown in figure 5.3 is the response of the nonlinear model for the cantelevered beam when the scheme shown in figure 5.2 has been used to damp the vibrations.

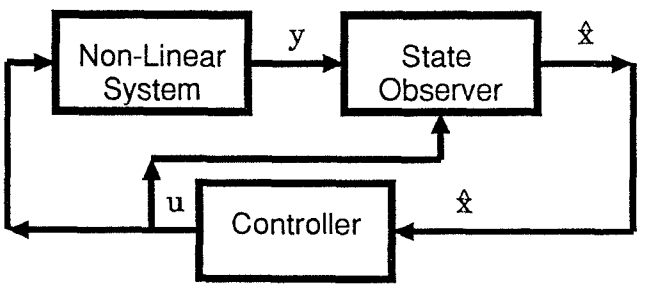

Figure 5.2: G-H type controller with observer

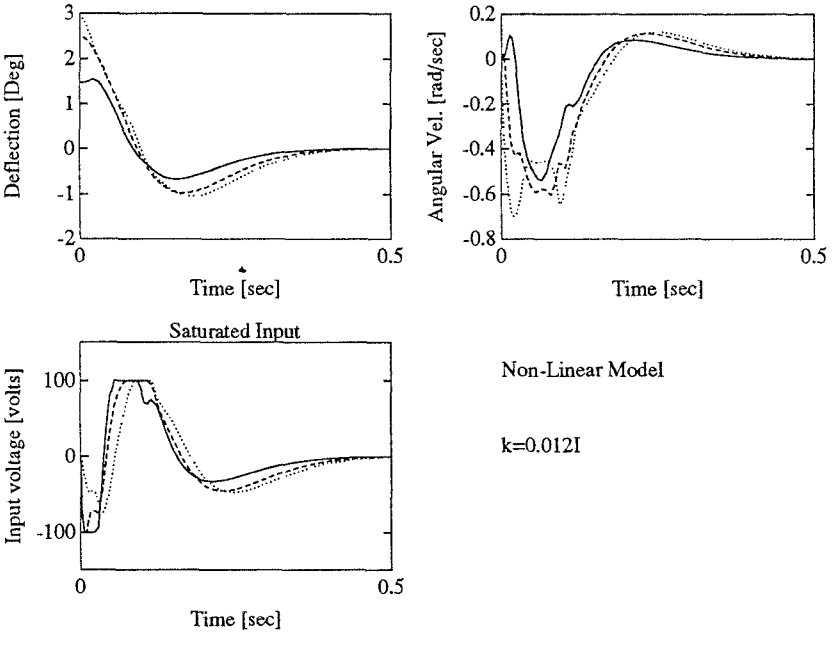

Figure 5.3: G-H type controller with observer

Chapter 6

Numerical Solution to the PDE for the Cantilever Beam With Control and Derivation of the Closed Form Transfer Function

The purpose of this chapter is to first develop a simulation tool to study the behavior of the PDE describing the transversal deflection of an Euler-Bernoulli beam and then to exploit it to derive an empirical transfer function matrix of a set up that is used to damp out the vibrations of the beam. To solve the PDE (numerically) an implicit finite difference method is used. The approximation is done using a scheme based on least squares proposed in [15], that employs rational wavelets. A closed form expression for the transfer function of the system is obtained at the end and comparison has been made with the empirical transfer function estimated by the spectral analysis.

6.1 Introduction

The transversal deflection of a beam can be described by a partial differential equation and proper boundary conditions which reflect the type of constraints at its ends.

In the process of designing a controller, to damp out the vibrations of the beam, one may fit a finite dimensional model to the beam or use some distributed parameter controller. In either case, to test the efficiency of the controller we need to apply the control method to a model of the beam, which takes into account the infinite dimensional nature of the system. Here, we developed a simulation program to solve the PDE pertaining to Euler-Bernoulli beam. At first, we had considered the free oscillation of the beam, and then to verify the results, the total energy of the beam as a function of time was computed (since no damping mechanism exists this should be a constant). Next, we include the effect of internal and viscous damping in the model to make it more realistic. With the simulation tool in hand we apply a swept sine wave as the input to the actuating piezo material and observe the voltage across the piezoceramics that are used as sensors. The input and output waveforms then enable us to get an empirical frequency responce. In the next step we fit a finite dimensional model to the frequency response.

6.2 Finite Difference Approximation

The following PDE describes the transversal vibrations of a thin beam [1],

$$\rho\frac{\partial^2 y}{\partial t^2}+\gamma\frac{\partial y}{\partial t}+\frac{\partial^2}{\partial x^2}(EI\frac{\partial^2 y}{\partial x^2}(t,x)+c_D I\frac{\partial^3 y}{\partial x^2\partial t}(t,x)+c_P V(t,x))=0, \quad (6.2.1)$$

where, $y = y(x, t)$ is the displacement from the equilibrium position at distance x along the beam from one end, γ is the coefficient of viscous damping, ρ is the linear mass density, and $c_D I$ is a coefficient reflecting the internal damping.

Define two new variables, v and w as below

$$\begin{cases} v &= y_t \\ w &= \sqrt{\frac{EI}{\rho}} y_{xx} - \frac{c_P}{\sqrt{\rho EI}} V + \frac{c_D I}{\sqrt{\rho EI}} y_{xxt} \end{cases} . \tag{6.2.2}$$

Now, PDE (6.2.1) can be written in the following form

$$\begin{cases} v_t &= -\frac{\gamma}{\rho} v - a w_{xx} \\ w_t &= a v_{xx} - \frac{c_P}{\sqrt{\rho EI}} V_t + \frac{c_D I}{\sqrt{\rho EI}} v_{xxt} \end{cases} \tag{6.2.3}$$

where $a = \sqrt{EI/\rho}$ and subscripts x and t are stand for partial derivatives with respect to x and t.

One possible finite difference representation of the above PDE is

$$\begin{cases} \frac{v_j^{n+1} - v_j^n}{\Delta t} &= -a \frac{(\delta^2 w)_j^{n+1} + (\delta^2 w)_j^n}{2(\Delta x)^2} - \frac{\gamma}{\rho} v_j^n \\ \frac{w_j^{n+1} - w_j^n}{\Delta t} &= (a - \frac{c_D I \gamma}{\sqrt{\rho EI}}) \frac{(\delta^2 v)_j^{n+1} + (\delta^2 v)_j^n}{2(\Delta x)^2} - \frac{c_P}{\sqrt{\rho EI}} V_t + \frac{a c_D I}{\sqrt{\rho EI}} \frac{(\delta^2 v)_j^{n+1} - (\delta^2 v)_j^n}{(\Delta x)^2 (\Delta t)} \end{cases} \tag{6.2.4}$$

where, superscripts and subscripts represent time and position indices, respectively and $\frac{\delta^2 w}{(\Delta x)^2}$ is the finite difference approximation of the second derivative of w with respect to x. That is,

$$(\delta^2 w)_j^n = w_{j+1}^n - 2w_j^n + w_{j-1}^n. \tag{6.2.5}$$

Of course, instead of (6.2.4), one could use some other scheme of finite difference approximation, but the advantage of using (6.2.4) is that it makes the difference equation unconditionally stable (i.e. stability does not depend on the

choice of step sizes Δx, and Δt). Nevertheless, system (6.2.4) entails other complications due to its implicit nature, and the extra effort required to solve the system (6.2.4).

We may cast (6.2.4) into the form

$$-Au_{j+1} + Bu_j - Au_{j-1} = d_j, \qquad j = 1, 2, \cdots, J-1 \tag{6.2.6}$$

in which,

$$\begin{aligned}
\alpha &= \tfrac{a\Delta t}{(\Delta x)^2}, \\
\beta &= 2[(a - \tfrac{c_D I\gamma}{\sqrt{\rho EI}})\tfrac{a\Delta t}{2(\Delta x)^2} + \tfrac{c_D Ia}{\sqrt{\rho EI}(\Delta x)^2} - \tfrac{c_P \Delta t}{\sqrt{\rho EI}} V_t], \\
u_j &= \begin{bmatrix} v_j^{n+1} \\ w_j^{n+1} \end{bmatrix}, \\
A &= \begin{bmatrix} 0 & -\frac{\alpha}{2} \\ \frac{\beta}{2} & 0 \end{bmatrix}, \\
B &= \begin{bmatrix} 1 & -\alpha \\ \beta & 1 \end{bmatrix}, \\
d_j &= \begin{bmatrix} v_j^n - \frac{\alpha}{2}(w_{j+1}^n - 2w_j^n + w_{j-1}^n) - \frac{\Delta t\gamma}{\rho} v_j^n \\ w_j^n + \frac{\beta}{2}(v_{j+1}^n - 2v_j^n + v_{j-1}^n) - \frac{c_P \Delta t}{\sqrt{\rho EI}} V_t \end{bmatrix}.
\end{aligned} \tag{6.2.7}$$

6.3 Preliminaries

In the algorithm that we are going to use, it is required that the boundary conditions to be specified in a form shown below (in the next section we show how to write down these equation for the cantilever beam)

$$u_{j_1} = Hu_{j_1+1} + l, \tag{6.3.1}$$

$$u_{J-j_2-1} = Mu_{J-j_2} + n, \tag{6.3.2}$$

with known matrices M, H and vectors n and l.

