2002-11-20

Implementation of an Actuator Placement, Switching Algorithm for Active Vibration Control in Flexible Structures

Murali Murugavel Manjakkattuvalasu Swathanthira Kumar

Worcester Polytechnic Institute

Follow this and additional works at: https://digitalcommons.wpi.edu/etd-theses

Repository Citation
Swathanthira Kumar, Murali Murugavel Manjakkattuvalasu, "Implementation of an Actuator Placement, Switching Algorithm for Active Vibration Control in Flexible Structures" (2002). Masters Theses (All Theses, All Years). 1091.
https://digitalcommons.wpi.edu/etd-theses/1091

This thesis is brought to you for free and open access by Digital WPI. It has been accepted for inclusion in Masters Theses (All Theses, All Years) by an authorized administrator of Digital WPI. For more information, please contact wpi-etd@wpi.edu.
Implementation of an Actuator Placement, Switching Algorithm for Active Vibration Control in Flexible Structures

By

M S Murali Murugavel

A Thesis

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Master of Science

In

Mechanical Engineering

by

_________________________

September 2002

APPROVED:

____________________________
Dr. Michael A. Demetriou, Advisor

____________________________
Dr. Mark W. Richman, Committee Member

____________________________
Dr. Zhikun Hou, Committee Member

____________________________
Dr. Nikolaos A. Gatsonis, Graduate Committee Representative
Abstract

The recent years have seen the innovative system integration of a great many actuator technologies, such as point force actuators for space vehicle applications and the use of single fire actuators; such as pyrocharges to guide a free falling bomb to it’s target. The inherent limitations of these developments, such as nonlinear behavior under extreme environments and/or prolonged/repeated usage leading to a relaxation time component between firing of actuators and inherent system power limitations, have resulted in greater need for sophisticated control algorithms that allow for optimal switching between various actuators in any given embedded configuration so as to achieve the best possible performance of the system. The objective of this investigation is to offer a proof of concept experimental verification of a real time control algorithm, which switches between online piezoelectric actuators, employed for vibration control in an aluminum beam with fixed boundary conditions. In this investigation at a given interval of time, only one actuator is activated and the rest are kept dormant. The reason is to demonstrate the better vibration alleviation characteristics realized in switching between actuators depending on the state of the system, over the use of a single actuator that is always in fire mode. This effect is particularly pronounced in controlling systems affected by spatiotemporal disturbances. The algorithm can be easily adapted for various design configurations or system requirements. The optimality of switching is with respect to the minimal cost of an LQR performance index that corresponds to each actuator. Computer simulations with repeatable disturbance profiles, revealed that this algorithm offered better performance over the non-switched case. Performance measures employed were
the time varying total energy norm of the dynamic system and position traces at any particular location on the beam. This algorithm was incorporated on a dSPACE rapid prototyping platform along with suitable hardware. Experimental and simulation results are discussed.
Acknowledgement

I would like to dedicate this thesis to my Parents, my Sister and Ganapathi Raman Balasubramanian for having provided me with the basic belief system on which everything has been founded. A special word of thanks to all my well wishers and former teachers, whose wishes worked their inherent goodness.

I would like to record my sincere thanks and appreciation to my advisor Professor Michael A. Demetriou for his invaluable time, guidance and effort over the last two years.

I would also like to take this opportunity to record my sincere appreciation of the time that many engineers and technicians from various firms across the country spent, sharing with me their experience and in the process taught me ‘Systems Engineering’ as practiced in America.

I would also like to place on record the contributions by the following people, who helped directly towards this project,

1. Worcester Polytechnic Institute’s trustees and everyone affiliated for keeping alive the vision of its founders.

2. All the Professors who taught me various courses at WPI; Professors Sullivan, Demetriou, Hou, Pryputneiwicz, Richman, Dimentberg, Rencis and Fehribach.

3. Professors Olinger, Gatsonis, Durgin for their support during my stay.

4. Professors Grandin, Savilonis, Furlong and Weber with whom I was associated with during my stay here.

5. The decision makers at UTC and at the NSF for their indirect financial support.

6. Professor Hall for providing me with the initial loan of equipment for my work.
7. Professor Blandino for his help with the vacuum pump and in choosing hardware.

8. Radhakrishnan for his time and effort in fabricating the beam holding mechanism.

9. Wei and Vikram for their time over the last two years.

10. Tom for his help in soldering the wires to the disturbance patches.

11. Mr. Johnston, Mr. Hefti, Mr. Taylor, Mr. Billings, Mr. Tom (EE), Mr. James (EE)
    and Mr. Derosier for their invaluable time and effort during the last two years.

12. Mr. Najafi and Mr. Korzeniowski for their systems support.

13. Ms. Edilberti, Ms. Furhman, Ms. Dresser and Ms. St.Louis for their ever present
    support and assistance in every possible manner.

14. Mr. Joe, Mr. Larry and the rest of the maintenance crew for their assistance over
    the last two years.

15. Mr. Thomsen and Ms. Martin of the International house for their support.

16. Krishna, Praveen, Jayant, Vinay, Toby, Vitaly, Anoop, Dr.Chen and Tilak for
    their time and support.

17. Visiting Professor Carlos for his time and advise.

    It is possible that I might have inadvertently forgotten to mention someone’s
    contribution, but my sincere thanks goes without mention to anyone whose true effort is
    behind this project.
List of Figures, Tables

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Generalized structural vibration control system architecture</td>
<td>02</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Beam in transverse vibration and a free body diagram of a small element</td>
<td>09</td>
</tr>
<tr>
<td>Figure 2a</td>
<td>ACX QP20N piezoelectric actuator</td>
<td>22</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Plot of possible actuator locations versus corresponding $J_{opt} = tr[P(x)]$</td>
<td>24</td>
</tr>
<tr>
<td>Figure 4a</td>
<td>Identification of the control patches for the computer simulation</td>
<td>24</td>
</tr>
<tr>
<td>Figure 4b</td>
<td>Identification of the disturbance patches for the computer simulation</td>
<td>25</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Voltage signal applied to disturbance ‘Patch 20’, ‘Patch 38’ in this simulation</td>
<td>27</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Closed loop energy norm for switching plus feedback control (solid) with ‘Patch 03’, ‘Patch 10’, ‘Patch 48’ on call, feedback control plus single actuator (dash-dot) Patch 10, and the open loop case (dotted)…</td>
<td>28</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Displacement trace at 0.6667 m from one end of beam. Switching plus feedback control (solid) with Patch 03, Patch 10, Patch 48 on call, feedback control plus single actuator (dash-dot) Patch 10, and the open loop case (dotted)…</td>
<td>29</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Displacement trace at location 0.6667 m from one end of the beam as given by the simulated plant (solid blue) and that estimated by the observer (red dotted). This plot is to demonstrate the efficiency of the observer……………………………………………………………………………………………………</td>
<td>29</td>
</tr>
<tr>
<td>Figure 9</td>
<td>The switching algorithm calls for different controllers at different instants of time. During the first 8 seconds, the controllers are not engaged……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………</td>
<td>30</td>
</tr>
<tr>
<td>Figure 10</td>
<td>Front view of the hardware implementation</td>
<td>32</td>
</tr>
<tr>
<td>Figure 11</td>
<td>View of the disturbance patches and the sensor at 0.6667 m from left end</td>
<td>33</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Rapid Control Prototyping system at The Controls Laboratory, WPI...</td>
<td>34</td>
</tr>
</tbody>
</table>
Table 1 Properties associated with the simulation.............................. 23
Table 2 Patch identification.......................................................... 23
Table 3 Comparison of the first four fundamental frequencies.............. 26
# Table of Contents

Chapter 1. Introduction ............................................... 01  
  1.1 Literature Review......................................................... 03  
  1.2 Research Objective and Generic Design Approach............... 05  
  1.3 Uniqueness and Layout of Current Work.......................... 07  

Chapter 2. Mathematical Modeling .................................. 09  
  2.1 Bernoulli-Euler Beam Equation ....................................... 09  
  2.2 Piezoelectric Patch Contribution and Galerkin based Approximation... 11  
  2.3 Optimal Actuator Placement Algorithm ............................ 16  
  2.4 Actuator Activation Policy............................................. 18  
  2.5 The Switching Algorithm.............................................. 20  

Chapter 3 Numerical Simulation, Results......................... 22  

Chapter 4 Experiment Design, Results ............................ 31  
  4.1 Hardware Implementation.............................................. 31  
  4.2 Rapid Control Prototyping on dSPACE ACE 1103 Board........... 33  
  4.3 Real-Time Interface to Simulink..................................... 35  
  4.4 Experimental Results.................................................. 38  

Chapter 5 Conclusions and Recommendations................... 40  

Appendix ........................................................................... 42  

References ........................................................................ 58
Nomenclature

\( A \) : Beam rectangular cross sectional area

\( h_y, b \) : Width of the beam

\( h_z, t \) : Thickness of the beam

\( l \) : Length of the beam

\( \omega \) : Transverse displacement of beam

\( E \) : Young’s modulus of the beam

\( I \) : Moment of Inertia of the beam

\( M \) : Bending moment along the beam

\( V \) : Shear force along the beam

\( \rho \) : Density of the beam

\( E_a \) : Young’s modulus of the piezoelectric actuator

\( K_a \) : Piezoelectric constant

\( d_{31} \) : Piezoelectric charge coefficient

\( t_a \) : Thickness of the piezoelectric actuator

\( u(t) \) : Applied voltage to piezoelectric actuator

\( \chi \) : Binary value indicating presence of piezoelectric actuator

\( x \) : Spatial variable along length of beam

\( \varepsilon \) : Length of piezoelectric actuator

\( M \) : Mass matrix

\( K \) : Stiffness matrix

\( B_a(x) \) : \( x \) Parameterized input matrix
\[ D \quad : \quad \text{Damping matrix} \]

\[ \alpha_1 \quad : \quad \text{Kevin-Voigt damping coefficient} \]

\[ \alpha_2 \quad : \quad \text{Air damping coefficient} \]

\[ z \quad : \quad \text{State vector of the vibrating beam} \]

\[ C \quad : \quad \text{Observation matrix} \]
Chapter 1 Introduction

The Engineering world is witnessing the beginnings of the ‘Smart Materials age’, which will bring to the forefront specialists in Biotechnology, Nano-technology, Artificial Intelligence, Neural networks, Sensors, Controls and Material sciences.

