

7-2006

## Comparison of sliding mode and state-feedback control applied to a partially treated actively constrained layer damped (ACL D) beam

Jose J. Rodriguez  
*University of Texas-Pan American*

Follow this and additional works at: [https://scholarworks.utrgv.edu/leg\\_etd](https://scholarworks.utrgv.edu/leg_etd)



Part of the [Mechanical Engineering Commons](#)

---

### Recommended Citation

Rodriguez, Jose J., "Comparison of sliding mode and state-feedback control applied to a partially treated actively constrained layer damped (ACL D) beam" (2006). *Theses and Dissertations - UTB/UTPA*. 751.  
[https://scholarworks.utrgv.edu/leg\\_etd/751](https://scholarworks.utrgv.edu/leg_etd/751)

This Thesis is brought to you for free and open access by ScholarWorks @ UTRGV. It has been accepted for inclusion in Theses and Dissertations - UTB/UTPA by an authorized administrator of ScholarWorks @ UTRGV. For more information, please contact [justin.white@utrgv.edu](mailto:justin.white@utrgv.edu), [william.flores01@utrgv.edu](mailto:william.flores01@utrgv.edu).

COMPARISON OF SLIDING MODE AND STATE-FEEDBACK CONTROL  
APPLIED TO A PARTIALLY TREATED ACTIVELY CONSTRAINED LAYER  
DAMPED (ACLD) BEAM

A Thesis

By

JOSE J. RODRIGUEZ

Submitted to the Graduate School of the  
University of Texas Pan American  
In partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

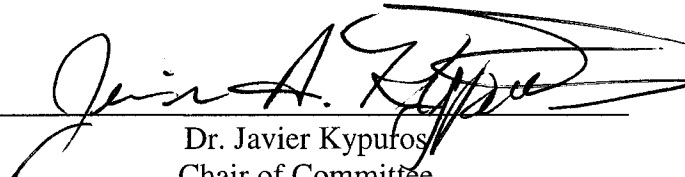
July 2006

Major Subject: Mechanical Engineering

COMPARISON OF SLIDING MODE AND STATE-FEEDBACK CONTROL  
APPLIED TO A PARTIALLY TREATED ACTIVELY CONSTRAINED LAYER  
DAMPED (ACLD) BEAM


A Thesis  
By  
JOSE J. RODRIGUEZ

Approved as to style and content by:



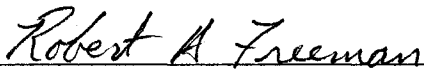
---

Dr. Javier Kypuros  
Chair of Committee



---

Dr. Arturo Fuentes  
Committee Member



---

Dr. Robert Freeman  
Committee Member

July 2006

## ABSTRACT

Rodriguez, Jose J., Comparison of Sliding Mode and State-Feedback Control

Applied to a Partially Treated Actively Constrained Layer Damped (ACLD) Timoshenko

Beam. Master of Science (MSE), December 2005, 00 pp., 19 figures, 2 tables, 28

references

In this research, a sliding mode control (SMC) was utilized in the control of a partially treated, actively constrained layer damped (ACLD), Timoshenko beam model. The resulting vibration control was compared to the vibration control achieved by a state-feedback linear quadratic regulator (LQR) for several loading conditions. An observer was designed and model order reduction (MOR) was performed to achieve a simplified, efficient, and more controllable finite element system model. As a result of model simplification, modeling errors in the form of unstructured uncertainties were introduced into the system. It was determined that the SMC and LQR achieved similar vibration control for all loading conditions when saturation limits were imposed. The saturation limits were enforced to replicate realistic voltage constraints. Saturation limits were then removed to investigate the ideal control action of the SMC and LQR. The ideal case revealed that the SMC achieved a significant reduction in the maximum deflection and settling time (as much as 37.44% and 16.61%, respectively) for all loading conditions when compared to the LQR. The improvement in response was due to the increase in control activity and the utilization of a robust control scheme in the presence of unstructured uncertainties.

## ACKNOWLEDGEMENTS

I would like to extend my deepest gratitude and many thanks to my thesis advisor, Dr. Javier Kypuros, for all his guidance and support throughout this work. His research and insight on ACLD treatments were invaluable to this thesis. I would also like to thank the other members of my committee, Dr. Arturo Fuentes and Dr. Robert Freeman. I am grateful for their input and have sincerely enjoyed learning from them over the course of my college career.

I would also like to express my gratitude to my parents John and Norma Rodriguez. This work would not have been possible without their advice and encouragement.

## TABLE OF CONTENTS

ABSTRACT.....	iii
ACKNOWLEDGEMENTS.....	iv
TABLE OF CONTENTS.....	v
LIST OF TABLES.....	viii
LIST OF FIGURES.....	ix
CHAPTER 1: INTRODUCTION.....	1
CHAPTER 2: SYSTEM MODEL.....	7
2.1 INTRODUCTION TO MODEL DERIVATION.....	7
2.2 SYSTEM DESCRIPTION.....	8
2.3 BEAM LAYER FINITE ELEMENTS MATRICES.....	11
2.4 VISCOELASTIC LAYER ELEMENTS MATRICES.....	13
2.5 PIEZOELECTRIC LAYER ELEMENTS MATRICES.....	15
2.6 MODEL ASSEMBLY.....	17
2.7 STATE SPACE REPRESENTATION.....	19
2.8 CONTROLABILITY AND OBSERVABILITY.....	20
CHAPTER 3: MODEL MODIFICATION.....	22
3.1 INTRODUCTION TO MODEL MODIFICATION.....	22
3.2 OBSERVER DESIGN.....	22
3.3 MODEL ORDER REDUCTION.....	24