Now, consider the difference equation,

$$u_j = E_j u_{j+1} + f_j, \qquad j = 0, 1, \cdots \tag{6.3.3}$$

Equations (6.2.6) and (6.3.2) specify a two-parameter family of solutions of (6.2.6). Equation (6.3.3) also has a two-parameter family of solutions and these two families are identical if we choose E and f as,

$$\begin{aligned} E_{j_1} &= H, \\ f_{j_1} &= l, \\ E_j &= (B - AE_{j-1})^{-1}A, \\ f_j &= (B - AE_{j-1})^{-1}(d_j + Af_{j-1}) \qquad j \geq 1. \end{aligned} \tag{6.3.4}$$

Rewrite (6.3.3) for the index $J - j_2 - 1$ to get

$$u_{J-j_2-1} = E_{J-j_2-1}u_{J-j_2} + f_{J-j_2-1}. \tag{6.3.5}$$

Eliminating u_{J-j_2-1} between (6.3.5) and (6.3.2) give us u_{J-j_2} in terms of other variables:

$$u_{J-j_2} = (E_{J-j_2-1} - M)^{-1}(n - f_{J-j_2-1}). \tag{6.3.6}$$

Now that we have defined the various variables involved, we can state the algorithm.

6.4 Algorithm

Assume that the state of the system, $u_j^n, \quad j = 0,\ 1,\ \cdots,\ J$, for some n is known. The algorithm described below shows how to obtain $u_j^{n+1}, \quad j = 0,\ 1,\ \cdots,\ J$ at the next time step.

- *Step 1:* Solve the difference equations (6.3.4) for $j = j_1,\ j_1+1,\ \cdots, J-1$ with the initial values specified. (For the problem that we want to study, it can be shown that E_i and f_i both remain bounded)
- *Step 2:* Using the matrix E_{J-j_2-1} and the vector f_{J-j_2-1} compute u_{J-j_2} from equation (6.3.6).
- *Step 3:* Compute $u_i, \quad i = J-j_2-1,\ \cdots,\ j_1+1,\ j_1$ by solving (6.3.3) backward in j.
- *Step 4:* It remains to obtain a few of the state variables, u_i, at each end. i.e. $u_0,\ \cdots, u_{j_1-1}$ at one end and $u_J,\ \cdots,\ u_{J-j_2+1}$ at the other end. These can be obtained from (6.2.6)

6.5 Boundary Conditions for the Cantilever Beam

In order to use the above algorithm we need to specify the boundary conditions for the cantilever beam in the form of equations (6.3.2) and (6.3.2). So, the

parameters j_1, j_2, M, H, n, and l are need to be computed. The boundary conditions are:

$$\begin{cases} y(t,0) = y_x(t,0) = 0 & \forall t > 0 \quad (clamped\ end) \\ M(t,L) = M_x(t,L) = 0 & \forall t > 0 \quad (free\ end) \end{cases} \tag{6.5.1}$$

where, M is the total moment (internal and external) defines by:

$$M(x,t) = \frac{\partial^2}{\partial x^2}(EI\frac{\partial^2 y}{\partial x^2}(t,x) + c_D I\frac{\partial^3 y}{\partial x^2 \partial t}(t,x) + c_P V(t,x)). \tag{6.5.2}$$

The boundary condition at the clamped end implies that $v_0 = v_1 = 0$. Equation (6.2.6) together with the boundary conditions at the clamped end have a solution which is linearly dependent on two parameters. Choose $\lambda \equiv [w_0\ w_1]^T$ as the vector of parameters. Then one can write

$$u_j = U_j\lambda + g_j, \qquad j = 0,\ 1,\ \cdots,\ J, \tag{6.5.3}$$

where, U_j is defined by,

$$(U_j)_{ik} = \frac{\partial u_{ji}}{\partial \lambda_k}. \tag{6.5.4}$$

The procedure is to compute U_i for $i = 0,\ 1,\ \cdots,\ j_1$ such that U_{j_1} is the fist invertible matrix in that sequence of matrices. Then we have

$$\lambda = U_{j_1}^{-1}(u_{j_1} - g_{j_1}), \tag{6.5.5}$$

which in turn results,

$$u_{j_1-1} = U_{j_1-1}U_{j_1}^{-1}(u_{j_1} - g_{j_1}) + g_{j_1-1} \tag{6.5.6}$$

which is in the form of (6.3.2) with H and l as below,

$$\begin{aligned} H &= U_{j_1-1}U_{j_1}^{-1}, \\ l &= -U_{j_1-1}U_{j_1}^{-1}g_{j_1} + g_{j_1-1}. \end{aligned} \tag{6.5.7}$$

For the problem considered here U_2 is the first nonsingular one and it can be readily verified that the previous ones are,

$$U_0 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \tag{6.5.8}$$

$$U_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \tag{6.5.9}$$

$$U_2 = \begin{bmatrix} 0 & \frac{2}{\beta} \\ -1 & 2 \end{bmatrix}. \tag{6.5.10}$$

Now, we need to compute g_1 and g_2:

$$\begin{aligned} g_2 &= u_2 - U_2\lambda \\ &= A^{-1}Bu_1 - u_0 - A^{-1}d_1 - U_2\lambda \\ &= -A^{-1}d_1. \end{aligned} \tag{6.5.11}$$

Also, for g_1 we have,

$$\begin{aligned} g_1 &= u_1 - U_1\lambda \\ &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \end{aligned} \tag{6.5.12}$$

Thus,

$$j_1 = 2, \quad H = U_1U_2^{-1}, \quad l = U_1U_2^{-1}A^{-1}d_1. \tag{6.5.13}$$

Now, consider the boundary conditions at the free end of the beam. Again, from (6.5.1) we get $w_J = w_{J-1} = 0$ and this time we choose $\lambda \equiv [v_J \quad v_{J-1}]^T$ as the parameters. Similarly, define U_j by,

$$(U_j)_{ik} = \frac{\partial u_{ji}}{\partial v_{J-k+1}}. \tag{6.5.14}$$

Going backward in position from the free end toward the clamped end, U_{J-2} is the first nonsingular matrix and we have,

$$U_J = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \tag{6.5.15}$$

$$U_{J-1} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \tag{6.5.16}$$

$$U_{J-2} = \begin{bmatrix} -1 & 2 \\ 0 & \frac{-2}{\alpha} \end{bmatrix}. \tag{6.5.17}$$

Thus, $j_2 = 2$. Following the same steps as for the clamped end but this time for the free end, one gets,

$$\lambda = U_{J-2}^{-1}(u_{J-2} - g_{J-2}), \tag{6.5.18}$$

and,

$$u_{J-3} = U_{J-3}U_{J-2}^{-1}(u_{J-2} - g_{J-2}) + g_{J-3}. \tag{6.5.19}$$

Again, it can be shown that $g_{J-2} = -A^{-1}d_{J-1}$. To obtain g_{J-3} first we

need to compute U_{J-3}. So, write

$$\begin{aligned} u_{J-3} &= A^{-1}Bu_{J-2} - u_{J-1} - A^{-1}d_{J-2}, \\ &= A^{-1}B(A^{-1}Bu_{J-1} - u_J - A^{-1}d_{J-1}) - u_{J-1} - A^{-1}d_{J-2}, \\ &= \begin{bmatrix} 3 - \frac{4}{\alpha\beta} & \frac{8}{\beta} \\ -\frac{8}{\alpha} & 3 - \frac{4}{\alpha\beta} \end{bmatrix} u_{J-1} - A^{-1}Bu_J - A^{-1}BA^{-1}d_{J-1} - \\ & A^{-1}d_{J-2}. \end{aligned} \tag{6.5.20}$$

from which we can easily obtain U_{J-3}:

$$U_{J-3} = \begin{bmatrix} -2 & 3 - \frac{4}{\alpha\beta} \\ \frac{2}{\alpha} & -\frac{8}{\alpha} \end{bmatrix}. \tag{6.5.21}$$

Substitute for u_{J-3} and U_{J-3} into the relation for g_{J-3} and simplify to get

$$\begin{aligned} g_{J-3} &= u_{J-3} - U_{J-3}\lambda, \\ &\hat{=} -A^{-1}BA^{-1}d_{J-1} - A^{-1}d_{J-2}. \end{aligned} \tag{6.5.22}$$

To compute M and n, plug in the expressions that we obtained for g_{J-3}, g_{J-2}, U_{J-3}, and U_{J-2} into (6.5.19)

$$\begin{aligned} M &= U_{J-3}U_{J-2}^{-1}, \\ n &= U_{J-3}U_{J-2}^{-1}A^{-1}d_{J-1} - A^{-1}d_{J-1} - \\ & \quad A^{-1}BA^{-1}d_{J-1} - A^{-1}d_{J-2}. \end{aligned} \tag{6.5.23}$$

So, we have computed all the terms that are required by the algorithm.