A smart structure consists of a structure provided with a set of actuators and sensors coupled by a controller; if the bandwidth of the controller includes some vibration nodes of the structure, its dynamic response must be considered. If the set of actuators and sensors are located at discrete points of the structure, they can be treated separately. The distinctive feature of smart structures is that the actuators and sensors are often distributed and have a high degree of integration inside the structure, which makes separate modeling impossible. Moreover, in some applications like vibroacoustics, the behavior of the structure itself is highly coupled with the surrounding medium; this also requires coupled modeling.

From a mechanical point of view, classical structural materials are entirely described by their elastic constants relating stress and strain, and their thermal expansion coefficient relating the strain to the temperature. Smart materials are materials where strain can also be generated by different mechanisms involving temperature, electric field or magnetic field, etc., as a result of some coupling in their constitutive equations.

Piezoelectric materials are used in this work, which have a recoverable strain of 0.1% under electric field; they can be used as actuators as well as sensors. There are two broad classes of piezoelectric materials used in vibration control: ceramics and polymers.
The piezo polymers are used mostly as sensors, because they require extremely high voltages and have limited control authority; the best known is the polyvinylidene fluoride (PVF2). Piezoceramics are used extensively as actuators and sensors, for a wide range of frequency including ultrasonic applications; they are well suited for high precision in the nanometer range. The best-known piezoceramic is the Lead Zirconate Titanate (PZT).
Some of the most interesting applications of this development are in the area of vibration control in large flexible space structures and in alleviating buffet load strains in fighter aircraft. Figure 1 (from www.acx.com) also shows the generalized structural vibration control architecture, which was implemented at ‘The Controls Laboratory’, Worcester Polytechnic Institute during the course of this investigation.

1.1 Literature Review

The initial application of piezoelectric materials as actuators involved the vibration control of beams that were done first by Bailey and Hubbard [3] and Crawley and De Luis [13]. Their research lead to better understanding of vibration control of beam with piezoelectric actuators and thereby the development of the simplified models for beams. Advances in theoretical models led to formulation of two theories. The theories are 1) piezoelectric plate theory by Lee[35], Wang and Rogers [54] and 2) piezoelectric thin shell theory by Tzou and Gadre [30]. Both these theories were based on the assumptions of classical plate and shell theory, respectively.

Subsequent development in the formulation of theoretical models led to a variety of finite element formulations for beams, plate and shells. Beam formulation has been done by Robins and Reddy [44], Shieh [48]. Finite element plate formulation has been done by Chandrashekara and Agrawal [8], Hwang and Park [29]. Lammering [32], Tzou and Ye [52] have done finite element shell formulation.

Development in the area of material research aims at incorporating intelligence into engineering materials, enabling them to sense the external stimuli and alter their own
properties to adapt to the changes in the environment. Iyer and Haddad [30] discussed possible terms of intelligence that may be incorporated in these materials. Three basic terms of intelligent materials namely: sensor, processor, and actuator functions are described. Recently Matsuzaki [56] presented an overview of the smart structure research activities in Japan which were reported after 1992 till 1996 including a brief description of the recent situation in the research circle. The author mentioned about vibration, shape and motion controls of space structures, shape memory alloys, design approaches.

A finite element formulation for piezoelectric vibration has been given by Allik and Huges [1]. A consistent plate model of induced strain actuator system which combines both actuators and substrates into one integrated unit was developed by Crawley and Lazarus [12]. The models and solutions developed were experimentaly verified. Static and dynamic models were derived for piezoelectric actuators by Crawley and Luis [13]. The authors performed scaling analysis and evaluated various piezoelectric materials based on their effectiveness in transmitting strain to the substructure. A one-dimensional beam formulation based on layer-wise theory was developed using finite element method by Donthireddy and Chandrashekara [17]. Various parametric studies to demonstrate the influence of boundary condition ply orientation on change in shape of laminated beam with piezoelectric actuation have been done.

The buckling control of column strips using piezoceramic actuators in conjunction with a closed loop control system was done by Thompson and Loughlan [51]. A finite element procedure to compute the electromechanical response is proposed by Gaudenzi [20]. A new beam element including adhesive and actuators has been developed by Lin
et.al [36]. Different models for laminated composite plates with piezoelectric actuators and sensors were developed by Sosa and Castro [50], Seeley and Chattopadyay [9]. Higher order shear deformation theory is employed in the problem.

Recently thermo-piezoelectric analysis for isotropic as well as composite beams have been presented [25], [53]. Lee and Saravans [34] presented a generalized finite element formulation of composite plates in thermal environment by employing a bilinear 4-noded element. The numerical studies have indicated the significance of thermal effect on the performance of piezoelectric substructures. Only limited research has been done in this area. A number of papers [43,7,23,4] dealing with the recent advances and related research work in the field of piezoelectric sensors and actuation have been listed in the reference.

Piezoelectric actuators and sensors have been used in many vibration control applications of flexible structures [11,16,19,39,40,41]. While this discussion provides the backdrop for the current work, the relevant algorithms and methods used in this investigation are referenced through Chapter 2 in the course of developing the associated mathematical models.

1.2 Research Objective and Generic Design Approach

The objective of this investigation was to adapt and implement the real time algorithm proposed by Demetriou [15]. This unique algorithm proposes a means to identify and engage a particular actuator from among an array of actuators on a flexible structure. This algorithm also addresses the latent requirement associated with the
developments in actuator technologies such as point force thrusters for space vehicle applications (since these systems have a time to recharge component and cannot be used continuously) and in innovative use of single fire actuators, such as pyrocharges to guide a ‘Smart’ bomb to it’s target. The proposed algorithm accounts for the inherent limitations of such developments, such as relaxation time between firing actuators and system power limitations, by optimally switching between various actuators in any given configuration so as to achieve the best possible performance.

The design procedure begins with the understanding of the mechanical system, the required performance objectives and a specification of the disturbances applied to it; the controller cannot be designed without some knowledge of the disturbance applied to the system. If the frequency distribution of the energy of the disturbance is known, the open loop performances can be evaluated and the need for an active control system can be assessed. If an active system is required, its bandwidth can be roughly specified. The next step consists of selecting the proper type and location of sensors to monitor the behavior of the system, and actuators to interfere with the states of the system. Once the actuators and sensors have been selected, a model of the structure is developed, usually with finite elements; it can be improved by identification if experimental transfer functions are available. Such models generally involve too many degrees of freedom to be directly useful for only a few degrees of freedom, usually the vibration modes of the system, which carry the most important information about the system behavior. At this point, if the actuators and sensors can be considered as perfect (in the frequency band of interest), they can be ignored from the model; their effect on the control system performance will be tested after the design has been completed. If, on the contrary, the
dynamics of the actuators and sensors may significantly affect the behavior of the system, they must be included in the model before the controller design. Even though most controllers are implemented in a digital manner, nowadays, there are good reasons to carry out a continuous design and transform the continuous controller into a digital one with an appropriate technique. This approach works well when the sampling frequency is two orders of magnitude higher than the bandwidth of the control system, as is generally the case in structural control.

1.3 Uniqueness and Layout of Current Work

The present work is unique in the sense it implements a real time switching policy algorithm for actuators, which offers better performance in control of vibrations over the use of a single control actuator that is permanently engaged. This is achieved by determining at the end of a predetermined time step (in built into the controller) which of the available control actuators is better suited to control the vibrations during the time interval, based on the state of the system at that time instant. The following are important milestones crossed in the course of this work,

1. A general purpose FEA model for beams with piezoelectric actuators, which also allows implementation of different boundary conditions, was developed.
2. An algorithm, which determines the best control actuator locations in a vibrating member, was incorporated in the code.
3. Implemented an Active switching controller algorithm for controlling vibrations in structures. The simulations confirm that such an algorithm offers better performance in controlling vibrations than in a non-switched case.

4. This algorithm basically addresses the spatio-temporal variations of disturbances, i.e., enhances performance by improving robustness with respect to time and space varying disturbances. It also robust with respect to any unmodelled dynamics. The uniqueness lies in the idea that the design of a robust controller requires some idea of the bandwidth of the disturbance, whereas the proposed switching algorithm simply calls whichever actuator is better suited to address the disturbance.