CHAPTER 4: CONTROL METHODS.....	29
4.1 INTRODUCTION TO CONTROL METHODS .....	29
4.2 LQR DESIGN .....	29
4.3 SLIDING MODE CONTROL DESIGN .....	31
CHAPTER 5: SIMULATION AND RESULTS.....	35
5.1 INTRODUCTION TO SIMULATION AND RESULTS .....	35
5.2 EXPERIMENTAL SETUP.....	35
5.3 CASE 1: SINGLE 1-N PULSE.....	36
5.4 CASE 2: DOUBLE 1-N PULSE.....	41
5.5 CASE 3: HARMONIC LOAD.....	44
CHAPTER 6: CONCLUSION.....	49
REFERENCES.....	52
APPENDIX A: MATLAB FILES.....	55
1. MixedBeamModel.m .....	57
2. SystemMatices03.m.....	61
3. TimoshenkoFEM2.m .....	64
4. BalamuraganNarayananDerivation.m.....	66
5. BalamuruganNarayananFEM2.m.....	68
6. FEASMBL.m.....	72
7. FEASMBL2.m.....	73
8. schurmrJAK.m.....	74
APPENDIX B: SIMULINK FILES.....	76

1. timoshenko_beam_lqr.....	77
2. Tip Load Duration and Original Plant Blocks.....	78
3. Observer and Model Order Reduction Blocks.....	79
4. LQR and SMC Blocks .....	80
VITA.....	81



## LIST OF TABLES

TABLE 1: MODEL PARAMETERS.....	36
TABLE 2: SUMMARY OF RESULTS .....	48

## LIST OF FIGURES

FIGURE 1: PCLD, ACTIVE CONTROL, AND ACLD TREATMENTS .....	2
FIGURE 2: CONTROL SYSTEM BLOCK DIAGRAM.....	6
FIGURE 3: MODEL CONFIGURATION .....	8
FIGURE 4: DEGREES OF FREEDOM FOR A SCLD TREATED BEAM.....	10
FIGURE 5: HANKEL SINGULAR VALUES FOR ACLD TREATED BEAM.....	28
FIGURE 6: CHATTERING AND BOUNDARY LAYER.....	31
FIGURE 7: SIGNUM AND SATURATION .....	34
FIGURE 8: SINGLE 1-N PULSE .....	37
FIGURE 9: 1-N PULSE WIHTOUT ACTIVE CONTROL .....	37
FIGURE 10: 1-N PULSE: LQR WITH SATURATION .....	38
FIGURE 11: 1-N PULSE: SMC WITH SATURATION.....	39
FIGURE 12: 1-N PULSE: LQR WITHOUT SATURATION.....	40
FIGURE 13: 1-N PULSE: SMC WITHOUT SATURATION .....	40
FIGURE 14: DOUBLE 1-N PULSE.....	41
FIGURE 15: DOUBLE 1-N PULSE: LQR WITH SATURATION.....	33
FIGURE 16: DOUBLE 1-N PULSE: SMC WITH SATURATION .....	33
FIGURE 17: DOUBLE 1-N PULSE: LQR WITHOUT SATURATION.....	34
FIGURE 18: DOUBLE 1-N PULSE: SMC WITHOUT SATURATION .....	35
FIGURE 19: HARMONIC LOAD.....	44

FIGURE 20: FREQUENCY RESPONSE OF FULL ORDER MODEL .....	45
FIGURE 21: HARMONIC LOAD: LQR WITH SATURATION .....	45
FIGURE 22: HARMONIC LOAD: SMC WITH SATURATION .....	46
FIGURE 23: HARMONIC LOAD: LQR WITHOUT SATURATION.....	47
FIGURE 24: HARMONIC LOAD: SMC WITHOUT SATURATION.....	47

## CHAPTER 1

### INTRODUCTION

Vibration and noise control have long been an area of interest in many fields of engineering including mechanical, civil, and aerospace engineering. The effects of vibration can be undesirable because repetitive oscillations in a structure can often lead to problems such as excess noise, fatigue failure, and structural instability. Advances in technology and research have led to several methods in controlling vibration. One common method is to add damping to a structure through constraining layer treatments. The most common of these treatments include passive constrained layer damping (PCLD) treatments, active damping treatments, and active constrained layer damping (ACLD) treatments.

A common method in attenuating vibration is to add passive constrained layer damping (PCLD) to the vibrating structure. PCLD treatments are constructed by sandwiching a viscoelastic material (VEM) between the vibrating structure and a constraining layer as shown in Figure 1(a). This method dissipates vibration energy through heat in the cyclic shearing of the viscoelastic material and results in a steady decrease in the vibration response. Passive damping is applicable for a wide range of vibrating frequencies and does not require any control hardware or algorithms. Passive damping also has the desirable feature of being able to lower the resonance peaks of the

vibrating structure without significantly altering the mass or stiffness properties [2].

These features have led to the study and implementation of PCLD treatments for several applications including those in the automotive, aerospace, and civil industries [2, 19, 21].

The major disadvantage of using passive damping is the inability to alter the damping properties once they have been installed.