6.6 Simulation

To implement the algorithm mentioned above, first we must specify the initial shape and velocity of the beam. We assumed a uniform load on the beam of length $1m$ and the load is such that it results in a deflection of $8cm$ at the tip. The deflection, y, at $t = 0$ and at distance x from the clamped end is [5],

$$y = \frac{\delta}{3L^4}(6L^2x^2 - 4Lx^3 + x^4). \tag{6.6.1}$$

where, δ is the maximum deflection (which occurs at the tip) and L is the length of the beam.

First, we set both the internal and damping coefficients to zero. This gives us a way to verify the correctness of the algorithm. Because in the absence of damping, the total energy must remain constant with time. Figure (6.1) shows the velocity and deflection of various fixed point on the beam versus time.

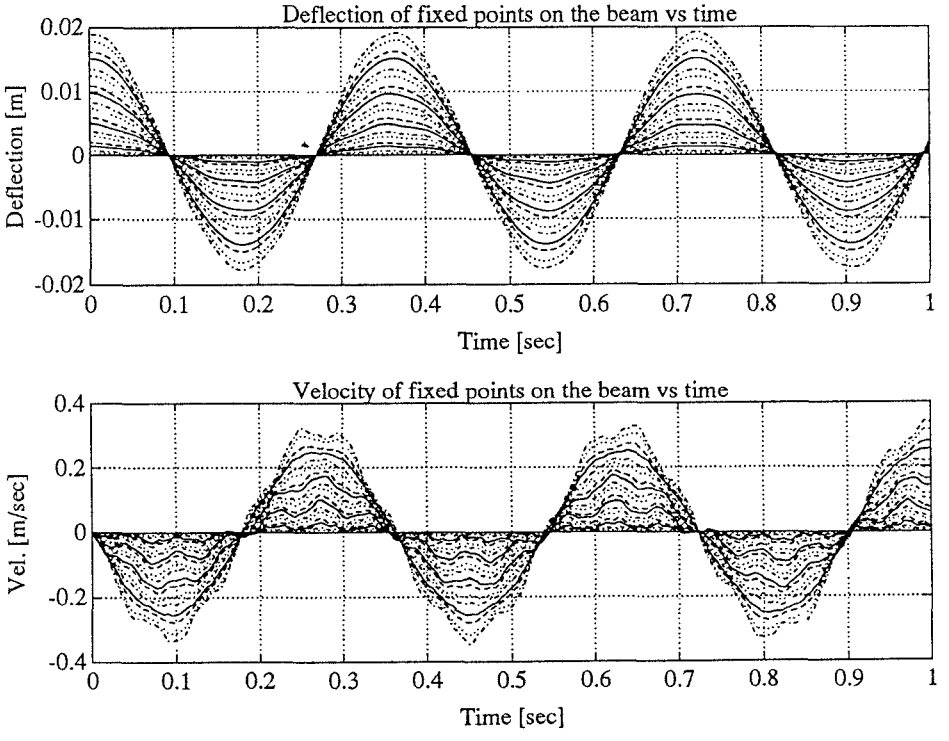

Figure 6.1: Velocity and deflection as a function of time (no damping)

The velocity and deflection at the free end of the beam are shown in figure (6.2)

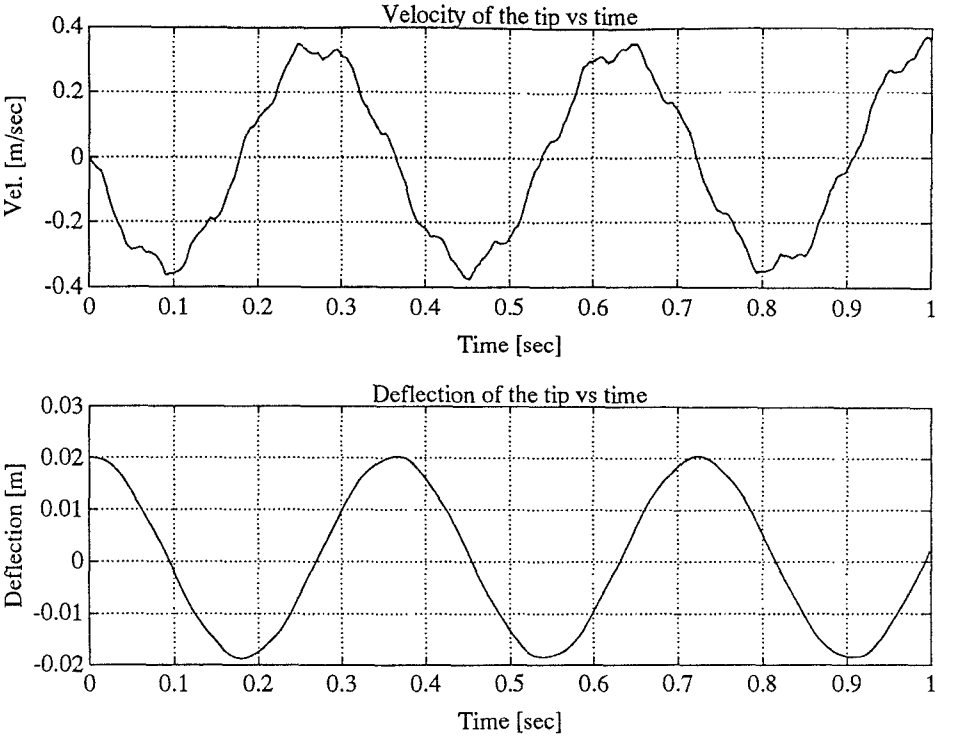

Figure 6.2: Velocity and deflection of the free end as a function of time

Also, we can show the shape of the beam as it oscillates. Each curve in figure (6.3) represents the shape of the beam at a fixed instant of time.

The energy of the system at time t is obtained from the following relation:

$$E(t) = \int_0^L (\rho(\frac{\partial y}{\partial t})^2 + EI(\frac{\partial^2 y}{\partial x^2})^2)\ dx. \tag{6.6.2}$$

Using the first order approximation of the integral the above equation can be easily written in terms of the variables v and w introduced at the beginning of the chapter,

$$E^n = \sum_{j=0}^{J} (\rho((v_j^n)^2) + (w_j^n)^2)). \tag{6.6.3}$$

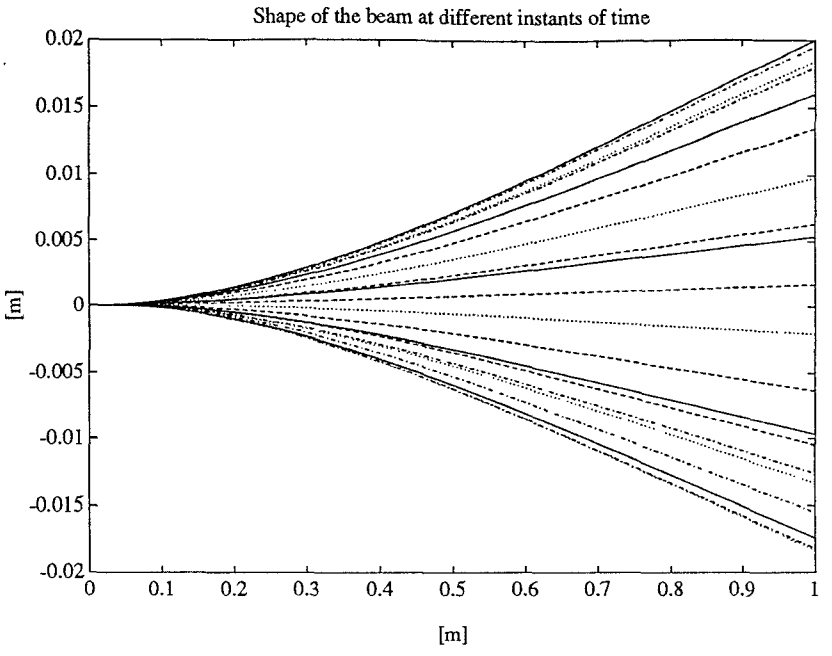

Figure 6.3: Free oscillation of the beam

Since the deviation of the energy from its average value is very small, in figure (6.4) we show the percentage of deviation of it versus time.

As it can be seen, the error is relatively small and perhaps the major part of the error is due to the first order approximation that we used for integration of (6.6.2). Introducing internal and viscous damping give us the following set of graphs.