In Chapter 2, a mathematical model of the beam using a Galerkin scheme is developed. An algorithm for optimal placement of actuators and an actuator switching policy is developed. The controller is designed using Linear Quadratic Regulator (LQR) techniques. Numerical simulation results are discussed in Chapter 3. Chapter 4 elaborates on the experimental procedure and design of the experimental setup. The online switching policy is implemented on hardware and results discussed. Conclusions and recommendations for future work are presented in Chapter 5.
Chapter 2 Mathematical Modeling

2.1 Bernoulli-Euler Beam Equation

![Beam in transverse vibration and a free body diagram of a small element.](image)

**Figure 2:** Beam in transverse vibration and a free body diagram of a small element.

This section primarily develops the basic transverse vibration or flexural vibration equations of a beam. Figure 2 illustrates a cantilevered beam with the transverse direction of vibration indicated [i.e., the deflection, $w(x,t)$, is in the y direction]. The beam is of rectangular cross section $A(x)$ with width $h_y$, thickness $h_z$, and length $l$. Also associated with the beam is a flexural (bending) stiffness $EI(x)$, where $E$ is the Young’s elastic modulus for the beam and $I(x)$ is the cross-sectional area moment of inertia. From
mechanics of materials, the beam sustains a bending moment \( M(x,t) \) which is related to the beam deflection, or bending deformation, \( \omega(x,t) \), by

\[
M(x,t) = EI(x) \frac{\partial^2 w(x,t)}{dx^2}.
\]  

(2.1)

A model of bending vibration may be derived from examining the force diagram of an infinitesimal element of the beam as indicated in Figure 2. Assuming the deformation to be small enough such that the shear deformation is much smaller than \( w(x,t) \) (i.e., so that the sides of the element \( dx \) do not bend), a summation of forces in the \( y \) direction yields

\[
\left(V(x,t) + \frac{\partial V(x,t)}{\partial x} \right) dx - V(x,t) + f(x,t) dx = \rho A(x) dx \frac{\partial^2 w(x,t)}{dt^2}.
\]  

(2.2)

Here \( V(x,t) \) is the shear force at the left end of the element \( dx \), \( V(x,t) + V_s(x,t) dx \) is the shear force at the right end of the element \( dx \), \( f(x,t) \) is the total external force applied to the element per unit length and the term on the right side of the equality is the inertial force of the element. The assumption of small shear deformation used in the force balance of equation (2.2) is true if \( l/h_z \geq 10 \) and \( l/h_y \geq 10 \) (i.e., for long slender beams).

Next the moments acting on the element \( dx \) about the \( z \) axis through point \( Q \) are summed. This yields

\[
\left[ M(x,t) + \frac{\partial M(x,t)}{\partial x} dx \right] - M(x,t) + \left[ V(x,t) + \frac{\partial V(x,t)}{\partial x} dx \right] dx + \left[ f(x,t) dx \right] \frac{dx}{2} = 0.
\]  

(2.3)

Here the left-hand side of the equation is zero since it is also assumed that the rotary inertia of the element \( dx \) is negligible. Simplifying this expression yields

\[
\left[ \frac{\partial M(x,t)}{\partial x} + V(x,t) \right] dx + \left[ \frac{\partial V(x,t)}{\partial x} + \frac{f(x,t)}{2} \right] (dx)^2 = 0.
\]  

(2.4)
Since \( dx \) is assumed to be very small, \( (dx)^3 \) is assumed to be almost zero, so that this moment expression yields (\( dx \) is small, but not zero)

\[
V(x,t) = -\frac{\partial M(x,t)}{\partial x}.
\]  

(2.5)

This states that the shear force is proportional to the spatial change in the bending moment. Substitution of this expression for the shear force into equation (2.2) yields

\[
-\frac{\partial^2}{\partial x^2}[M(x,t)]dx + f(x,t)dx = \rho A(x)dx \frac{\partial^2 w(x,t)}{\partial t^2}.
\]

(2.6)

Further substitution of equation (2.1) into (2.6) and dividing by \( dx \) yields

\[
\rho A(x) \frac{\partial^2 w(x,t)}{\partial t^2} + \frac{\partial^2}{\partial x^2} \left[ EI(x) \frac{\partial^2 w(x,t)}{\partial x^2} \right] = f(x,t).
\]

(2.7)

The assumptions used in formulating this model are that the beam be

1. Uniform along its span, or length, and slender.
2. Composed of a linear, homogeneous, isotropic, elastic material without axial loads.
3. Such that plane sections remain plane.
4. Such that the plane of symmetry of the beam is also the plane of vibration so that rotation and translation are decoupled.
5. Such that rotary inertia and shear deformation can be neglected.

### 2.2 Piezoelectric Patch Contribution and Galerkin Based Approximation

The equation that describes the transverse vibration dynamics of an isotropic beam of uniform cross sectional area with bonded piezoelectric actuators and fixed at both ends, as taken from Banks et al. [5], using (2.7) is given by,
\[
\rho A \frac{\partial^2 w(x,t)}{\partial t^2} + EI \frac{\partial^4 w(x,t)}{\partial x^4} + \alpha_1 \frac{\partial w(x,t)}{\partial t} + \alpha_2 \frac{\partial^5 w(x,t)}{\partial x^4 \partial t} = \frac{\partial^2}{\partial x^2} \left( K_a \chi(x) u(t) \right) 
\]  
(2.8)

where \( K_a \) is the piezoceramic constant, as taken from Moheimani [41] and given by

\[
K_a = (1/2)E_a d_{31} b(t_a + t_b). 
\]

The constant \( 1/2 \) indicating that actuators are not in collocated configuration, i.e. they do not have a counterpart on the other side of the beam.

\( \chi(x) = 1 \) if a piezoelectric actuator is present in that location, else \( \chi(x) = 0 \). \( \alpha_1, \alpha_2 \) are the air damping and Kevin-Voigt damping terms respectively.

Integrating along the length of the beam equation (2.8), after multiplying both sides of the equation by a test function \( \phi(x) \), which satisfies the applied boundary conditions,

\[
\int_0^I \rho A \frac{\partial^2 w(x,t)}{\partial t^2} \phi(x) dx + \int_0^I EI \frac{\partial^4 w(x,t)}{\partial x^4} \phi(x) dx + \int_0^I C_a \frac{\partial w(x,t)}{\partial t} \phi(x) dx 
\]

\[
+ \int_0^I C_d \frac{\partial^5 w(x,t)}{\partial x^4 \partial t} \phi(x) dx = \int_0^I \frac{\partial^2}{\partial x^2} \left( K_a \chi(x) u(t) \right) \phi(x) dx. 
\]

Using a Galerkin-based approximation as outlined in [40] with cubic polynomials on the interval \([0,l]\) having uniform partition \( \{ \frac{il}{n} \}_{i=0}^n \), the beam displacement \( w(t,x) \) is approximated by,

\[
w^n(t,x) = \sum_{i=1}^\infty w^n_i(t) \phi_i(x),
\]
with, \( w(0, t) = \frac{\partial w}{\partial x}(0, t) = 0 \), \( w(t, l) = \frac{\partial w}{\partial x}(t, l) = 0 \); i.e., satisfying the applied boundary conditions. For \( n = 1, 2, \ldots \), let \( \{\phi_j\}_{j=1}^{n+1} \) be the standard cubic B-splines [42] on the interval \([0, l]\) with respect to the uniform mesh \( \left\{0, \frac{l}{n}, \frac{2l}{n}, \ldots, l\right\} \),

\[
\begin{align*}
\phi_j''(x) &= \begin{cases}
\left(\frac{nx}{l} + 2 - i\right)^3, & x \in \left[\frac{(i-2)l}{n}, \frac{(i-1)l}{n}\right] \\
1 + 3\left(\frac{nx}{l} + 1 - i\right)\left(1 + \left(\frac{nx}{l} + 1 - i\right)\left(i - \frac{nx}{l}\right)\right), & x \in \left[\frac{(i-1)l}{n}, \frac{il}{n}\right] \\
1 + 3\left(1 + i - \frac{nx}{l}\right)\left(1 + \left(1 + i - \frac{nx}{l}\right)\left\{\frac{nx}{l} - i\right\}\right), & x \in \left[\frac{il}{n}, \frac{(i+1)l}{n}\right] \\
\left(2 + i - \frac{nx}{l}\right)^3, & x \in \left[\frac{(i+1)l}{n}, \frac{(i+2)l}{n}\right] \\
0, & \text{otherwise on } [0, l].
\end{cases}
\end{align*}
\] (2.10)

To simulate the plant, we modify the cubic splines to account for the essential boundary conditions at the fixed ends (i.e. at \( x = 0, x = l \)). The \( (n-1) \) splines \( \{\phi_j\}_{j=1}^{n-1} \), are related to the original \( (n+3) \) splines via the transformation,

\[
\{\phi_1, \phi_2, \ldots, \phi_{n-1}\} = \{-2\phi_{-1} + \phi_0 - 2\phi_1, \phi_2, \ldots, \phi_{n-3}, \phi_{n-2}, -2\phi_{n-1} + \phi_n - 2\phi_{n+1}\}. \tag{2.11}
\]

This is implemented by creating a Transformation matrix \( T \) of dimension \((n-1) \times (n+3)\), which captures the applied boundary condition. For the present case of fixed-fixed boundary conditions (for the case of \( n = 5 \),

13
If this procedure is to be used for a cantilever beam, then the Transformation matrix $T$ has the dimension $(n+1) \times (n+3)$ and for the case of $n=5$, is given by,

$$
T = \begin{bmatrix}
-2 & 1 & -2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -2 & 1 & -2
\end{bmatrix}.
$$

When we substitute $w^n(x,t) = \sum_{i=1}^{n-1} w_i^n(t) \phi_i(x)$ in equation (2.9) we have the following weak formulation,

$$
\sum_{i=1}^{n-1} \rho A \frac{\partial^2 w_i(t)}{\partial t^2} \int_0^l \phi_i(x) \phi(x) dx + \sum_{i=1}^{n-1} EI w_i(t) \int_0^l \frac{\partial^4 \phi_i(x)}{\partial x^4} \phi(x) dx + \sum_{i=1}^{n-1} \alpha_i \frac{\partial w_i(t)}{\partial t} \int_0^l \phi_i(x) \phi(x) dx
$$

$$
+ \sum_{i=1}^{n-1} \alpha_2 \frac{\partial w_i(t)}{\partial t} \int_0^l \frac{\partial^2 \phi_i(x)}{\partial x^2} \phi(x) dx = \int_0^l \frac{\partial^2}{\partial x^2} \left( K_s \chi(x) u(t) \right) \phi(x) dx
$$

Choose $\phi(x)$ (test function) to satisfy the applied boundary conditions. Choose it as $\phi_i(x)$.