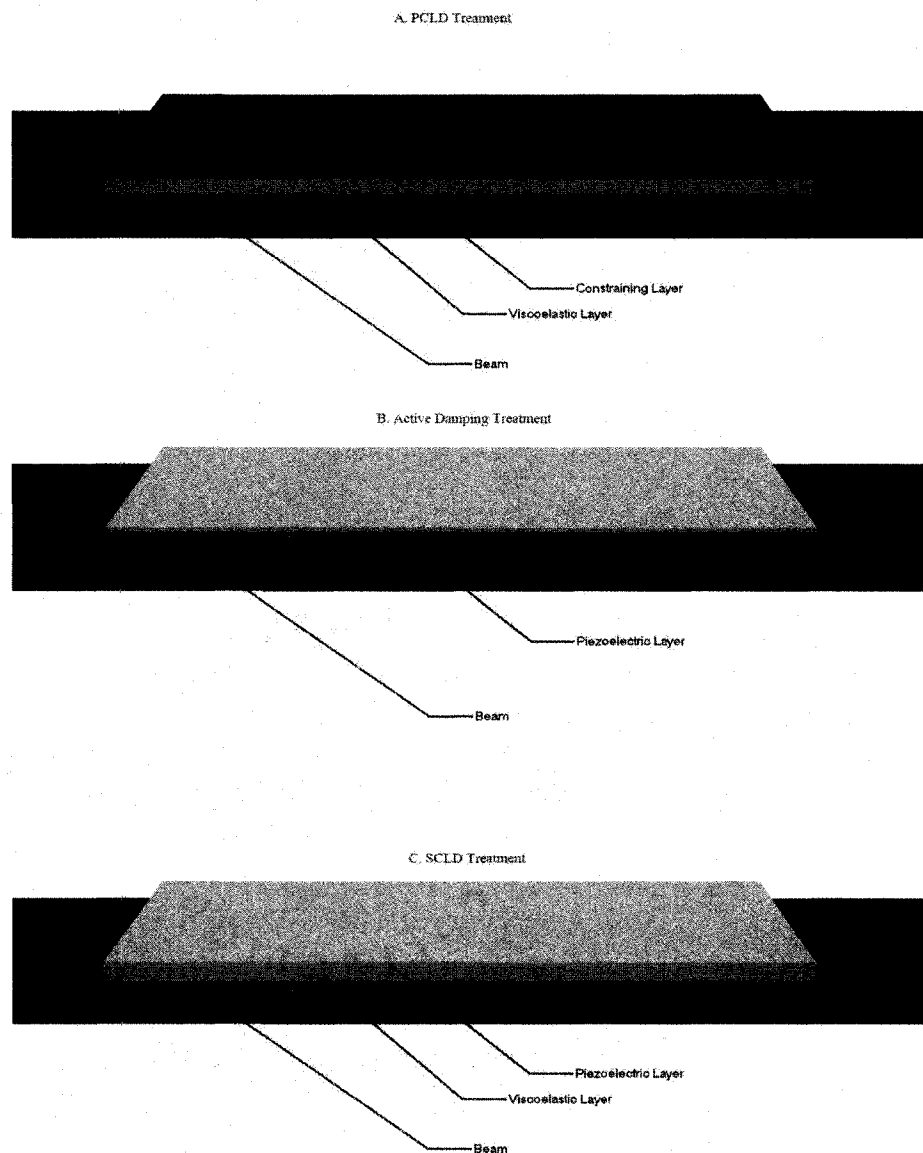


Figure 1: PCLD, Active Damping, and ACLD Treatments

Another currently used method in controlling vibration is through active damping control. One common configuration in active damping control can be found in Figure 1(b). The active damping treatments are constructed by adhering a piezoelectric sensor and actuator to a vibrating structure. A control is then designed to contract and expand the piezoelectric actuator so that it strains the vibrating structure and creates bending moments that counter the bending moments created by the vibration. Active damping has received increasing attention especially in the aerospace industry because of its ability to provide adjustable damping that passive damping cannot [8]. Rao and Sunar [23] have described other recent interests in the area of active damping. Although active damping has more flexibility in terms of controlling the damping parameters, it does have some limitations. Active control can sometimes lead to unstable plant dynamics because of large control gains or non-collocated sensor/actuator configurations [2]. Safety issues also arise because hardware failure or unexpected changes in the environment can cause actuator failure resulting in complete loss of vibration damping.

Active constrained layer damped (ACLD) treatments, also known as smart constrained layer treatments (SCLD) and hybrid damping, have been developed to combine the advantages and reduce the disadvantages of the previously discussed treatments. An ACLD treatment consists of a viscoelastic material sandwiched between a piezoelectric actuator and a host structure (refer to Figure 1(c)). Vibration energy is dissipated by multiple means in this configuration. The piezoelectric actuator now enhances the passive damping by contracting and relaxing the viscoelastic material so that the cyclic shearing is always augmented [2]. Furthermore, the constraining layer develops bending moments that also help to dampen the vibration. ACLD treatments also

have the advantage of having a fail-safe mechanism that purely active control does not. If the structure now loses active damping because of hardware failure, the passive damping remains to provide some vibration control. ACLD treatments have been found to be very effective and have been the focus of current research [2, 8, 13-15, 17, 23-24, 27, 28]. Balamurugan and Narayanan [2] have developed a finite element model for a partially treated ACLD beam using Timoshenko beam theory and the Golla-Hues-McTavish (GHM) method. Trindade *et al* [27,28] have developed additional ACLD finite element models based on mixed Euler and Timoshenko beam elements and have also studied the effects of temperature on ACLD treatments for several VEM dissipation models. Liao and Wang [13] have added edge elements to the finite element model to improve the performance of the ACLD treatment. Liao and Wang [14] have also investigated the influence of viscoelastic material parameters on the passive damping ability, the active action authority, and their combined effect on an ACLD treatment. Lim, Va. Varadan, and Vi. Varadan [15] have compared the performance of the ACLD treatment to purely passive and active damping for multiple configurations. Margaretha [17] has compiled a thorough history of PCLD, active damping, and ACLD treatments and has also investigated the performance of several ACLD configurations.