6.7 Estimating the Empirical Transfer Function

We are interested in finding the transfer function from the output which is the voltage across piezoceramics attached to the beam and operate as sensors, to

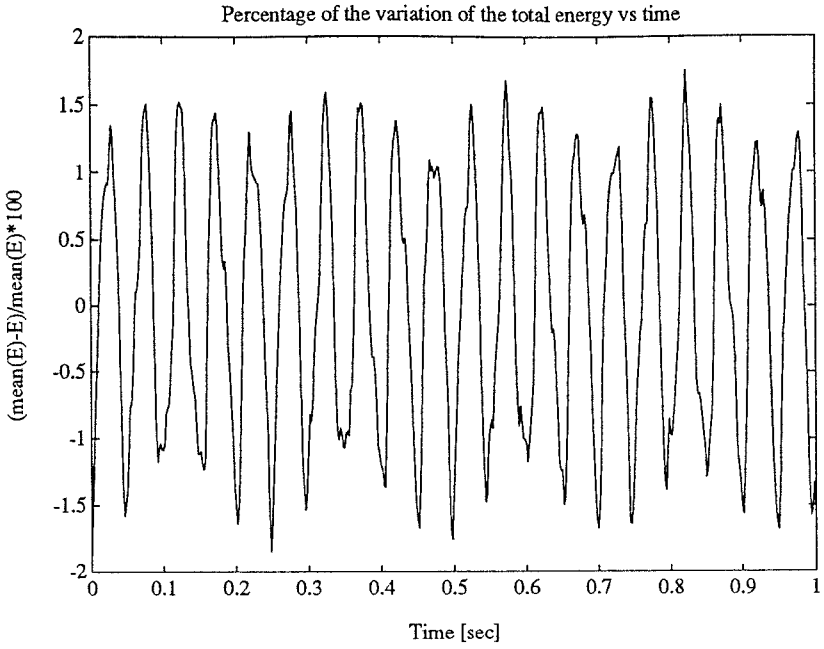

Figure 6.4: The energy stored in the beam remains constant.

the input voltage that is applied to the actuating piezoceramics. The model that we use is shown in Fig. (6.7).

Since the PDE (6.2.1) is linear, we may use the ratio of Fourier transform of the output to that of the input and this give an estimate of the transfer function [9]. That is,

$$\widehat{G}(e^{i\omega}) = \frac{Y_N(\omega)}{U_N(\omega)}, \tag{6.7.1}$$

and $Y_N(\omega)$ is defined by,

$$Y_N(\omega) = \frac{1}{\sqrt{N}} \sum_{t=1}^{N} y(t) e^{-i\omega t}. \tag{6.7.2}$$

It can be shown that [9] the following relationship holds between the empirical transfer function, $\widehat{G}(e^{j\omega})$, and the true transfer function $G(e^{j\omega})$:

$$\widehat{G}(e^{j\omega}) = G(e^{j\omega}) + \frac{R_N(\omega)}{U_N(\omega)}, \tag{6.7.3}$$

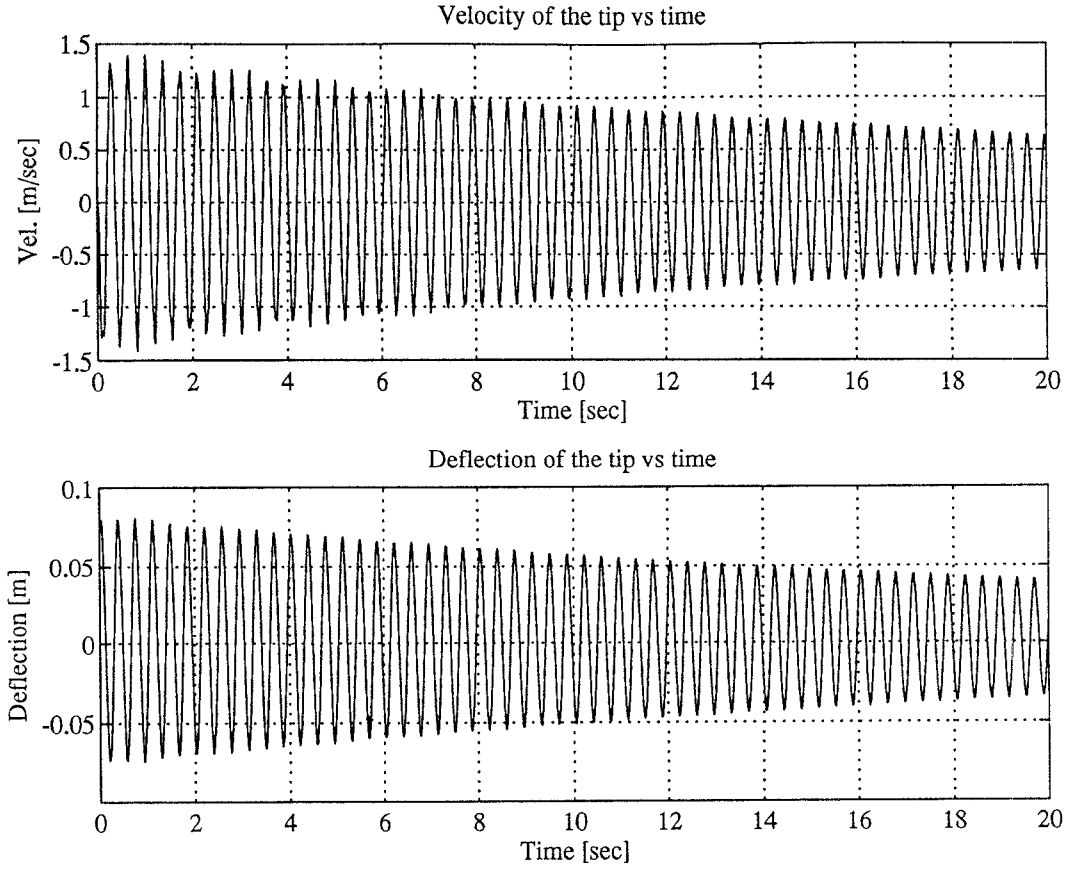

Figure 6.5: Velocity and deflection as a function of time (with passive damping)

where $R_N(\omega)$ decays as $1/\sqrt{N}$.

In the case of periodic input the above procedure gives a good estimation of the transfer function around the frequencies that exist in the periodic input. But, if the input is not periodic we must assume that the values of the true transfer function are related at different frequencies. As an alternative procedure, we may obtain the spectral estimate $\widehat{G}(e^{j\omega})$ from the expression,

$$\widehat{G}(e^{j\omega}) = \frac{\widehat{\Phi}_{yu}^N(\omega)}{\widehat{\Phi}_u^N(\omega)}, \tag{6.7.4}$$

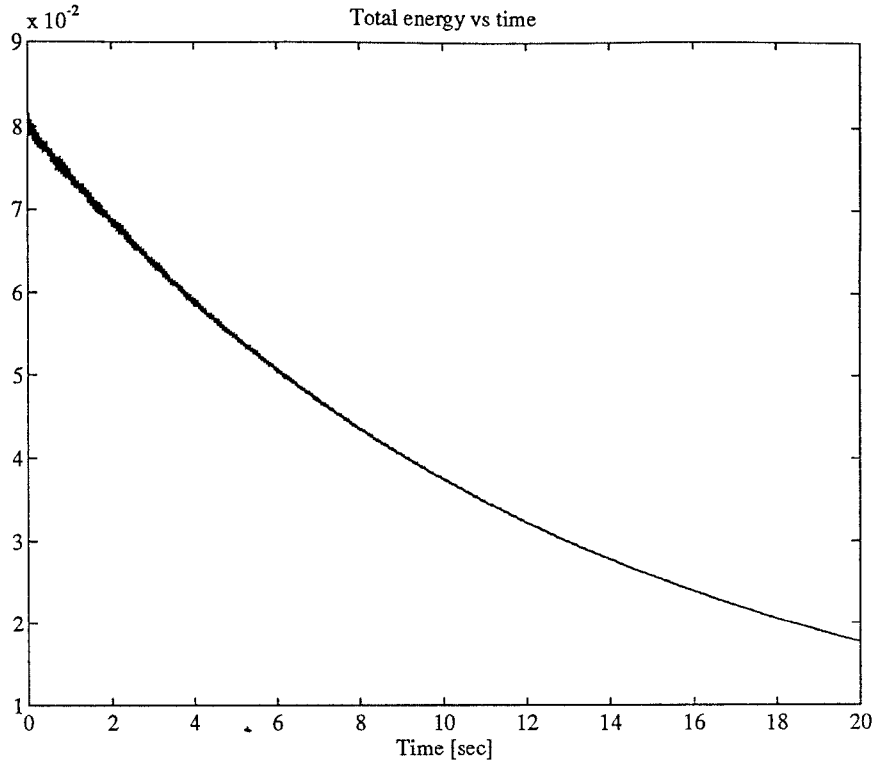

Figure 6.6: The energy stored in the beam decreases with time as a result of passive damping

where, $\widehat{\Phi}_u^N(\omega)$ is the power spectrum of the input and is obtained from,

$$\widehat{\Phi}_u^N(\omega) = \sum_{\tau=-\infty}^{\infty} w_\gamma(\tau)\widehat{R}_u^N(\tau)e^{-i\tau\omega},$$

with $\widehat{R}_u^N(\tau)$ as the autocorrelation function defined as,

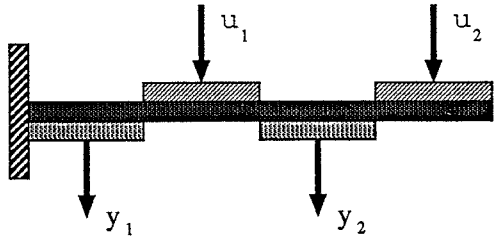

Figure 6.7: The inputs and outputs to the system

$$\widehat{R}_u^N(\tau) \;=\; \frac{1}{N}\sum_{t=1}^{N} u(t)u(t-\tau),$$

and $w_\gamma(\tau)$ is a lag window with shape parameter γ and is chosen such that vanishes for $|\tau| > \delta_\gamma$ where $\delta_\gamma << N$. This allows us to approximate 6.7.5 with [9].