Now setting $\phi(x) = \phi_j(x)$, we have,

$$
\sum_{i=1}^{n-1} \rho A \frac{\partial^2 w_i(t)}{\partial t^2} \int_0^l \phi_i(x) \phi_j(x) dx + \sum_{i=1}^{n-1} EI w_i(t) \int_0^l \frac{\partial^2 \phi_i(x)}{\partial x^2} \frac{\partial^2 \phi_j(x)}{\partial x^2} dx
$$

$$
+ \sum_{i=1}^{n-1} C_a \frac{\partial w_i(t)}{\partial t} \int_0^l \phi_i(x) \phi_j(x) dx
$$

$$
+ \sum_{i=1}^{n-1} C_a \frac{\partial w_i(t)}{\partial t} \int_0^l \frac{\partial^2 \phi_i(x)}{\partial x^2} \frac{\partial^2 \phi_j(x)}{\partial x^2} dx = \int_0^l \frac{\partial^2}{\partial x^2} \left( K_s \chi(x) u(t) \right) \phi_j(x) dx
$$
Equation (2.13) is represented by the following \((n-1)\) dimensional second order vector system,

\[
\mathbf{M}\ddot{\mathbf{w}}(t) + \mathbf{D}\dot{\mathbf{w}}(t) + \mathbf{K}\mathbf{w}(t) = \mathbf{B}_o\mathbf{u}(t),
\]

(2.14)

where the matrices \(\mathbf{M}\), \(\mathbf{K}\) and \(\mathbf{B}_o\) are given by (for \(i, j = 1, 2, \ldots, n-1\)),

\[
\mathbf{M}_{ij} = \rho A \int_0^l \phi_i(x) \phi_j(x) \, dx + \rho_a A \int_0^l \phi_i(x) \phi_j(x) \chi(x_i) \, dx, 
\]

\[
\mathbf{K}_{ij} = EI \int_0^l \frac{\partial^2 \phi_i(x)}{\partial x^2} \frac{\partial^2 \phi_j(x)}{\partial x^2} \, dx + \frac{1}{3} E_a a_h \int_0^l \frac{\partial^2 \phi_i(x)}{\partial x^2} \frac{\partial^2 \phi_j(x)}{\partial x^2} \chi(x_i) \, dx, 
\]

\[
\mathbf{B}_{0j} = K_a \int_0^l \chi(x_i) \frac{\partial^2 \phi_j(x)}{\partial x^2} \, dx = K_a \int_{x_0}^{x_0+\varepsilon} \frac{\partial^2 \phi_j(x)}{\partial x^2} \chi(x_i) \, dx = K_a \left[ \frac{\partial}{\partial x} \phi_j(x_0 + \varepsilon) - \frac{\partial}{\partial x} \phi_j(x_0) \right].
\]

The mass and the stiffness of the actuators are incorporated into the model (as shown above) by the use of the terms \(\rho A \int_0^l \phi_i(x) \phi_j(x) \chi(x_i) \, dx\) and

\[
\frac{1}{3} E_a a_h \int_0^l \frac{\partial^2 \phi_i(x)}{\partial x^2} \frac{\partial^2 \phi_j(x)}{\partial x^2} \chi(x_i) \, dx,
\]

wherein \(\rho_a\) is the mass/unit length of the actuator and \(a_a\) is given by \(\left( \frac{h_y}{2} + t_a \right)^3 - \left( \frac{h_y}{8} \right)^3\). See [5] for a discussion on these terms. Damping has been introduced into the system by the addition of a damping matrix \(\mathbf{D} = \mathbf{D}^T\) of dimension \((n-1) \times (n-1)\) with the form \(\mathbf{D} = \alpha_1 \mathbf{K} + \alpha_2 \mathbf{M}\). \(\mathbf{B}_o(x)\) is the \((n-1) \times q\) parameterized input matrix. The dependence of the input matrix on the spatial variable is written explicitly to emphasize the dependence of the actuating device (in this case the piezoelectric patch of length \(\varepsilon\), which is affixed at locations \(x_0\)) on the actuator location.

Using Meirovitch [43], equation (2.14) can be placed in the following first order vector form.
\[
\frac{d}{dt} \begin{bmatrix} w(t) \\
\dot{w}(t) \end{bmatrix} = \begin{bmatrix} 0 & I \\
-M^{-1}K & -M^{-1}D \end{bmatrix} \begin{bmatrix} w(t) \\
\dot{w}(t) \end{bmatrix} + \begin{bmatrix} 0 \\
-M^{-1}B_{0}(x) \end{bmatrix} u(t) 
\] (2.15)

Equation (2.15) in a more compact form is given by,
\[
\dot{z}(t) = Az(t) + B(x)u(t), \quad z = \begin{bmatrix} w \\
\dot{w} \end{bmatrix} 
\] (2.16)

along with the expression for the measured output,
\[
y(t) = C(x)z(t) 
\] (2.17)

where the observation matrix \( C(x) \) is also location dependent. Position or velocity (point) sensors are assumed to provide measurements of the beam displacement and / or velocity. The vector representation is given by
\[
C_{i}(x_{s}) = \int_{0}^{I} \delta(x - x_{s})\phi_{i}(x)dx = \phi_{i}(x_{s}), \quad i = 1,...,n-1, 
\] (2.18)

where \( \delta(x) \) denotes the spatial Dirac-Delta function and \( x_{s} \) denotes the location of the sensor. If a point velocity sensor (as in this experimental investigation) is used, then \( C_{i}(x_{s}) = [0,...,0,C_{1}(x_{s}),...,C_{n-1}(x_{s})] = [0,...,0,\phi_{1}(x_{s}),...,\phi_{n-1}(x_{s})] \).

2.3 Optimal Actuator Placement Algorithm

This work primarily deals only with the actuator placement algorithm as proposed by Demetriou [14] while taking advantage of a known result for optimal location of a single sensor. Sensor and actuator placement studies in the past mainly focused on the system’s observability and controllability indices [21]. The algorithms developed using this approach along with references on prior work can be found in the
books [49,22,31]. The present work is unique, in the idea that it is based on minimizing the optimal value of a proposed performance index as opposed to considering observability and controllability measures. Similar work along the same lines of the present investigation is found in [18] by considering only the problem of actuator placement in flexible structures.

For a fixed actuator location $x_i$, one can obtain the optimal control law $u(\cdot)$ that minimizes the $H^2$ cost functional

$$J(u) = \int_0^\infty \left\{ z^T(t)Qz(t) + u^T(t)Ru(t) \right\} dt \quad Q \geq 0, \ R > 0. \quad (2.19)$$

For the LQR ($H^2$) case, the optimal control is given by

$$u(t) = -R^{-1}B(x_i)Pz(t) \quad (2.20)$$

where $P$ solves the Algebraic Riccati Equation (ARE)

$$PA + A^TP + Q - PB(x_i)R^{-1}B(x_i)^TP = 0. \quad (2.21)$$

The optimal value of the cost functional is then given [14] by

$$J_{opt} = z^T(0)Pz(0). \quad (2.22)$$

Now, the optimal cost given by equation (2.22) is minimized with respect to the actuator locations $x_i, i = 1,2,\ldots,q$. The value of the described functional depends on both the actuator location and the initial condition $z(0)$. A way to remove this dependence on the initial conditions is to minimize the trace of the location parameterized solution $P(x)$ of the ARE (2.21), instead of the expression (2.22). This is developed further to give the following important result

$$J_{opt}(x_i) = tr[P(x_i)]. \quad (2.23)$$
where $P(x_i)$ solves the location parameterized ARE,

$$P(x_i)A + A^T P(x_i) + Q - P(x_i)B(x_i)R^{-1}B^T(x_i)P(x_i) = 0 \quad (2.24)$$

and $tr[\cdot]$, denotes the trace of a matrix, and is defined as the sum of all the diagonal elements of the matrix. The trace is also defined as the sum of the eigenvalues of the matrix [33]. This allows for choosing the actuator locations that are optimal in controlling vibrations from a possible list of locations by identifying that particular actuator or actuator bank which corresponds to the minimum value of the $J_{opt}(x_i)$. This scheme is employed in this work.

### 2.4 Actuator Activation Policy

Demetriou [15] further proposes a single actuator control and / or ‘fire’ policy in a real time scenario. The significance of the work lies in determining which actuator among the choice of $i = 1, 2, ..., q$ actuators located at distances $x_i$, $i = 1, 2, ..., q$ respectively, from one end of the flexible structure, to engage during a particular time interval based on the state of the system at that time. The single actuator to be activated in the time interval $[t_j, t_j + \Delta t]$ is identified as the one that minimized the sub-optimal measure

$$J^* = \int_{t_j}^{t_j + \Delta t} z^T Q z + u^T R u \; dt,$$

which expresses the cost-to-go and whose optimal value is given by,

$$J^* = z^T(t_j)P(x_i)z(t_j), \quad i = 1, 2, ..., q \quad (2.25)$$

for all $x_i$, where $P(x_i)$ solves the location parameterized ARE equation (2.24),
While, the author would have liked to employ full state feedback control, it was not viable practically as it an array of such sensors would add mass and damping to the system and non-contact solutions like the use of an image processing system operating at a minimum of 1000 Hz is beyond the scope of this investigation. So, an observer-based controller, LQG (Linear Quadratic Gaussian) was designed for the model,

\[
\begin{align*}
\dot{z} &= A z + B(\bar{x}_i)u + w_i \\
y &= C z + w_2,
\end{align*}
\]

where \( w_i, w_2 \) are the state noise and the observation noise, with error covariance’s \( E(w_i w_i^T), E(w_2 w_2^T) \) assigned to parameters \( \bar{Q} \) and \( N^{-1} \), respectively. In equation (2.26) the \( A \) matrix has been incorporated with the locations of the control and disturbance actuators along with their mass and stiffness contributions using [5]. \( B(\bar{x}_i) \) corresponds to the control actuator engaged by use of the proposed algorithm. The \( B(\bar{x}_i) \) vector is also incorporated with the mass and stiffness contribution associated with the particular actuator \( x_i \). Now we have the following equation for the Kalman filter (optimal state estimator),

\[
\dot{\hat{z}}_i = A \hat{z}_i + B(\bar{x}_i)u + L(y - C \hat{z}_i), \quad i = 1, 2, \ldots, q.
\]