The control algorithm is an integral component in the application of the ACLD treatment. Han, Rew, and Lee [8] have used a constant gain control (CGC) and a bang-bang control (BBC) in conjunction with a linear quadratic Gaussian (LQG) control algorithm to design a feedback control system. Rao and Veley [24] have implemented a proportional and derivative (PD) control to control an ACLD treated beam. The control scheme most commonly utilized in ACLD structures is state-feedback control with

controller gains determined using linear quadratic regulator (LQR) theory [2,13-15,17, 27-28]. These control methods have been shown to be effective and produce similar vibration damping when compared to each other. In this research, a sliding mode control (SMC) is designed to control the piezoelectric actuator in the ACLD treatment. One major advantage in using a sliding control is that it is a robust control method that allows for both structured and unstructured uncertainties in the development of the model. Structured uncertainties arise from unknown plant parameters such as mass, stiffness, and modulus values while unstructured uncertainties involve simplified model dynamics such as modeling friction as linear, or neglecting structural modes in a reasonably rigid mechanical system [26].

In this research, a SMC was designed to control the actuator in the ACLD treatment. The resulting vibration response was compared to the response for a state-feedback LQR control for various loading conditions. To accurately simulate the vibration response for each control scheme, the beam finite element matrices were derived using Timoshenko beam theory and the energy dissipation in the viscoelastic material was modeled using the Golla-Hues-McTavish (GHM) method. An observer and model order reduction (MOR) were then implemented to achieve a simplified, efficient, and more controllable finite element beam model.

The outline of this thesis follows the control system block diagram in Figure 2 beginning with the “System Model” block and moving clockwise through the control loop. The detailed SIMULINK model *timoshenkobeam\_lqr* can be found in Appendix B.



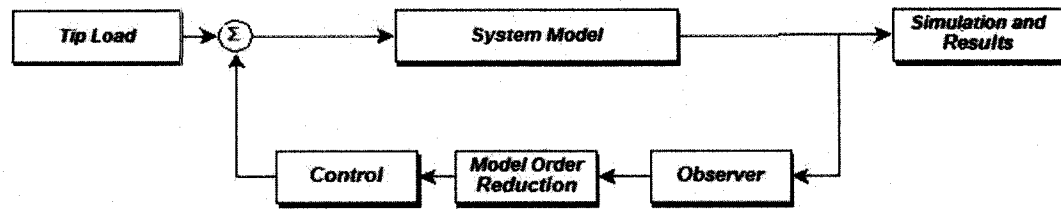


Figure 2: Control System Block Diagram

Chapter two describes the finite elements used to derive the state-space model of the treated beam. Chapter three explains the design of the observer and model order reduction. In chapter four, the control methods are explained and a state-feedback LQR control and a SMC are designed. Chapter five describes and analyzes the numerical results for the different loading conditions studied. Chapter six summarizes the results and conclusions. The introduction of chapters two through five contain a modified version of Figure 2 in which the control system blocks corresponding to the chapter content are highlighted.

## CHAPTER 2

### SYSTEM MODEL

#### 2.1 Introduction to Model Derivation

In this chapter the finite element matrices for the beam and ACLD treatment are described and then assembled to obtain a system mass, damping, and stiffness matrix. The beam finite element matrices are obtained by utilizing Timoshenko beam theory as detailed in Felippa [5], Kwon [11], Reddy [25], and Balamuragan [2]. The element matrices for the ACLD treatment are obtained by following the procedures described in Balamuragan [2] and Liao [13]. The beam and ACLD finite element matrices are then assembled to create a system state space model. The controllability and observability of the state space model is then discussed.

The state space model is necessary to simulate loading conditions and compare the vibration damping achieved with state-feedback LQR control and sliding mode control (SMC). The corresponding block in the control system diagram is highlighted below.

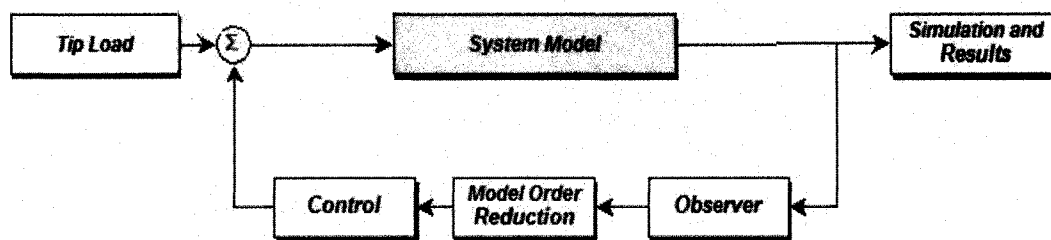


Figure 2: Control System Block Diagram

## 2.2 Model Description

A finite element model was constructed to analytically compare the effectiveness of a SMC and a LQR output feedback control. The model consisted of three sections; the first section was a plain-beam section and was rigidly attached to a surface, the second section was an ACLD treated beam, and the third section was a plain-beam section with a load applied at the tip as show in Figure 3.

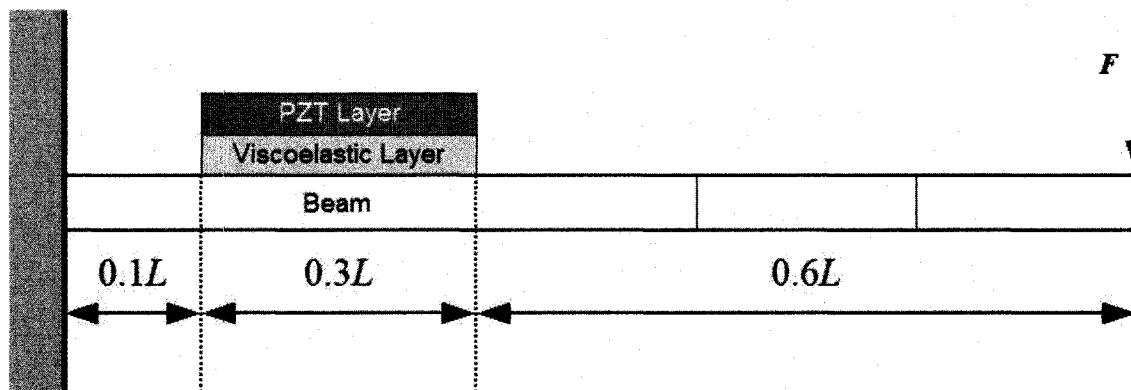


Figure 3: Model Configuration

This configuration has been extensively used in ACLD research and consequently finite element models have been derived by several researchers [2, 13-15, 17, 24]. In this work, the finite elements are obtained by utilizing Timoshenko beam theory methods described in Felippa [5], Kwon [11], Reddy [25], and Balamuragan [2]. The ACLD finite elements

are obtained by following the procedures described in Balamuragan [2] and Liao [10].