$$\widehat{\Phi}_u^N(\omega) \;=\; \sum_{\tau=-\delta_\gamma}^{\delta_\gamma} w_\gamma(\tau)\widehat{R}_u^N(\tau)e^{-i\tau\omega}.$$

The expression for $\widehat{\Phi}_{yu}^N(\omega)$ is similar. There are different choices for the lag window $w_\gamma(\tau)$ among which we use the Hamming lag window function defined as [9].

$$w_\gamma(\tau) \;=\; \frac{1}{2}(1+\cos\frac{\pi\tau}{\gamma}) \qquad |\tau| \le \gamma, \tag{6.7.5}$$

here, the scaling parameter γ is chosen to be equal to δ_γ in 6.7.5. The effect of choosing a relatively large δ_γ is that it results in a large bias in the estimated transfer function, whereas decreasing δ_γ results a larger variance in $\widehat{G}(e^{j\omega})$.

To get a "good" estimate of the transfer function the test input should be as "informative" as possible. A good choice may be a white noise because its spectrum ideally contains all frequencies. However, the input voltage to the piezoceramics , $V(x,t)$, must be differentiable (with respect to time) as it can be seen from 6.2.1. In fact, piezoceramics behave like small capacitors and a discontinuity of the voltage across them theoretically requires an infinite current. A sweeping sinusoid (frequency modulation of a triangular waveform) seems to be a reasonable input because it is smooth enough and more importantly covers a wide range of frequencies.

To demonstrate the effectiveness of the method and also to verify the mentioned procedure, we used the above method to obtain the frequency responce of a known system. Note that here we are obtaining the frequency responce just by one experiment and we do not have to wait until the settling time of the system. The following system has been used as the test model,

$$H(s) = \frac{289.}{s^2 + 0.5s + 289.}. \tag{6.7.6}$$

As input, we excited the system with the frequency modulated of a sinusoid that varies between zero and ω_{max}. i.e.

$$u(t) = \sin(\omega(t)t), \tag{6.7.7}$$

with,

$$\omega(t) = \omega_{max}(1 - \cos\omega_0 t)/2. \tag{6.7.8}$$

The input-output waveforms together with the empirical frequencies are shown in figures (6.8) and (6.9).

The same procedure now can be use to form the empirical transfer function of the beam. Here we have two inputs and two outputs. Since the superposition principle holds it is a meaningful thing to talk about the matrix transfer function. In the set up shown in figure (6.7) there are two inputs and two outputs.

The outputs are designed to be the voltages that are induced in the piezoceramics due to bending of the beam. Figure (6.10) shows a portion of the flexible beam which is bended. Assume that the piezo material is attached to the upper (or lower) surface of the beam and the objective is to compute the change of length in it. The neutral axis is shown with the dotted line and its length would not change due to bending since axial forces has not been considered. The change in slope for the neutral axis is,

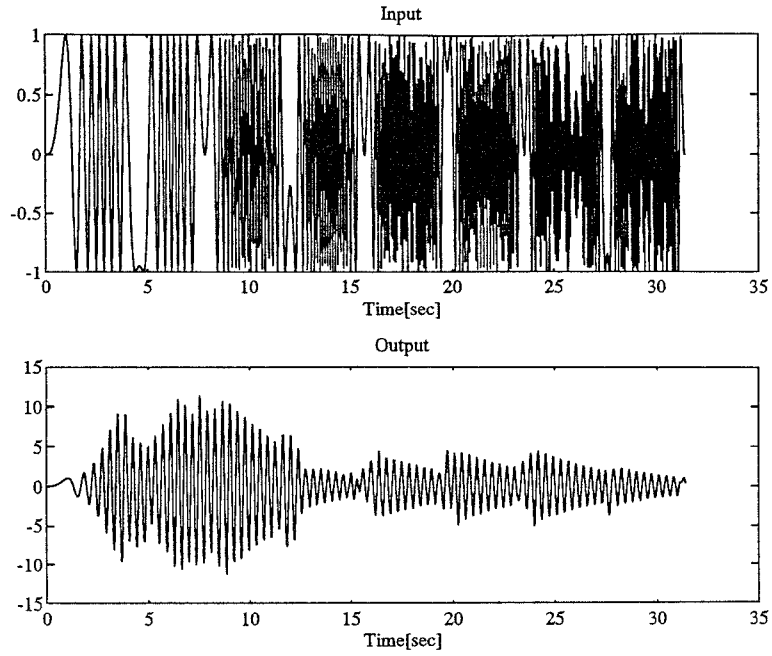

Figure 6.8: The inputs and outputs to the system

$$\Delta\theta \;=\; y_{xx}(x)dx.$$

With c the distance from neutral axis to the outer surface and from the equal angles we have,

$$\frac{e}{c} \;=\; \Delta\theta \;=\; y_{xx}dx.$$

Thus, the strain in the outer surface is

$$\epsilon(x) \;=\; \frac{e}{dx} \;=\; cy_{xx}(x).$$

The total change of length in a patch with its ends at x_1 and x_2 is obtained

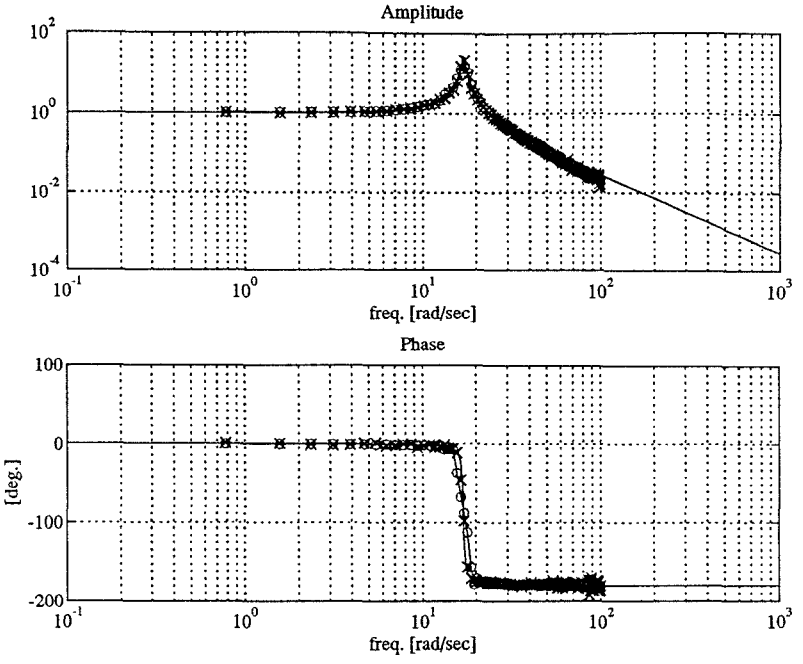

Figure 6.9: Bode plot of the true system together with the estimated ones

by integration,

$$\begin{aligned} \Delta L &= \int_{x_1}^{x_2} \epsilon(\sigma) d\sigma \\ &= \int_{x_1}^{x_2} c y_{xx}(\sigma) d\sigma \\ &= c(y_x(x_2) - y_x(x_1)). \end{aligned}$$

The induced voltage in the piezoceramic is proportional to the total change in length and the constant of proportionality if called d_{31}

$$\begin{aligned} V_{out} &= d_{31} \Delta L \\ &= d_{31} c(y_x(x_2) - y_x(x_1)). \end{aligned} \tag{6.7.9}$$

Shown in figure (6.11) is the Bode plot obtained by the spectral analysis described above (Eq. 6.7.4).

Since the damping that has been introduced into the model is fairly small, we still may expect that the eigenvalues of the system do not change considerably.