(2.27)

The parameters \( \bar{Q}, N^{-1} \) were tuned in the following filter Riccati equation,

\[
\begin{align*}
A \Sigma + \Sigma A^T - \Sigma C^T N^{-1} C \Sigma + \bar{Q} &= 0, \\
L &= \Sigma C^T N^{-1},
\end{align*}
\]

to ensure that the observer dynamics governed by \( A - L C \) converged faster than the system dynamics which is governed by \( A - B(\bar{x}_i) \kappa(\bar{x}_i) \) for all \( i = 1, 2, \ldots, q \). The gain matrix is given by \( \kappa(\bar{x}_i) = R^{-1}B(\bar{x}_i)^T P(\bar{x}_i) \). The numerical simulations were designed to
realistically capture the time lag between the vibration data at a point, which precedes the overall controller action by one time step. The controller that corresponds to each actuator is given by,

$$u(\bar{x}_i) = -R^{-1}B(\bar{x}_i)^T P(\bar{x}_i) \hat{z}_i,$$ \hspace{1cm} (2.28)

where, $P(\bar{x}_i)$ satisfies equation (2.24) and each $\hat{z}_i$ is given by equation (2.27).

### 2.5 The Switching Algorithm

This algorithm is valid only for engaging a single actuator at any given time.

**Step 1.** For each possible actuator location $x_i$, compute $J_{opt}(x_i) = tr[P(x_i)]$, as given by equation (2.23) for a specific value of $n$. Also store the values of $P(x_i)$ corresponding to each actuator location $i = 1, 2, ..., q$.

**Step 2.** Based on the minimal values of $J_{opt}(x_i)$ and on other mechanical design considerations (such as interference from the structure or preexisting critical components at the proposed location), decide on the locations where the actuators are going to be located.

**Step 3.** If access to full state is not available, design a suitable observer as given by equation (2.27) and also design the corresponding controllers for each of the actuators as given by $u_i(t) = -R^{-1}B^T(\bar{x}_i)P(\bar{x}_i)\hat{z}_i(t)$.

**Step 4.** Since the algorithm will take up processing power especially if incorporated into true three-dimensional structures, the hardware capabilities and system bandwidth will affect the choice of a suitable time
interval $\Delta t$ after which the system will decide on the need to switch actuators or continue with the initial choice.

**Step 5.** Engage one suitable actuator from the lot, at say time $t = 0$.

**Step 6.** Start infinite loop (or for duration of controller engagement) \{At the end of every $\Delta t$ seconds, compute $J^* = \hat{z}^T(t_j)P(x_i)\hat{z}(t_j), \quad i = 1,2,\ldots,q$.

Engage the actuator that returns the minimum value of $J^*$ for the next $\Delta t$ seconds.\}.

**Step 7.** Deactivate system at end of mission.

It must be noted that $J^* = \hat{z}^T(t_j)P(x_i)\hat{z}(t_j), \quad i = 1,2,\ldots,q$ is also a sub optimal measure, but is used here for computational efficiency.
Chapter 3 Numerical Simulation Results

The most important aspect of this investigation lies in the seamless transition of the envisioned algorithm into a proof of concept hardware system. This was accomplished by maintaining a focus on the limitations of available hardware, which allowed for better computer simulations of the real system. A commercially available piezoelectric actuator ACX QP20N [26] was chosen for this work, considering the excellent electromechanical characteristics and the actual physical dimensions (individual actuators to be less than about 5% of the total length of the beam).

![ACX QP20N piezoelectric actuator](image)

Figure 2a: ACX QP20N piezoelectric actuator

The beam material was chosen to be Aluminum. The various physical properties and constants [27] associated with the experimental system are listed in Table 1. With the parameters in Table 1 as input, along with suitable choice of other tuning variables (detailed in Appendix A), the algorithm discussed in Chapter 2.3 was developed using MATLAB®. Fifty possible uniformly spaced actuator locations, each of length 0.0508 m (governed by the actual length of ACX QP20N), were considered. The actuator candidates were uniquely identified by their left end point location along the beam. Then their $J_{opt} = tr[P(x_i)]$ values as given by equation (2.23) were computed and plotted in Figure 3, as a function of their distance from the one end of the structure.
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of Aluminum beam between fixed-fixed supports</td>
<td>1 m</td>
</tr>
<tr>
<td>Width of Aluminum beam</td>
<td>0.0508 m</td>
</tr>
<tr>
<td>Thickness of Aluminum beam</td>
<td>0.00127 m</td>
</tr>
<tr>
<td>Young’s Modulus of Aluminum beam</td>
<td>$69 \times 10^9$ N/m$^2$</td>
</tr>
<tr>
<td>Density of Aluminum beam</td>
<td>2700 kg/m$^3$</td>
</tr>
<tr>
<td>Length of piezoelectric device</td>
<td>0.0508 m</td>
</tr>
<tr>
<td>Length of piezoelectric element</td>
<td>0.045974 m</td>
</tr>
<tr>
<td>Width of piezoelectric device / element</td>
<td>0.020574 m</td>
</tr>
<tr>
<td>Thickness of piezoelectric device / element</td>
<td>0.000127 m</td>
</tr>
<tr>
<td>Piezoelectric charge coefficient $d_{31}$</td>
<td>$-179 \times 10^{-12}$ m/V</td>
</tr>
<tr>
<td>Mass of one piezoelectric actuator</td>
<td>0.00481942 m</td>
</tr>
<tr>
<td>Young’s Modulus of piezoelectric element</td>
<td>$6.9 \times 10^{10}$ N/m$^2$</td>
</tr>
</tbody>
</table>

**Table 1:** Properties associated with the simulation.

<table>
<thead>
<tr>
<th>Actuator location from one end of structure (m)</th>
<th>Unique Actuator Number</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.03874</td>
<td>Patch 03</td>
<td>Controller</td>
</tr>
<tr>
<td>0.17434</td>
<td>Patch 10</td>
<td>Controller</td>
</tr>
<tr>
<td>0.91045</td>
<td>Patch 48</td>
<td>Controller</td>
</tr>
<tr>
<td>0.36806</td>
<td>Patch 20</td>
<td>Disturbance</td>
</tr>
<tr>
<td>0.71674</td>
<td>Patch 38</td>
<td>Disturbance</td>
</tr>
</tbody>
</table>

**Table 2:** Patch identification.
Figure 3: Plot of possible actuator locations versus corresponding $J_{opt} = tr[P(x)]$.

Based on the input from Figure 3 and due to limitations on the number of piezoelectric actuators that may be affixed on the beam, the locations identified in Table 2 were chosen as appropriate controller and disturbance patch locations for this investigation.

Figure 4a: Identification of the control patches for the computer simulation.
The disturbance patches are affixed to provide a repeatable disturbance signal for testing the patch-switching algorithm. They also offer greater flexibility in design or testing of newer algorithms. The disturbance patches and controller patches are affixed separately on the two sides of the aluminum beam.

A point velocity sensor is assumed to lie on the side of the beam on which the disturbance patches are affixed at a distance of 0.6667 m from one end of the beam. For a realistic simulation a suitable LQG controller was designed and implemented in addition to the switching algorithm, as discussed in Section 2.4. In essence the system that is being simulated is given by;

\[
\dot{z} = Az + B(\bar{x}_i)u(t) + Dw_i(t)
\]

\[
y = Cz,
\]

with the observer given by \( \hat{z}_i = A\hat{z} + B(\bar{x}_i)u(t) + L\left(y - C\hat{z}_i\right) \). The controller signal corresponding to each control actuator is given by \( u_i(t) = R^{-1}B(\bar{x}_i)^TP(\bar{x}_i)\hat{z}_i(t) \) and the choice of the control patch determined by the switching algorithm with \( \Delta t = 0.3 \) seconds. \( D \) is a distribution matrix of the disturbance patches (basically \( D = B(x_d) \), where \( x_d \) refers to the location of that particular disturbance patch) excited by a voltage signal.
The simulation in this investigation lasted 20 seconds, with the disturbance patches engaged in discrete intervals with the controller engaged after the first 8 seconds. The observer was provided access with the disturbance profile for the first 8 seconds only.

The disturbance patches are excited with signal weighing in with roughly the first four fundamental frequencies (only to elicit better response that can be picked up by the sensor as the objective was experimental implementation of the proposed algorithm) of the system in addition to a uniform random signal (which is captured and used for all simulations to ensure a uniform disturbance signal) and the entire system is simulated for 20 seconds. The disturbance signal traces are shown in Figure 5. The fundamental frequencies are given by the square root of the Eigenvalues of \( \left( M^T K \right) \). This compared favorably to the real system, which was excited with sinusoidal inputs of varying frequencies and their response observed and quantified on an oscilloscope to determine the first four fundamental frequencies. This was basically engineered by including the contributions due to the patches in the \( M, K \) matrices. See Section 2.2 for details.

<table>
<thead>
<tr>
<th>Computed from model (Hz), see equation (2.15.)</th>
<th>Experimentally determined (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.33</td>
<td>6.36</td>
</tr>
<tr>
<td>17.14</td>
<td>16.62</td>
</tr>
<tr>
<td>33.72</td>
<td>33.04</td>
</tr>
<tr>
<td>56.00</td>
<td>54.10</td>
</tr>
</tbody>
</table>

**Table 3:** Comparison of the first four fundamental frequencies.
Figure 5: Voltage signal applied to disturbance ‘Patch 20’ and ‘Patch 38’.