The main assumptions made in Balamuragan [2] and Liao [13] are as follows.

- 1) The piezoelectric actuator, viscoelastic material, and beam are perfectly bonded to each other.
- 2) The applied voltage to the piezoelectric actuator is assumed to be constant along its length.
- 3) Linear theories are used to describe elasticity, viscoelasticity, and piezoelectricity.
- 4) The density and thickness are uniform over the beam.

In Balamuragan [2] and Liao [13] the Golla-Hues-Mctavish (GHM) [7] method was used to capture the energy dissipation effects of the viscoelastic layer and the IEEE standard for piezoelectricity was used to model the effects of the actuator.

In Timoshenko beam theory the effects of shear deformation during bending are included in the derivation of the governing differential equations. Including the effect of shear deformation is equivalent to no longer assuming that plane sections before bending remain plane and perpendicular to the longitudinal axis while in bending as in Euler beam theory [25]. As derived in Kwon [11], Rao [19], and Reddy [22], the shear deformation results in two second order differential equations

$$-\frac{\partial}{\partial x} \left( G_b \cdot A_b \cdot k_\alpha \cdot \left[ \phi - \frac{\partial w}{\partial x} \right] \right) + f(x, t) = \rho_b \cdot A_b \cdot \frac{\partial^2 w}{\partial t^2} \quad (1)$$

$$\frac{\partial}{\partial x} \left( E_b \cdot I_b \cdot \frac{\partial \phi}{\partial x} \right) - G_b \cdot A_b \cdot k_\alpha \left( \phi - \frac{\partial w}{\partial x} \right) = \rho_b \cdot I_b \cdot \frac{\partial^2 \phi}{\partial t^2}. \quad (2)$$

A finite element model based on Timoshenko beam theory is necessary because the shear deformations in the beam result in shear deformations in the viscoelastic layer, which

subsequently produce the damping found in the actual system. The resulting degrees of freedom for the system include an axial displacement,  $u$ , a transverse displacement,  $w$ , a rotation angle,  $\theta$ , and a shear angle,  $\phi$ . The degrees of freedom are contained in the nodal displacement vector  $\{q_e\}$

$$\{q_e\} = \{u_1 \quad w_1 \quad \theta_1 \quad \phi_1 \quad u_2 \quad w_2 \quad \theta_2 \quad \phi_2\}^T. \quad (3)$$

Figure 4 shows the degrees of freedom and the lengths of a representative ACLD finite element.

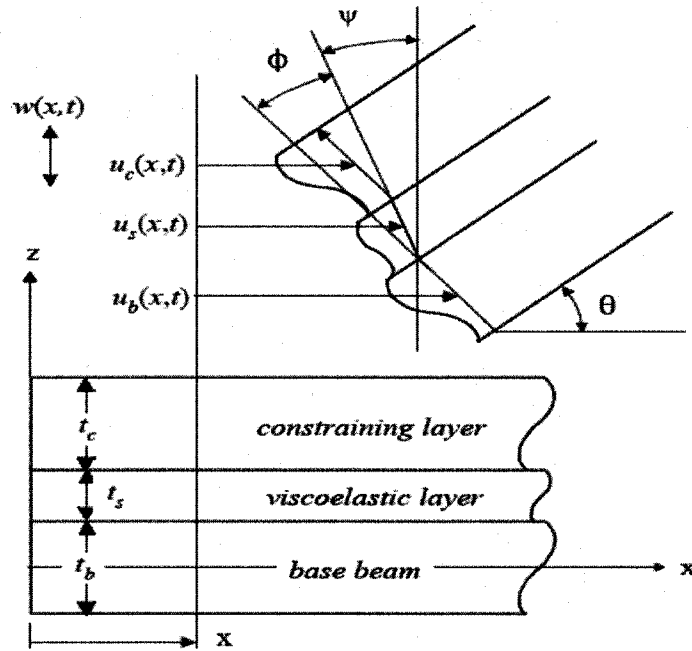


Figure 4: Degrees of Freedom for ACLD Treated Beam (provided by [17])

As described by Felippa [5], Kwon [11], and Reddy [25], the axial displacement, transverse displacement, rotation angle, and shear angle are expressed in terms of nodal displacements by finite element shape functions:

$$u = \{N_u(x)\} \cdot \{q_e\} \quad w = \{N_w(x)\} \cdot \{q_e\} \quad (4) - (5)$$

$$\theta = \{N_\theta(x)\} \cdot \{q_e\} \quad \phi = \{N_\phi(x)\} \cdot \{q_e\} \quad (6) - (7)$$

As detailed by Felippa [5], the Timoshenko beam theory assumptions facilitate the use of linear interpolation functions:

$$\{N_u(x)\} = \begin{bmatrix} 1 - \frac{x}{L_e} & 0 & 0 & 0 & \frac{x}{L_e} & 0 & 0 & 0 \end{bmatrix} \quad (8)$$

$$\{N_w(x)\} = \begin{bmatrix} 0 & 1 - \frac{x}{L_e} & 0 & 0 & 0 & \frac{x}{L_e} & 0 & 0 \end{bmatrix} \quad (9)$$

$$\{N_\theta(x)\} = \begin{bmatrix} 0 & 0 & 1 - \frac{x}{L_e} & 0 & 0 & 0 & \frac{x}{L_e} & 0 \end{bmatrix} \quad (10)$$

$$\{N_\phi(x)\} = \begin{bmatrix} 0 & 0 & 0 & 1 - \frac{x}{L_e} & 0 & 0 & 0 & \frac{x}{L_e} \end{bmatrix}. \quad (11)$$

The procedures developed in [2] and [13] are used to evaluate the potential and kinetic energy equations based on Hamilton's principle and obtain the stiffness and mass matrices.