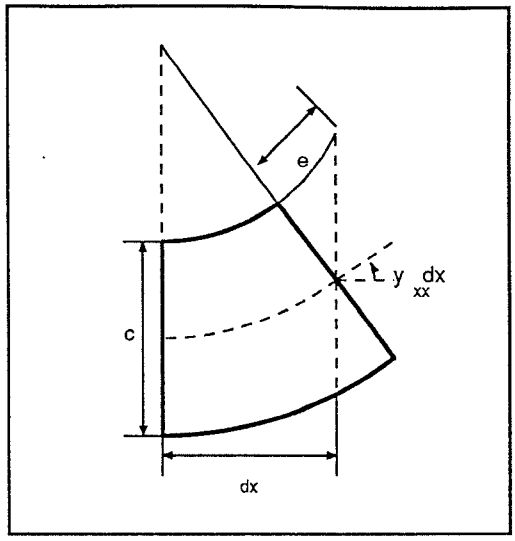

Figure 6.10: The strain induced in piezo material

The eigenvalues of the undamped system can be computed analytically [8] and the first few of them are,

$$\omega_i = \lambda_i^2 \sqrt{\frac{EI}{\rho}}, \tag{6.7.10}$$

where,

$$\begin{aligned} \lambda_1 &= 1.875/L, \\ \lambda_2 &= 4.695/L, \\ \lambda_3 &= 7.855/L, \\ \lambda_4 &= 10.996/L, \\ \lambda_5 &= 14.137/L, \end{aligned}$$

Based on the above figures and the parameters used in the first fifth simulation the natural modes are,

$$\omega_1 = 16.68, \quad \omega_2 = 104.57, \quad \omega_3 = 292.83,$$
$$\omega_4 = 573.852, \quad \omega_5 = 948.516,$$

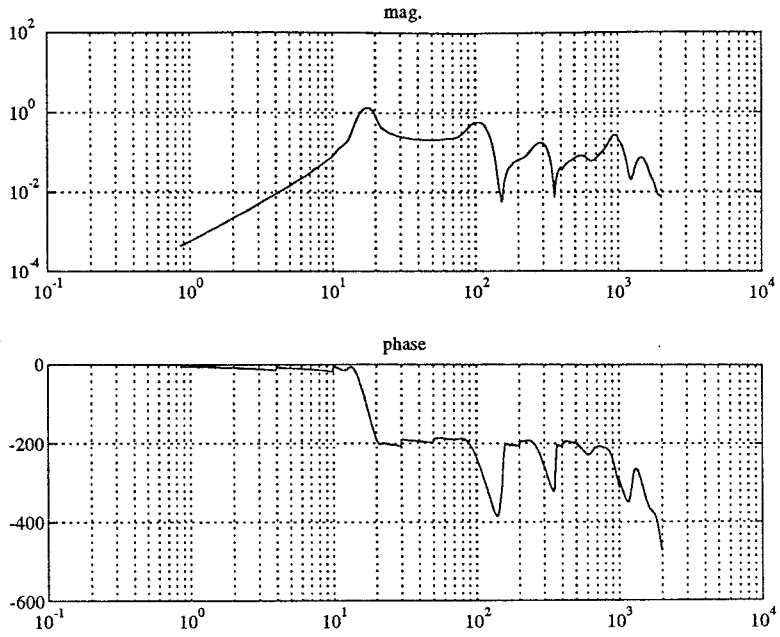

Figure 6.11: Bode plot of the empirical transfer function from u_1 to y_1

which closely match the frequencies at which the peaks of the modulus of the transfer function in Fig. (6.11) occurs.

6.8 Fitting a Finite Dimensional Transfer Function to the Empirical Data

The next and the last step in the non-parametric estimation of the transfer function is to actually obtain a finite dimensional transfer function that describes the input-output behavior of the system with a reasonable accuracy. Inspecting the Bode plot shown in Fig. (6.11), one note that it is well localized in the frequency domain. Roughly speaking, this means that the transfer function is more concentrated at certain frequencies.

ξ	γ	a_0	b_0
1.0	1.60	2.0	3.33

Table 6.1: Parameters of the analyzing wavelet and translation-dilation step sizes

The shape of the transfer function (localization property) together with the stability of the overall system (due to internal and viscous damping) motivate us to use the method of rational wavelets to approximate the transfer function [2].

This method has shown to be superior (e.g. in terms of the degree of the transfer function used as approximation) to the conventional methods such as the Laguerre model.

There are several parameters that one can change to get a good approximation with rational wavelets. One of the important factors is the analyzing wavelet. Unfortunately, there is no systematic way to choose such a wavelet and usually one has to find a proper analyzing wavelet through experiment.

The translation step size is another parameter that has a dramatic effect on the approximation. In Fig. (6.12) the result of a high order approximation is shown. For practical purposes we have only considered the frequency responce up to $200 rad/sec$. Figure (6.13) shows the results for a rational transfer function of degree 42.

The following function is used as the analyzing wavelet

$$\Psi(s) = \frac{1}{(s+\gamma)^2+\xi}. \tag{6.8.1}$$

The dilation and translation step sizes, γ, and ξ are as in Table (6.1).

Figure 6.14 depicts the frequency-time localization of the empirical transfer

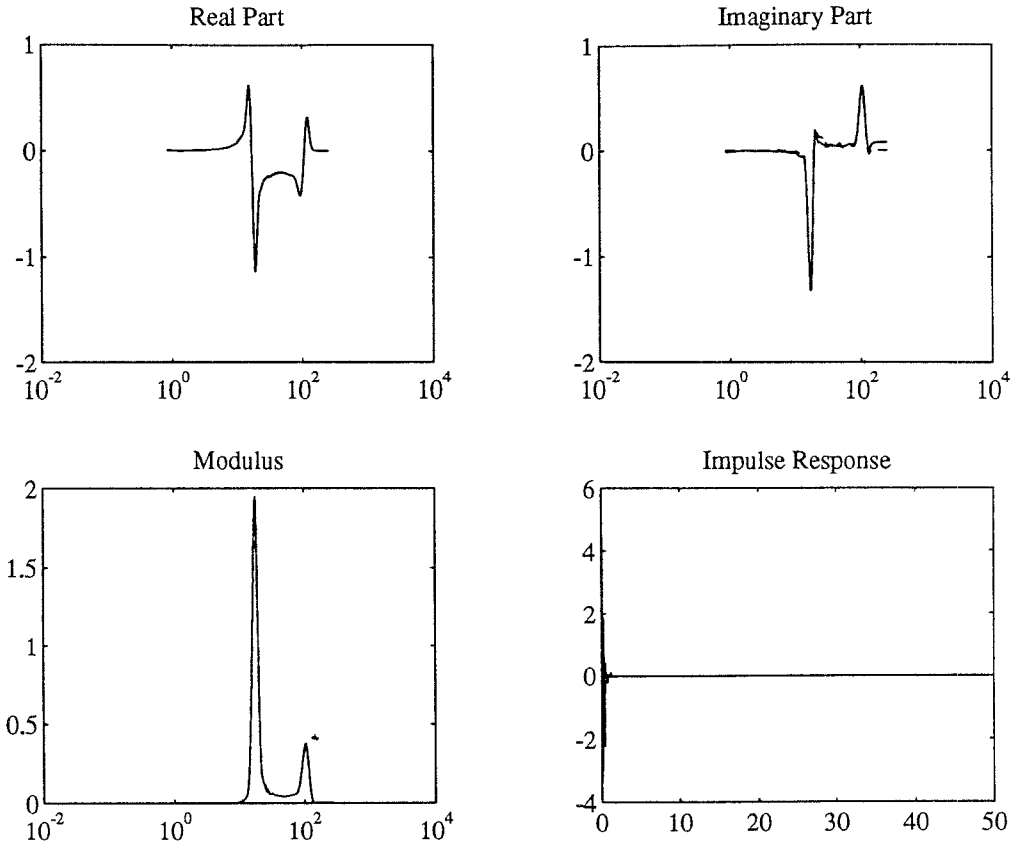

Figure 6.12: High order approximation (least squares)

function obtained. It can be seen from this figure that the certain coefficients are considerably bigger (and thus more important) than the other and the usage of wavelet systems for approximation is justified. Also the poles and zeros of the approximated transfer function are shown in 6.15.