Since, access to the entire state of the system is unavailable due to limitations in the sensor system, an energy norm (sum of kinetic and strain energy) that employs the state provided by the observer and defined by,

\[ J = f(z) = \sqrt{\dot{w}^T \dot{M} \dot{w} + \dot{w}^T \dot{K} \dot{w}} \]  

(3.1)
is used as a performance measure. A plot showing the performance measure generated using a computer simulation (as given in APPENDIX A) of the system for both the open loop and the closed loop case (with and without the actuator switching algorithm employed) is shown in Figure 6. In Figure 6 the performance measure was computed
using the state provided by the observer, i.e., \( J = \sqrt{\dot{\hat{w}}^T \dot{\hat{w}} + \hat{w}^T K \hat{w}} \). This is basically implemented from the idea that,

\[
\begin{split}
  z^T Q z = \begin{bmatrix} w \end{bmatrix}^T \begin{bmatrix} K & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} w \\ \dot{w} \end{bmatrix}, \ 	ext{with} \ Q = \begin{bmatrix} K & 0 \\ 0 & M \end{bmatrix}.
\end{split}
\]

Other relevant plots such as displacement trace at the location of the accelerometer, actuator-switching trace, and trace of the input signal to the disturbance patches are included in this chapter. Figure 7 shows the better vibration alleviation capability of the algorithm in the measure of the displacement trace. Figure 8 is included to show that the observer closely follows the plant dynamics and Figure 9 indicates that the switching algorithm has been correctly incorporated in the computer simulation.

![Figure 6: Closed loop energy norm for switching plus feedback control (solid) with ‘Patch 03’, ‘Patch 10’, ‘Patch 48’ on call, feedback control plus single actuator (dash-dot) Patch 10, and the open loop case (dotted).](image)

Figure 6: Closed loop energy norm for switching plus feedback control (solid) with ‘Patch 03’, ‘Patch 10’, ‘Patch 48’ on call, feedback control plus single actuator (dash-dot) Patch 10, and the open loop case (dotted).
**Figure 7:** Displacement trace at 0.6667 m from one end of beam. Switching plus feedback control (solid) with Patch 03, Patch 10, Patch 48 on call, feedback control plus single actuator (dash-dot) Patch 10, and the open loop case (doted).

**Figure 8:** Displacement trace at location 0.6667 m from one end of the beam as given by the simulated plant (solid blue) and that estimated by the observer (red dotted). This plot is to demonstrate the efficiency of the observer.
**Figure 9:** The switching algorithm calls for different controllers at different instants of time. During the first 8 seconds, the controllers are not engaged.
Chapter 4 Experiment Design, Results

4.1 Hardware Implementation

The physical implementation of the proposed algorithm was done at ‘The Controls Laboratory’, Worcester Polytechnic Institute. The assembly was setup on a TMC passive vibration isolation table. A 1 m long aluminum beam with properties as described in Chapter 3, was bonded with 5 QP-20N patches sourced from ACX-CYMER at locations specified for the simulation (see Chapter 3). The suggested [28] vacuum bonding procedure was employed. The controller patches were bonded on one side of the beam, while the disturbance patches were bonded on the other side. The point sensor was placed on the same side as the disturbance patches at a location of 0.6667 m from one end of the beam. Fixed-Fixed boundary conditions were applied through the use of simple fixtures, which were designed to allow for easy wiring and were screwed onto the TMC vibration isolation table, see Figure 11.

The sensor used is basically a PCB Piezotronics Shear ICP accelerometer Model number 352A24 with a sensitivity of 102.3 mV/g in the range of 1-8000 Hz. This sensor was used to provide velocity data by use of a Dual-Mode Amplifier and signal conditioner Model 443B101, sourced from PCB Piezotronics. This particular unit allowed for integration of the incoming signal and thus provided the point velocity input. The inbuilt Low Pass filter allowed for removing any aliasing effects. The PCB signal conditioner unit applied a Low Pass filter tuned at 100 Hz. The signal was amplified to provide a sensitivity of 100 V/m/s, which was removed using ‘Gain’, blocks in the
software associated with the implementation of the algorithm and the associated controller.

The output from the digital system was passed through a 4 channel Low Pass Butterworth filter, Krohn-Hite Model 3360. This hardware filter was not critical although it appeared to reduce some high frequency noise in the audible range from the controller output. The power amplifiers used for the piezoelectric patches were sourced from CYMER-ACX, identified by Model EL-1224. One power amplifier, Model 7602M was sourced from Krohn-Hite. The specific details of operation are elaborated in Appendix B.

![Figure 10: Front view of the hardware implementation.](image)

The wiring for the control patches was implemented using the CB 011 connected sourced from CYMER-ACX. This did not add significant mass as the control patches
were near the clamping assembly. However, the disturbance patches required thinner wiring, so that un-modeled dynamics could be kept to a minimum. All other electrical connections were done using standard 50 Ohm BNC connectors.

![Image](image.png)

**Figure 11:** View of the disturbance patches and the sensor at 0.6667 m from left end.

### 4.2 Rapid Control Prototyping on dSPACE ACE 1103 Board

The algorithm that has been developed was implemented on the commercially available dSPACE ACE 1103 based Rapid Control Prototyping system at The Controls Laboratory. The DS 1103 board used in this investigation is based on MATLAB®/Simulink®, a widely used control development software. It allows for the design of the controller graphically in the Simulink® block diagram environment. Using the Real-Time Interface to Simulink®, the control algorithms are downloaded to the DS1103 PPC Controller Board to interface with the actual hardware.

The DS1103 board is equipped with a Motorola PowerPC 604e processor for fast floating-point calculation at 333 MHz. It is a high-performance super scalar
microprocessor that has three integer execution units, one floating-point arithmetic unit, and a separate load/store unit for fast memory access. The on-chip cache size is 32 KB. A 64-bit time base is used for execution time measurement and generation of time stamps. A 2 MB local memory is used for program and data of the simulation model. The local memory is fully cached and cannot be accessed by the host PC in standard operating mode.

![Rapid Control Prototyping system at The Controls Laboratory, WPI.](image)

**Figure 12:** Rapid Control Prototyping system at The Controls Laboratory, WPI.

For data buffering and exchange between the PowerPC and the host, 128 MB of non-cached global memory is present. The host interface of the board, in this case a Pentium 4 based system, was used to perform board setups, program downloads, and runtime data transfer. High-resolution A/D converter (16-bit) with a sampling time of 4 µs was used to read in the point velocity sensor data and the controller communicated to
the ACX QP 20N actuators using a D/A output channels with a resolution of 14-bit and a 5 µs settling time.

4.3 Real-Time Interface to Simulink®

The Real-Time Interface enhances the Simulink® block library with additional blocks, which provide the link between Simulink® and the real-time hardware. These blocks cover the I/O functionality of the prototyping hardware. To graphically specify an I/O channel the corresponding block icon has to be picked up from the I/O block library and attached to the Simulink® controller model. For multitasking applications, a pre-emptive scheduler guarantees real-time behavior with response times of a few microseconds. Tasks and priorities are also defined graphically within the Simulink® block diagram.

The Simulink® model was then transferred into real-time code, using the Real-Time Workshop and the Real-Time Interface. Code generation includes the I/O channel specification and the multitasking setup, which are translated into appropriate function calls of the Real-Time Library. This library is a C function library providing a high-level programming interface to the hardware. The Real-Time Library includes access functions for the slave TMS320F240 slave DSP micro controller.

The screen snapshot of the actual implementation of the controller and the developed algorithm is shown in Figure 13. The digital implementation of the controller required the conversion of the continuous time model into a discrete time model. This was done using the MATLAB® command c2d. A MATLAB® code that automatically
generates the discrete LQR based observer with a time step of 0.001 seconds, when given an input of the relevant continuous time model is given in APPENDIX A.

Figure 13: Screen shot of the implementation of the proposed algorithm in SIMULINK®.

The implementation is basically timed by a ‘Digital clock’, which also serves as the base for driving the disturbance patches using a predetermined disturbance trace over a period of 20 seconds apart from timing the engagement of the controllers after the first 8 seconds. The part of the Simulink® model that is associated with the disturbance patches is independent of the rest of the model and is shown in red in Figure 13.

The ADC block provides the input from the point velocity sensor, which is passed through an Observer subsystem that estimates the state of the vibrating beam. The resulting estimate is passed into the ‘Implementation of algorithm, Assigning control’ block through a precursor block ‘Switching Algorithm Precursor’, which determines
which control actuator to engage at that particular time interval. Initial default is Controller Patch 03. This switching algorithm is implemented every 0.3 seconds; i.e. $\Delta t = 0.3$ seconds (governed by the ‘Switch decision or Algorithm call timer’ block leading from the ‘Digital clock’ block). At any particular time interval, only one controller patch is engaged. The intricate details of the functioning of each block are provided in APPENDIX B. The ‘To Compute performance measure J’ block takes the input of the state estimate and computes the performance measure as described by equation (3.1). The real time interface provided by the Control Desk software allows for the capture of the various data sets from the physical system.

Figure 14 clearly shows the functionality of the disturbance subsystem. Figure 15 offers physical proof of the successful implementation of the proposed online switching algorithm. Figure 16 is probably the most important result of this investigation as it shows the experimentally determined energy norms for the cases of switching with feedback control, no control and the application of a single fixed actuator. During the generation of data for the three cases there was no change whatsoever to either the disturbance applied to the system or to the sensor system. Hence it clearly proves that switching available actuators to control vibration is better than the use of a single fixed actuator. The discrepancy in the magnitude the values of the energy norm between the simulation and the experimental system is attributed to scaling in the sensor system. This is not significant as the disturbance and the sensor system was constant for all the three cases tested and the scaling has appeared equally in all the profiles.
4.4 Experimental Results

Figure 14: The voltage trace applied to the disturbance ‘Patch 20’ and ‘Patch 38’. The presented experimental results were in response to this disturbance trace.