### 2.3 Beam Layer Finite Element Matrices

As described in Balamuragan [2] and Liao [13], evaluation of the potential energy due to bending yields the stiffness matrix  $[K_{wbb}]$ .

$$\begin{aligned} \frac{1}{2} \cdot E_b \cdot I_b \cdot \int_0^{L_e} \left( \frac{\partial \theta}{\partial x} \right)^2 dx &= \frac{1}{2} \cdot E_b \cdot I_b \cdot \{q_e\}^T \cdot \int_0^{L_e} (\{N_\theta'\}^T \cdot \{N_\theta'\}) dx \cdot \{q_e\} = \\ &= \frac{1}{2} \cdot \{q_e\}^T \cdot [K_{wbb}] \cdot \{q_e\} \end{aligned} \quad (12)$$

Evaluation of the potential energy due to extension yields the stiffness matrix  $[K_{ub}]$ :

$$\frac{1}{2} \cdot E_b \cdot A_b \cdot \int_0^{L_e} \left( \frac{\partial u_b}{\partial x} \right)^2 dx = \frac{1}{2} \cdot E_b \cdot A_b \cdot \{q_e\}^T \cdot \int_0^{L_e} (\{N_u'\}^T \cdot \{N_u'\}) dx \cdot \{q_e\} =$$

$$\frac{1}{2} \cdot \{q_e\}^T \cdot [K_{ub}] \cdot \{q_e\}, \quad (13)$$

where  $E_b$ ,  $I_b$ , and  $A_b$  are the elastic modulus, moment of inertia, and cross sectional area of the beam, respectively. The prime symbols over the shape functions denote the first derivative with respect to  $x$  (i.e.  $\{N_u'\} = \frac{d\{N_u\}}{dx}$ ). Evaluation of the potential energy due to transverse shear energy leads to a mathematical phenomenon known as shear locking. Shear locking occurs when the quadrilateral linear elements used in Timoshenko beam theory fail to accurately model the curvature in the beam. As a beam gets thinner, the linear elements introduce shear stresses not present in the actual beam and cause the beam to reach equilibrium with smaller displacements [16]. One common method to circumvent shear locking is to use consistent interpolation for  $w$  and  $\phi$  such that  $dw/dx$  and  $\phi$  are polynomials of the same order. Another method to avoid shear locking is to use equal interpolation for  $w$  and  $\phi$ , but evaluate the shear energy with a polynomial interpolation of  $\phi$  that is one order lower. The latter method is referred to as reduced integration and the details of both these methods can be found in Reddy [25]. Following the method detailed in Reddy [25], reduced integration is used in the evaluation of the potential energy due to transverse shear:

$$\begin{aligned} & \frac{1}{2} \cdot G_b \cdot A_b \cdot k_\alpha \cdot \int_0^{l_e} \left( \theta - \frac{\partial w}{\partial x} \right)^2 dx = \\ & \frac{1}{2} \cdot G_b \cdot A_b \cdot k_\alpha \cdot \{q_e\}^T \cdot \int_0^{l_e} \left( \begin{bmatrix} \{N_\theta\} \\ \{N_w'\} \end{bmatrix}^T \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix}^T \cdot \begin{bmatrix} \{N_\theta\} \\ \{N_w'\} \end{bmatrix} \right) dx \cdot \{q_e\} = \\ & \frac{1}{2} \cdot \{q_e\}^T \cdot [K_{wbs}] \cdot \{q_e\} \end{aligned} \quad (14)$$

where  $G_b$  is the modulus of rigidity and  $k$  is the shear correction factor (usually equal to 5/6). The kinetic energy of the beam associated with transverse and rotary inertia yields  $[M_{wb}]$ :

$$\begin{aligned} \frac{1}{2} \cdot \rho_b \int_0^{L_e} A_b \cdot \left( \frac{\partial w_b}{\partial t} \right)^2 + I_b \cdot \left( \frac{\partial \theta}{\partial t} \right)^2 dx = \frac{1}{2} \cdot E_b \cdot A_b \cdot \{\dot{q}_e\}^T \cdot \int_0^{L_e} (\{N_u\}^T \cdot \{N_u\}) dx \cdot \dot{q}_e^T = \\ \frac{1}{2} \cdot \{\dot{q}_e\}^T \cdot [M_{wb}] \cdot \{\dot{q}_e\}. \end{aligned} \quad (15)$$

The dot over the nodal displacement vector denotes the first partial derivative with respect to time (i.e.  $\{\dot{q}_e\} = \frac{\partial \{q_e\}}{\partial t}$ ). Evaluation of the kinetic energy due to axial motion yields  $[M_{ub}]$ .

$$\begin{aligned} \frac{1}{2} \cdot \rho_b \cdot A_b \cdot \int_0^{L_e} \left( \frac{\partial u_b}{\partial t} \right)^2 dx = \frac{1}{2} \rho_b \cdot A_b \cdot \{\dot{q}_e\}^T \cdot \int_0^{L_e} \{N_u\}^T \cdot \{N_u\} \cdot dx \cdot \{\dot{q}_e\} = \\ \frac{1}{2} \cdot \{\dot{q}_e\}^T \cdot [M_{ub}] \cdot \{\dot{q}_e\} \end{aligned} \quad (16)$$