6.9 Closed Form of the Transfer Function

In order to compare the empirical transfer function obtained in the previous section with the analytical one, here we derive a closed form expression for the

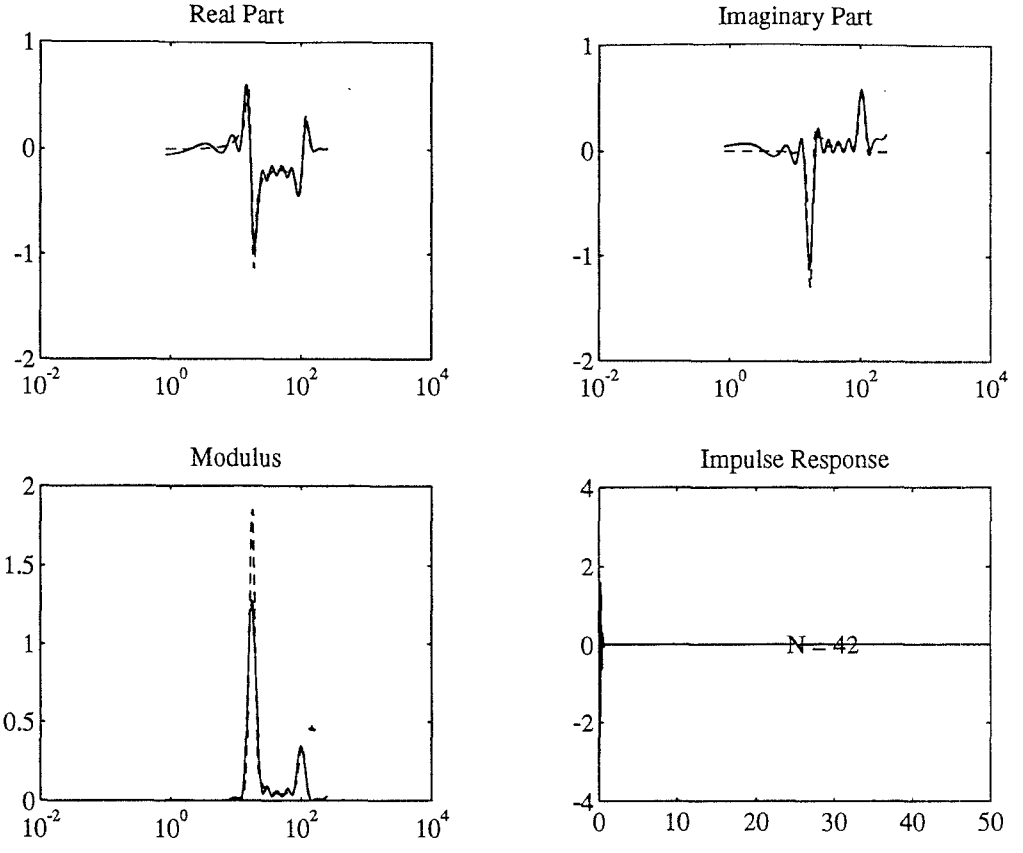

Figure 6.13: Approximation with a transfer function of degree 42 (least squares)

transfer function.

This transfer function would not be a rational one because it is derived from a partial differential equation rather than an ODE. Consider the following PDE which corresponds to the vibration without damping,

$$\begin{cases} \rho y_{tt}(x,t) + M_{xx}(x,t) = 0, \\ M(x,t) = EIy_{xx}(x,t) + C_p V(x,t), \\ y(0,t) = y_x(0,t) = 0, \\ y(x,0) = y_t(x,0) = 0, \\ M(L,t) = M_x(L,t) = 0. \end{cases} \tag{6.9.1}$$

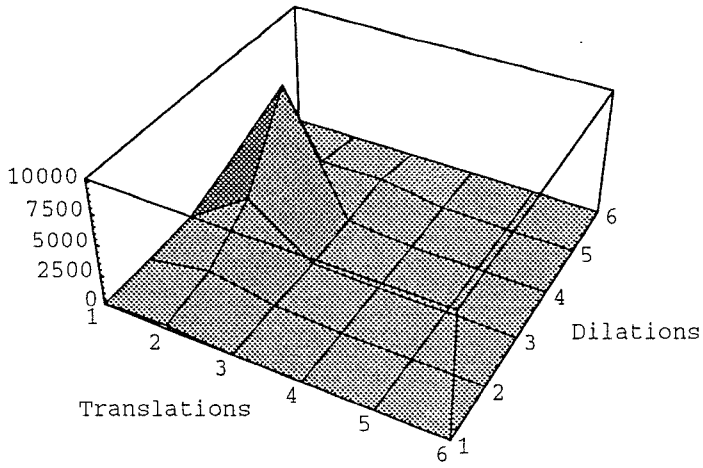

Figure 6.14: Magnitudes of the coefficients of the wavelet system vs. dilation and translation

Take the Laplace transform (with respect to t) to eliminate t and obtain a differential equation with respect to x.

$$\begin{cases} \rho s^2 Y(x,s) + \frac{d^2}{dx^2} M(x,s) = 0, \\ M(x,s) = EI \frac{d^2}{dx^2} Y(x,s) + C_p V(x,s), \\ y(0,s) = \frac{d}{dx} Y(0,s) = 0, \\ M(L,s) = M_x(L,s) = 0. \end{cases} \tag{6.9.2}$$

Define the state variables as,

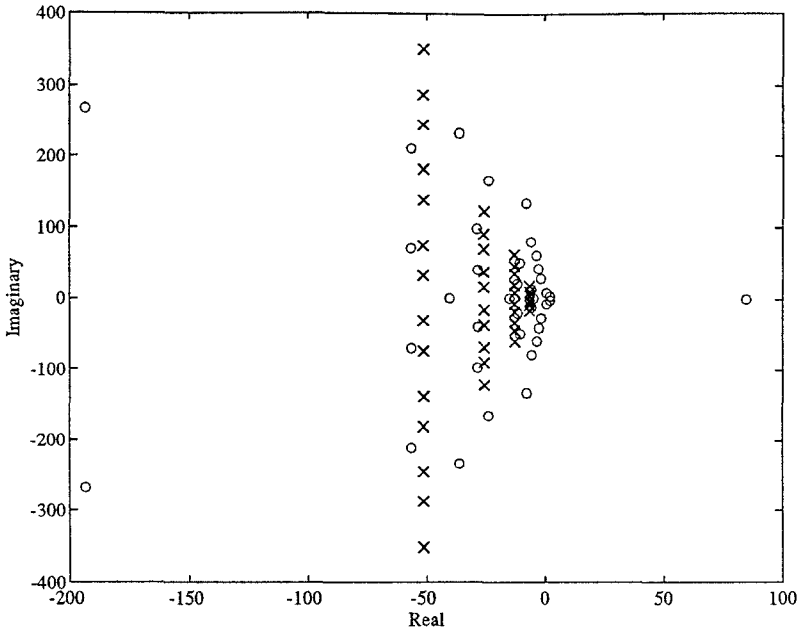

Figure 6.15: Pole and zeroes of the approximated transfer function

$$\begin{cases} u_1(x) &= Y(x,s), \\ u_2(x) &= \frac{d}{dx}Y(x,s), \\ u_3(x) &= M(x,s), \\ u_4(x) &= \frac{d}{dx}M(x,s), \\ u_1(0) &= u_2(0) = 0, \\ u_3(L) &= u_4(L) = 0. \end{cases} \tag{6.9.3}$$

The state space representation would be

$$U'(x) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{EI} & 0 \\ 0 & 0 & 0 & 1 \\ -\rho s^2 & 0 & 0 & 0 \end{bmatrix} U(x) + \begin{bmatrix} 0 \\ \frac{-C_p}{EI} \\ 0 \\ 0 \end{bmatrix} V(x). \tag{6.9.4}$$

The only difficulty with the above ODE is that u_1 and u_2 are specified at one end while u_3 and u_4 are given at the other end. The variation of constant for the system (6.9.4) is

$$U(x) = e^{Ax}U(x) + \int_0^x e^{A(x-\sigma)}BV(\sigma, s)d\sigma, \tag{6.9.5}$$

with the obvious definitions for the matrices A and B and the initial condition,

$$U(0) = \begin{bmatrix} 0 \\ 0 \\ u_3(0) \\ u_4(0) \end{bmatrix}. \tag{6.9.6}$$

To obtain $u_3(0)$ and $u_4(0)$ evaluate Eq. (6.9.5) at $x = L$ and collect the last two rows. i.e.

$$0 = \begin{bmatrix} u_3(L) \\ u4(L) \end{bmatrix} = Ke^{AL}u(0) + K(\int_{x_1}^{x_2} e^{A(L-\sigma)}Bd\sigma)F(s), \tag{6.9.7}$$

where we assumed a uniform electric field, with respect to x, between x_1 and x_2 and zero elsewhere and with the Laplace transform $F(s)$ (see Fig. (6.16). Also K is given by,

$$K = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.9.8}$$

Solving for $u_3(0)$ and $u_4(0)$ we get,

$$\begin{bmatrix} u_3(0) \\ u_4(0) \end{bmatrix} = -(Ke^{AL}K^T)^{-1}K\left[\int_{x_1}^{x_2} e^{A(L-\sigma)}Bd\sigma\right]F(s), \tag{6.9.9}$$

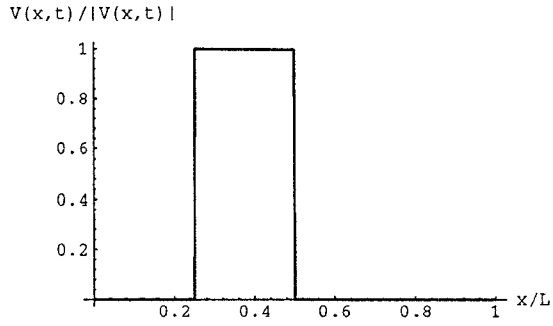

Figure 6.16: The electric field across the piezoceramic at one instant of time

Since we are only interested in deriving the transfer function and the output is proportional to the difference in slope of the beam (Eq. (6.7.9)), only the second state variable is important. Thus, to compute $u_2(x)$ substitute for the initial condition into Eq. (6.9.5) and pick the second row.