Figure 15: Trace of the control signal, switching between ‘Patch 03’ and ‘Patch 48’.
Figure 16: Closed loop energy norm for Switching plus feedback control (solid) with ‘Patch 03’, ‘Patch 10’, ‘Patch 48’ on call, feedback control plus single actuator (dash-dot) Patch 10, and the open loop case (dotted).
Chapter 5 Conclusions and Recommendations

This investigation resulted in the development of a general-purpose finite element code for uniform, isotropic beams subjected to excitations by use of moment actuators and any kind of boundary condition. An algorithm to determine the best locations for the actuators as proposed by Demetriou [14] was implemented. This investigation was focused to keep commercial implementation a viable idea, by near realistic simulation of the proposed algorithm. This was mainly achieved by using the properties of commercially available piezoelectric actuators and accounting for the lack of full state feedback.

The simulations clearly showed that the proposed switching algorithm (Chapter 2.5) was viable even without full state feedback, although a refining of the model through the addition of the mass and stiffness effects of the actuators was done at a later stage. The switching algorithm was successfully implemented on the Rapid Control Prototyping system. This implementation is robust and will serve as a template for further online switching studies. Some of the features of this online switching model (see Figure 13) are

1. Easy modification of input or disturbance conditions in the ‘Fire Pattern Subsystem’.
2. The time interval between calls of the switching algorithm can be varied by simply changing the number in the ‘Fcn’ block.
3. The logic associated with the ‘Switching Algorithm Precursor’ and ‘Implementation of algorithm, Assigning control’ blocks can be used for the implementation of similar algorithms for varied applications.
The results obtained (Figures 14, 15, 16) clearly demonstrate successful implementation of the switching algorithm for controlling vibrations in structures. This, albeit a ‘Proof of Concept’ investigation, clearly shows the superior performance of activating actuators by the use of the proposed switching policy over the use of a single or fixed actuator. This superior performance is especially pronounced, when there is no prior knowledge of the disturbance signal. It can be inferred that this controller design algorithm is ideally suited for vibration control in distributed parameter systems where prior knowledge of the spatial-temporal variations in disturbance is lacking.

This particular physical system which employed a Power PC 640e processor clocked at 400 MHz could deliver for this particular system a controller at 1000 Hz, which could be extended to 2000 Hz for the configuration discussed in Chapter 4. Since the mechanical frequencies considered were less than 200 Hz, a scaling of the system by a factor of 10 is possible considering a sampling rate in the region of 300 to 400 Hz. Newer developments in the incorporation of Just In Time (JIT) programming techniques such as those associated with the release of Matlab® 6.5 allow for extending this basic system towards vibration control applications in structures modeled in 2D and 3D. 64 Bit processors, which are due in mid of 2003, will enable this system to be truly commercialized.

Future work can involve verification of the proposed algorithm on a plate structure and in the use of point force actuators that will allow for true commercialization of this work. It is imperative future simulation models be developed in discrete time, as implementation of such systems will invariably be on digital computers with a finite time
step. It is also suggested that the model includes the essential composite nature of the system; however this is easier said than practiced as a fine line needs to be maintained between modeling or computational costs and experimental costs.
Appendix A

Main program Matlab® code

%******************************************************************************
% 1D FEA model using cubic B splines for beams with
% piezoactuators
% Murali Murugavel
% HL 248 The Controls laboratory
% Department of Mechanical Engineering
% Worcester Polytechnic Institute MA 01609 USA
%******************************************************************************

global A L n epi jt xo;
tic
L=1;%L=input('Enter the length of the beam :');
E=75e9;%input('Enter the Youngs Modulus of the beam :');
n=20;%n=input('Enter the number of nodes to be considered on the beam :');
p=2300*0.0508*0.00127;% mass/unit length, ie density*width*thickness input('Enter the density of the 
beam :');
pact=0.00481942/0.0508;% mass/unit length of actuator 2 inches length & not 1.81 inches
nx=2001;%defining the number of splines
h=L/n;

% Assembly of Mass and Stiffness Matrices
mkassembly;

% Assembling the tranformation Matrix T & incorporating a choice
transformat;
[ro,co]=size(T); % rows,columns

% Applying the Transformation Matrix on the M & K Matrices
Mtilda=T*M*T';
Ktilda=T*K*T';

b=0.0508;% 2 inches input('Enter the width of the Cantilever beam :');
d=0.00127;% 0.05 inches input('Enter the thickness of the Cantilever beam :');

l=(b*d^3)/12;
alpha1=E*l/1000;%input('Enter the Kevin-Voigt/Viscoelastic damping
  %Coefficient, generally equals EL/1000 :')
alpha2=0;%input('Enter the Air damping coefficient :')

Mtilda=Mtilda*p; % Multiplying by Density of the material of the beam
Ktilda=E*l*Ktilda; % Multiplying by Young's Modulus and Moment of Inertia of the beam

epi=0.0508; % length of the actuator

for i=1:nx
  yo(i)=(i-1)*L/(nx-1);
  for j=1:n+3
    KK(j,i)=xofunc(yo(i),j-2,n,L);
    KK1(j,i)=xoslope(yo(i),j-2,n,L);
  end
end

43
KK2(j,i)=xoslope2(yo(i),j-2,n,L); % curvature of Bsplines

% Check if KK is (n+3)*(nx-1)
TK=T*KK; % Check if TK is a ro*(nx-1) matrix
TK1=T*KK1;
TK2=T*KK2;

% Generating MATRIX B
K31=8.038e-4;%input('Enter the Piezoelectric Patch axis 31
electromechanical constant :');
K31= 0.000354990935178; %
epi=0.0508;% 2 inches or can use L/20 defining length of actuator
% (ACX QP 20N dimensions) % epi=L/n; for testing
npal=50; % defining Number of Possible Actuator Locations
for i=1:npal
    xo(i)=(i-1)*(L-epi)/(npal-1);
end

for i=1:npal % Number of Possible LHS locations of Actuators
    for j=1:n+3
        xolhs=xoslope(xo(i),j-2,n,L);
xorhs=xoslope(xo(i)+epi,j-2,n,L);
        B_Matrix(j,i)=K31*(xorhs-xolhs);
    end
end
B_Matrix=T*B_Matrix;

% Accounting for the masses of the actuator massactua;
R=1e-6;%R=input('Enter the value of R ~ 1e-6 :')
wpos=1.0;wvel=wpos/100;
wpos=1.0;wvel=wpos/100;
Mtemp=zeros(ro,ro);
for it=1:npal
    % Generating MATRIX A
    a11=zeros(ro,ro);
a12=eye(ro,ro);
    Mtemp(:,:,it)=Mglobal(:,:,it);
a21=-inv(Mtemp)*Ktilde;
    D=alpha1*Ktilde+alpha2*Mtemp;
    a22=alpha1*a21-alpha2*a12;
    A=[a11 a12; a21 a22];

    % State space assembly of B matrix
    Btemp = ro*length(xo) Matrix
    B=[zeros(ro,1); inv(Mtemp)*B_Matrix(:,:,it)];
    Q=[wpos*Ktilde zeros(size(Ktilde)); zeros(ro,ro) wvel*Mtemp];
    P(:,:,it)=are(A,B*inv(R)*B',Q);
    J(it)=trace(P(:,:,it));
end
plot(xo,J,'r')% the minimum values of J give the optimal location for the actuators

% Input of the initial displacement condition
tipdisp=0.02;% input('Enter the tip displacement for the
% cantilever beam in meters; typically 2 cms :');
for co=1:n+3
    jj=co-2;
    xpos(co)=xofuncinteg(jj,n,L,tipdisp);
end
xpos=xpos';
% Check on initial displacement
alphaweighting=inv(Mtilda)*T*xpos;
curvecheck1=TK'*alphaweighting;
for i=1:nx
    x1(i)=((i-1)/(nx-1))*L;
end
%er=((2*(x1/L).^2-(4/3)*(x1/L).^3+(1/3)*(x1/L).^4)*tipdisp);for cantilever
er=(16*tipdisp/(L^4))*x1.^4-(32*tipdisp/L^3)*x1.^3+(16*tipdisp/L^2)*x1.^2; % for fixed fixed
plot(x1,curvecheck1*p,x1,er,'r');
% return

% LQR design
a21temp=(Mglobal(:,:,3)+Mglobal(:,:,10)+Mglobal(:,:,20)+Mglobal(:,:,38)+Mglobal(:,:,48))-4*Mtilda;
% Mass matrix considering Actuator mass
% a21temp = Mass matrix
a21=inv(a21temp)*Ktilda;
a22=alpha1*a21-alpha2*a12;
A=[a11 a12; a21 a22];
Q=[wpos*Ktilda zeros(size(Ktilda)); zeros(ro,ro) wvel*a21temp];
for loc=1:50
    Bassign=[zeros(ro,1); inv(a21temp)*B_Matrix(:,loc)];
    [Kmat,Smat,Emat]=lqr(A,Bassign,Q,R);
    Bmat_fc(:,loc)=Bassign;
    Kmat_fc(loc,:)=Kmat;
end

%[Kmat,Smat,Emat]=lqr(A,B(:,1),Q,R); % for single actuator at the base of the cantilever
%[Kmatone,Smat,Emat]=lqr(A,B(:,1),Q,R); % defining Gain matrix for left end actuator
%[Kmatmid,Smat,Emat]=lqr(A,Bmid,Q,R);% defining Gain matrix for mid actuator
%[Kmatend,Smat,Emat]=lqr(A,Bend,Q,R);% defining Gain matrix for right end actuator