## 2.4 Viscoelastic Layer Finite Element Matrices

As outlined in Balamuragan [2] and Liao [10], evaluation of the kinetic energy due to transverse and axial motion yields the mass matrices  $[M_{ws}]$  and  $[M_{us}]$ , respectively.

$$\begin{aligned} \frac{1}{2} \cdot \rho_s \cdot \int_0^{L_e} \left( A_s \cdot \left( \frac{\partial w}{\partial t} \right)^2 + I_s \left( \frac{\partial \theta}{\partial t} \right)^2 \right) dx = \\ \frac{1}{2} \cdot \rho_s \cdot \{\dot{q}_e\}^T \cdot \int_0^{L_e} \left( A_s \cdot \{N_w\}^T \cdot \{N_w\} + I_s \cdot \{N_\theta\}^T \cdot \{N_\theta\} \right) dx \cdot \{\dot{q}_e\} = \\ \frac{1}{2} \cdot \{\dot{q}_e\}^T \cdot [M_{ws}] \cdot \{\dot{q}_e\} \end{aligned} \quad (17)$$

and



$$\begin{aligned} & \frac{1}{2} \cdot \rho_s \cdot \int_0^{L_e} A_s \cdot \left( \frac{\partial u_s}{\partial t} \right)^2 dx = \\ & \frac{1}{2} \cdot \rho_s \cdot A_s \cdot \{\dot{q}_e\}^T \cdot \int_0^{L_e} \begin{pmatrix} \{N_u\} \\ \{N_\theta\} \\ \{N_\phi\} \end{pmatrix}^T \cdot \begin{bmatrix} 1 \\ -(tb+ts)/2 \\ ts/2 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -(tb+ts)/2 \\ ts/2 \end{bmatrix}^T \begin{pmatrix} \{N_u\} \\ \{N_\theta\} \\ \{N_\phi\} \end{pmatrix} dx \cdot \{\dot{q}_e\} = \\ & \frac{1}{2} \cdot \{\dot{q}_e\}^T \cdot [M_{us}] \cdot \{\dot{q}_e\} \end{aligned} \quad (18)$$

where  $\rho_s$ ,  $I_s$ , and  $A_s$  are the density, moment of inertia, and area of the cross-section of the viscoelastic layer. The constitutive equation for a viscoelastic material can be described by the Stieltjes convolution [2, 7, 13, 27]

$$\tau(x, t) = G \circ \phi \equiv \int_{-\infty}^t (G(t-\tau) \frac{\partial \phi(x, \tau)}{\partial \tau}) d\tau. \quad (19)$$

$G(t)$  is the relaxation function of the viscoelastic material and represents energy loss from the material. The relaxation function can be evaluated in the time domain such as with a Prony series representation [19] or in the Laplace domain as with the GHM method [7]. Other methods used to model the dissipation effects of the VEM include the Anelastic Displacement Field (ADF) model and the Model Strain Energy (MSE) model [27]. In the GHM method, the material modulus is represented as a series of mini-oscillator terms [7]:

$$s \cdot G(s) = \kappa \left[ 1 + \sum_{r=1}^n \alpha_r \cdot \frac{s^2 + 2\zeta_r \cdot \omega_r \cdot s}{s^2 + 2\zeta_r \cdot \omega_r \cdot s + \omega_r^2} \right] \quad (20)$$

where  $\kappa$  is the equilibrium value of the modulus. The constants  $\alpha_r$ ,  $\zeta_r$ , and  $\omega_r$  govern the shape of the modulus function over the complex  $s$ -domain. Following the procedure used in Balamurugan [2] and Liao [13], two dissipation coordinates are obtained,

$$z = \{N_z(x)\} \cdot \{z\} = \begin{bmatrix} 1 - \frac{x}{Le} & \frac{x}{Le} \end{bmatrix} \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \quad (21)$$

Evaluation of the virtual work of the viscoelastic layer yields the mass, damping, and stiffness matrices:

$$\begin{aligned} \delta W_s = & -A_s \cdot \{\delta q_e\}^T \cdot \left( (\kappa + \alpha \cdot \kappa) \int_0^{Le} (\{N_\phi\}^T \cdot \{N_\phi\}) dx \cdot \{q_e\} - \alpha \cdot \kappa \int_0^{Le} (\{N_\phi\}^T \cdot \{N_z\}) dx \cdot \{z\} \right) = \\ & -\{\delta q_e\}^T \cdot [K_s] \cdot \{q_e\} + \{\delta q_e\}^T \cdot [K_{qz}] \cdot \{z\} \end{aligned} \quad (22)$$

where  $[K_s]$ ,  $[K_{qz}]$ , and  $[K_{zq}]$  were equal to

$$[K_s] = A_s \cdot (\kappa + \alpha \cdot \kappa) \cdot \int_0^{Le} (\{N_\phi\}^T \cdot \{N_\phi\}) dx \quad (23)$$

$$[K_{qz}] = A_s \cdot \alpha \cdot \kappa \cdot \int_0^{Le} (\{N_\phi\}^T \cdot \{N_z\}) dx \quad (24)$$

$$[K_{zq}] = [K_{qz}]^T \quad (26)$$

and

$$[K_z] = A_s \cdot \alpha \cdot \kappa \cdot \int_0^{Le} (\{N_z\}^T \cdot \{N_z\}) dx \quad (27)$$

$$[M_z] = A_s \cdot \alpha \cdot \kappa \cdot \frac{1}{\omega^2} \cdot \int_0^{Le} (\{N_z\}^T \cdot \{N_z\}) dx \quad (28)$$

$$[C_z] = A_s \cdot \alpha \cdot \kappa \cdot \frac{2 \cdot \xi}{\omega} \cdot \int_0^{Le} (\{N_z\}^T \cdot \{N_z\}) dx \quad (29)$$