$$V_{out} = \begin{bmatrix} 0 & K_s & 0 & 0 \end{bmatrix} (Us_2 - U_{s_1}), \tag{6.9.10}$$

where, s_1 and s_2 specify the position of the sensor and K_s is as in Eq. (6.9.5). Using the above equations and with the use of *Mathematica* the following expression has been obtained for the transfer function,

$$\frac{V_{out}(s)}{F(s)} = \frac{N(s)}{D(s)}, \tag{6.9.11}$$

where, $N(s)$ and $D(s)$ are defined as,

$$
\begin{aligned}
N(s) \;=\; & \{\tfrac{C_p K_s L}{2EIz}[2\cos(3z/4)\cosh(3z/4) - 2\cos(z/2)\cosh(z/2)] \\
& [-\cosh(3z/4)\sin(3z/4) + \cosh(3z/4)\sin(5z/4) - \\
& \cos(3z/4)\sinh(3z/4) + \cos(3z/4)\sinh(5z/4)] + \\
& [\cosh(z/2)\sin(z/2) - \cosh(3z/4)\sin(3z/4) - \\
& \cos(z/2)\sinh(z/2) + \cos(3z/4)\sinh(3z/4)] \\
& [\cos(5z/4)\cosh(3z/4) - \cos(3z/4)\cosh(5z/4) + \\
& 2\sin(3z/4)\sinh(3z/4) - \sin(5z/4)\sinh(3z/4) - \sin(3z/4)\sinh(5z/4)]\},
\end{aligned}
$$

$$
D(s) \;=\; 2 + \cos(z) + \cosh(z),
$$

$$
z \;=\; \sqrt{2}L(\frac{\rho}{EI})^{1/4}\sqrt{s}.
$$

To obtain the natural frequencies, one can replace s with $j\omega$ and after simplification we have,

$$
1 + \cos\sqrt{L(\frac{r}{EI})^{1/4}\sqrt{\omega}}\cosh\sqrt{L(\frac{r}{EI})^{1/4}\sqrt{\omega}} \;=\; 0. \qquad (6.9.12)
$$

The magnitude of the transfer function versus frequency is shown in Fig. 6.17.

One can follow the exact same procedure to get an expression for the transfer function without ignoring passive damping by replacing the A matrix in Eq. (6.9.4) with,

$$
\begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & C_D I\frac{s}{EI} & \frac{1}{EI} & 0 \\
0 & 0 & 0 & 1 \\
-\rho s^2 - \gamma s & 0 & 0 & 0
\end{bmatrix}.
$$

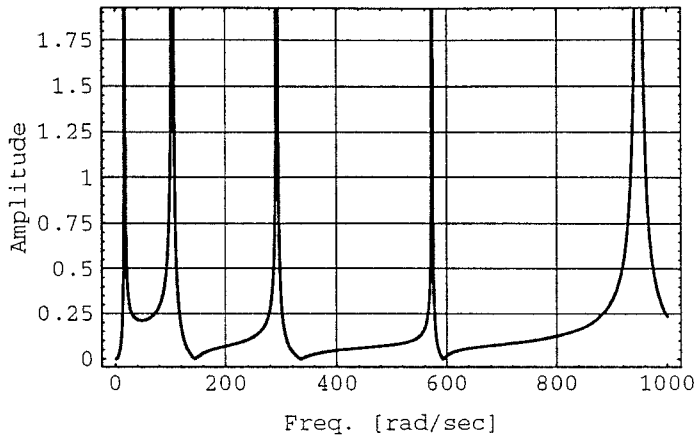

Figure 6.17: Modulus of the closed form transfer function vs. frequency

However the expression in that case would be very lengthy. Again it can be verified that the natural frequencies computed in Eq. (6.7.11) are actually roots of the denominator of the transfer function obtained (Eq. (6.9.12)) and also the peaks of the magnitude plot in Fig. (6.17) occurs almost at the same frequencies as in Fig. (6.11) which shows that the estimation method for computing the empirical transfer function is acceptable. Of course these two graphs are different because here for the sake of simplicity passive damping has not been included.

Chapter 7

Conclusion and Suggestions for Future Work

In this work we consider the problem of active vibration damping in flexible structures using piezo ceramics and some control strategies are tested using a simulation program written for this purpose.

Specifically, the application of a Gutman-Hagander type controller is shown to have both the advantage of stability under the assumption that there is a limit on the input signals that can be applied to the system and a faster settling time compared to the conventional LQR controller design.

Although the performance of these control methods has been tested through simulation programs, the actual implementation of these control designs into a cantilever beam is an interesting project that has not been done in the presented work.

In the last chapter, we developed an algorithm for nonparametric system identification and as a particular application we considered the problem of fitting a transfer function with piezo ceramics acting both, as sensors and actuators.

Through the usage of this method, one can obtain the transfer function (an

approximation) from each input to each output and then apply a MIMO control strategy to do the task of vibration damping. However, the controller design problem remains as a topic for further research and is not addressed in this thesis.

Also, once the controller is designed, the performance of it may be tested by using either the simulation program written for this purpose or by the actual implementation of it in an experiment.

Throughout this work, we did not consider any kind of distributed parameter control method for solving the problem at hand (which may seem to be more natural) and this again may be a challenging research problem.

For a general treatment of the distributed parameter control one may see the following references : [16], [7], [12], and, [10].

Bibliography

[1] H. T. Banks, Y. Wang, D. J. Inman, and J. C. Slater. Variable Coefficient Distributed Parameter System Models For Structures With Piezo Ceramic Actuators and Sensors. *Proceedings of the 31st Conference of Decision and Control*, Dec. 1992.

[2] R. D. Richtmyer, and K. W. Morton. *Difference Methods for Initial Value Problems.* Wiley-Interscience, 1967.

[3] Apex Microtechnology Corporation. Apex Corporation Technical Seminar. , 1991.

[4] E. F. Crawley and E. H. Anderson. Detailed Models of Piezoceramic Actuation of Beams. *J. of Intell. Mater. Syst. and Struct.*, Jan. 1990.

[5] E. P. Popov. *Mechanics of Materials.* Prentice Hall, 1978.

[6] Hewlett Packard Company. HP 35665A Dynamic Signal Analyzer Concept Guide. , September 1991.

[7] J. Bontsema, R. F. Curtain and J. M. Schumacher . Robust Control of Flexible Structures: A case Study. *Automatica*, pages 177–186, 1988.

[8] J. Dosch, D. J. Inman, and E. Garcia . A Self-sensing piezoelectric actuator for collocated control. *J. Intell. Material Systems and Structures*, pages 166–185, 1992.

[9] L. Ljung . *System Identification, Theory for the User.* Prentice-Hall, 1987.

[10] M. J. Chen and C. A. Desoer . Necessary and sufficient condition for robust stability of linear distributed feedback systems. *Int. J. Control*, pages 255–267, 1982.

[11] M. J. Balas. Modal Control of Certain Flexible Dynamic Systems. *SIAM J. Control and Optimization*, May 1978.

[12] P. P. Khargonekar and K. Poolla. Robust stabilization of Distributed Systems. *Automatica*, pages 77–84, 1986.

[13] P. Gutman and Per Hagander. A New Design of Constrained Controllers for Linear System. *IEEE Transaction on Automatic Cont.* , Jan. 1985.

[14] P. Stavroulakis. *Distributed System Theory: Part 1, Control.* Hutchinson Ross Publishing Co., Stroudsburg, Pa., 1983.

[15] Y. C. Pati. *Wavelets and Time-Frequency Methods in Linear Systems and Neural Networks.* PhD thesis, University of Maryland College Park, College Park, MD, 1992.

[16] R. F. Curtain and K. Glover . Robust stabilization of infinite dimensional systems by finite dimensional controllers. *Systems and Control Letters*, pages 41–47, Feb. 1986.

[17] R. Spangler, Jr. and S. R. Hall. Piezoelectric Actuation for Helicopter Rotor Control. *AIAA Journal*, pages 1–11, 1990.

[18] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. Preprint, 1992.

[19] S. G. Tzafestas. *Distributed Parameter Control Systems:Theory and Applications.* Pergamon Press, Elmsford, N.Y., 1982.

[20] T. Bailey and E. Hubbard Jr. Distributed Piezoelectric-Polymer Active Control of a Cantilever Beam. *J. of Guidance* , Sept.-Oct. 1985.

[21] Y. Yang, H.J. Sussmann, and E. D. Sontag. Stabilization of Linear Systems with Bounded Controls. *Proceedings of Nonlinear Control System Design Symposium*, pages 15–20, June 1992.