[aro,aco]=size(A);
% simulation of switching / firing pattern
acloc=1334;
t0=0; tf=20; % define initial and final time
%x0=initialalphas; % define the initial condition
x0=zeros(2*n-2,1); % Initializing the state to zero
xp(1,:)=x0';
xphat(1,:)=x0';
xhatpass=x0;
xc0=x0; % initializing the observer state to zero
nsteps = tf/100; hsteps = (tf-t0)/nsteps; %define # of steps in the interval [t0,tf]
options = odeset('JConstant','on', 'RelTol',1e-6, 'AbsTol',1e-6);
nip=2;
% start the time do-loop
global A Kchoice Bchoice C Lmat y xhatpass Bd1 Bd2 w1 w2 w3 w4 ra

% Design of Observer

C=[zeros(1,n-1) TK(:,accloc)']; % based on info from sensor at about 
% 1/3rd the distance from one end of beam
Q2=0.01*eye(size(Q));R2=1;
[Lmat,Sobsv,Eobsv]=lqr(A',C',Q2,R2);
Lmat=Lmat';% A-Lmat*C faster than A-B*Kmat
% [max(real(eig(A-Lmat*C))) max(real(eig(A-B*Kmat)))]
%Ac=A-Lmat*C+B*Kmat;
%Bc=Lmat;
%Cc=-Kmat;

w1=7.45462*2*pi;w2=20.54*2*pi;w3=40.28454*2*pi;w4=66.59238922*2*pi;
d1=20;d2=38;s1=3;s2=10;s3=48;
Bd1=Bmat_fc(:,d1);Bd2=Bmat_fc(:,d2);
firepattern=1;Kchoice=Kmat_fc(s1,:);Bchoice=Bmat_fc(:,s1);
time(1)=0; xsol(1,:)=x0'; **% save the initial time in the vector time**
jf=0;

w=1; for m=1:nsteps

tleft=(m-1)*hsteps; % define initial time of ith subinterval
tr=m*hsteps; % define final time of ith subinterval
ra=unirnd(-1,1);

[t,x] = ode23('xprimeswitch_cl',[tleft:(tr-tleft)/nip:tr],x0',options);
y=C*x(3,:);
[t,xhat]=ode23s('xprimecom',[tleft:(tr-tleft)/nip:tr],xcO',options);

f1=xhat(3,:)*(P(:,:,s1))*xhat(3,:);
f25=xhat(3,:)*(P(:,:,s2))*xhat(3,:);
f50=xhat(3,:)*(P(:,:,s3))*xhat(3,:);
firechoice=min([f1 f25 f50]);
fp=rem(tr,0.3);
if fp=0
   jf=jf+1;
timef(jf)=tr;

if firechoice==f1
   firepattern(jf)=s1;
   Kchoice=Kmat_fc(s1,:);
   Bchoice=Bmat_fc(:,s1);
end
if firechoice==f25
   firepattern(jf)=s2;
end

end
Kchoice = Kmat_fc(s2,:);
Bchoice = Bmat_fc(:,s2);
end

if firechoice == f50;
firepattern(jf) = s3;
Kchoice = Kmat_fc(s3,:);
Bchoice = Bmat_fc(:,s3);
end
disp([tleft tr firepattern(jf)])
end

xp(m+1,:) = x(nip+1,:);  % assign to the solution vector xp the current
% value at t=tr
xhat(m+1,:) = xhat(nip+1,:);  % assign to the solution vector xhat the current value of observer at t=tr
time(m+1) = tr;  % save the next entry in the time vector
xhatpass = xhat(nip+1,:);  % assigning xhat to find u in the defined functions
x0 = xp(m+1,:);  % assign the solution at t=tr to be the initial
xO = xhat(m+1,:);
xnorm(1+m) = xhat(m+1,1:n-1)*Ktilda*xhat(1+m,1:n-1)'+xhat(1+m,n:2*n-2)*Mtilda*xhat(1+m,n:2*n-2)';
% condition for the next time sibinterval
end  % end of time do-loop
figure(2),plot(time,TK(:,accloc)'*xp(:,1:n-1)','r') % plot of mid span
[firepattern; timef] % Firepattern
figure(1),plot(time,sqrt(xnorm),'r') % Plot of the xnorm
pause
toc

C = [zeros(1,n-1) TK(:,accloc)'];
Q2 = 0.01*eye(size(Q)); R2 = 1;
[Lmat, Sobs, Eobs] = lqr(A', C', Q2, R2);
Lmat = Lmat';  % A-Lmat*C faster than A-B*Kmat  % [max(real(eig(A-Lmat*C))) max(real(eig(A-B*Kmat)))]
Figure 17: Disturbance patch firing subsystem.

This section is to explain the details of each subsystem described in Figure 13. This section is also meant to document small details associated with the physical setup described in Chapter 4.

The ‘Fire pattern’ subsystem basically takes the input from the digital clock and using a series of ‘If’ statements, creates a disturbance profile that is fed out of the system into the ‘Disturbance patch signal conditioner’ block, where the outputs to the disturbance patches are divided by 10 X 20 in blocks DP20 and DP38, before being passed into a Analog Low Pass filter block (Optional) and saturation block (this keeps the total output
in the region – 0.99 to + 0.99). Remember that dSPACE interface board converts an incoming signal in the range of +10 to –10 Volts into a number between +1 to –1. This logic is reversed in the output scenario. This accounts for dividing the signal in blocks labeled DP20 and DP38 by 10. The 20 comes from the gain associated with the piezoelectric amplifier. The piezoelectric amplifier’s gain is adjusted by using a small screw driver and calibrating the same using a function generator and an oscilloscope.

**Figure 18:** Disturbance patch signal conditioner subsystem.

The ‘Compute performance measure J’ block is self-explanatory as it basically takes in the estimate of the state and applies equation (3.1) to it. The ‘Observer, Controller block’ subsystem basically implements the LQR based observer in discrete time. The most important note lies in the ‘Gain’ block, which applies a gain of 0.1 to the
incoming accelerometer signal (which was amplified by 100 at the source by the PCB signal conditioner). The dSPACE interface board has already applied a gain of 0.1.

Figure 19: Subsystem to compute performance measure J.
Figure 20: Subsystem showing Observer, Controller block.

This brings up the most crucial task of implementing the switching algorithm. This algorithm is implemented by the defining of memory variables that can be read at any time, basically some sort of Global memory variables. The variables are C03, C10 and C48 (corresponding to Patch03, Patch10 and Patch48 respectively). These are binary variables wherein they are initialized with 1, 0, and 0 respectively. These are applied using multiplication blocks in the ‘Observer, Controller block’ and ‘Signal conditioning, Fire control’ subsystems to allow only the relevant signal to pass.
Figure 21: Allocation of relevant memory variables.

The other memory variables are J03, J10, J48 (corresponding to Patch03, Patch10 and Patch48 respectively), which stores the numerical values computed for each of the patches, by use of equation (2.25), every time control is passed in to the ‘Switching Algorithm Precursor’ subsystem. Memory variables Pat03, Pat10, Pat48 (corresponding to Patch03, Patch10 and Patch48 respectively), stores the $P(\xi_i)$ value corresponding to each of the actuators.
Figure 22: The switching algorithm precursor subsystem.

The ‘Merge2’ block in the ‘Switching Algorithm Precursor’ subsystem merges the values J03, J10, J48 and passes it to the ‘Minimum’ block which reorders them and gives the indices of the vector (in ascending order). This information is passed into the ‘Implementation of Algorithm, Assigning Control’ subsystem which assigns the variables C03, C10, C48 their specific binary values, which in turn determine which of the patches are engaged in the other subsystems. The other setup parameters are presented in the following screen shots.
Figure 23: The implementation of the algorithm subsystem.

Figure 24: Signal conditioning, Controller engagement subsystem.
Figure 25: Parameters that need to be input into the Simulink® model.
Control Desk is used to capture data and the relevant screen shot is presented below. The main trick is to begin capture of data points from \( t = 0.0 \) seconds. This is implemented by the following procedure. For Simulink®/RTI models, to capture data:

The simState variable in each model controls its simulation state, i.e. it determines if the simulation is stopped, paused or running. By default, simState is initialized with RUN, so that the simulation is started immediately after the download:

You can use the RTI Options to initialize the simState variable with PAUSE or STOP so that the simulation does not start after the download. With single-processor RTI, add SSTATE=STOP or SSTATE=PAUSE to the Make command. You can find it on the Real-Time Workshop page of the Simulation Parameters dialog if "Target configuration" is the selected category.

1. Build and download the model as usual.
2. Start ControlDesk and prepare your experiment.
3. Connect a virtual instrument such as the RadioButton instrument to the simState variable. Use the following values:

   \[
   \begin{array}{c|c}
   \text{STOP} & 0 \\
   \text{PAUSE} & 1 \\
   \text{RUN} & 2 \\
   \end{array}
   \]

4. Connect the variables you want to capture to suitable data acquisition instruments.

Select the \textbf{Edit Capture Settings} context menu command of the data acquisition instrument to show the \textbf{Capture Settings} window and clear the \textbf{Auto Repeat} checkbox. This ensures that the captured data is not overwritten after the plot is complete.
5. Start ControlDesk's Animation mode.

6. Switch the simState to RUN. The data is traced from the first sampling step.

7. If your model is already built, downloaded and running, you can also use ControlDesk to switch the simState to "STOP" (see steps 4 to 7). Then restart the data capture services via the Start/Stop button in the Capture Settings window and set the simState back to RUN”.

**Figure 26:** Parameters that need to be input into Control Desk.
References


[8] Chandrashekara, K. and Agarwal, A.N. Active vibration control of laminated


Hanagud, S., Obal, M.W., and Calise, A.J. Optimal Vibration control by the


[50] Sosa, H. and Castro, M. Electroelastic analysis of piezo electric laminated