## 2.5 Piezoelectric Layer Finite Element Matrices

As described by Balamuragan [2] and others [8, 13-15], for one-dimensional structures with uni-axial loading, the constitutive equation of a piezoelectric material can be written as

$$\begin{bmatrix} \varepsilon \\ D \end{bmatrix} = \begin{bmatrix} S_{11}^E & d_{31} \\ d_{31} & \varepsilon_{33}^T \end{bmatrix} \cdot \begin{bmatrix} \tau \\ E \end{bmatrix}. \quad (30)$$

$D$  is the electrical displacement,  $E$  is the electric field ( $V(t)/t_c$ ),  $\gamma$  is the mechanical strain in the x direction, and  $\tau$  is the mechanical stress in the x direction.  $S_{11}^E$  is the elastic compliance,  $\varepsilon_{33}^T$  is the dielectric constant, and  $d_{31}$  is the piezoelectric constant. From the previous system of equations, the stress is related to the strain by

$$\tau = \frac{1}{S_{11}^E} \cdot (\varepsilon - d_{31} \cdot E) = Ec \cdot (\varepsilon - d_{31} \cdot E). \quad (31)$$

The virtual work done by the induced strain in the actuator is

$$\begin{aligned} \delta W &= \int_0^{L_e} \left( E_c \cdot d_{31} \cdot b \cdot V(t) \cdot \delta \left( \frac{du_c}{dx} \right) \right) dx = \\ &E_c \cdot d_{31} \cdot b \cdot V(t) \cdot \delta \{q_e\}^T \cdot \begin{bmatrix} -1 & 0 & h & -t_s & 1 & 0 & -h & t_s \end{bmatrix} = \\ &\delta \{q_e\}^T \cdot \{Pc\}_e \cdot V(t). \end{aligned} \quad (32)$$

$V(t)$  is the applied voltage to the constraining piezoelectric layer and  $\{Pc\}_e$  is the resulting elemental piezoelectric force vector.

As in the derivation of the mass and stiffness matrices for the Timoshenko beam, energy equations are evaluated to generate the mass and stiffness matrices for the piezoelectric layer. Evaluation of the potential energy due to bending yields

$$[K_{wcb}] = E_c \cdot I_c \cdot \int_0^{L_e} (\{N_\theta'\}^T \{N_\theta'\}) dx. \quad (33)$$

The potential energy due to extension results in

$$[K_{uc}] = E_c \cdot A_c \cdot \int_0^{L_e} \begin{bmatrix} \{N_u'\} \\ \{N_\theta'\} \\ \{N_\phi'\} \end{bmatrix}^T \cdot \begin{bmatrix} 1 \\ -h \\ t_s \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -h \\ t_s \end{bmatrix}^T \begin{bmatrix} \{N_u'\} \\ \{N_\theta'\} \\ \{N_\phi'\} \end{bmatrix} \cdot dx. \quad (34)$$

$E_c$ ,  $A_c$ , and  $I_c$ , are the elastic modulus, cross sectional area, and moment of inertia for piezoelectric layer, respectively. As in the beam element, the shear energy term for the piezoelectric layer is evaluated using reduced integration. The potential energy due to transverse shear yields the stiffness matrix

$$[K_{wcs}] = G_c \cdot A_c \cdot k_\alpha \cdot \int_0^{L_e} \left( \begin{bmatrix} \{N_\theta\} \\ \{N_w'\} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} \{N_\theta\} \\ \{N_w'\} \end{bmatrix} \right) dx. \quad (35)$$

$G_c$  is the modulus of rigidity and  $k_\alpha$  is the shear correction factor. The kinetic energy of the piezoelectric layer due to transverse motion and axial motion yield the mass matrices  $[M_{wc}]$  and  $[M_{uc}]$ , respectively,

$$[M_{wc}] = \rho_c \cdot \int_0^{L_e} \left( A_c \cdot \{N_w\}^T \cdot \{N_w\} + I_c \cdot \{N_\theta\}^T \cdot \{N_\theta\} \right) dx \quad (36)$$

$$[M_{uc}] = \rho_c \cdot A_c \cdot \int_0^{L_e} \left( \begin{bmatrix} \{N_u\} \\ \{N_\theta\} \\ \{N_\phi\} \end{bmatrix}^T \cdot \begin{bmatrix} 1 \\ -h \\ t_s \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -h \\ t_s \end{bmatrix}^T \cdot \begin{bmatrix} \{N_u\} \\ \{N_\theta\} \\ \{N_\phi\} \end{bmatrix} \right) dx. \quad (37)$$

$\rho_c$  is the density of the piezoelectric layer.

The finite element matrices for the ACLD section were evaluated in the MATLAB file *BalamuruganNarayananFEM2.m*<sup>1</sup>, the file *TimoshenkoFEM2.m*<sup>1</sup> was used to evaluate the finite element matrices for the plain beam sections, and *SystemMatrices03.m*<sup>1</sup> was used to assemble the matrices into a single system.

## 2.6 Model Assembly

The mass and stiffness matrices for the plain-beam sections only depend on the original physical coordinates,  $\{q_e\}$ , and can be written as

<sup>1</sup> Provided by Thesis advisor Dr. Javier Kypuros

$$[M_b] = [M_{wb}] + [M_{ub}] \quad (38)$$

$$[K_b] = [K_{wbb}] + [K_{ub}] + [K_{wbs}]. \quad (39)$$

The treated section of the beam consists of plain-beam, viscoelastic, and piezoelectric finite element matrices. The finite element matrices for the ACLD section corresponding to  $\{q_e\}$  are

$$[M]_e = [M_b] + [M_{wc}] + [M_{ws}] + [M_{uc}] + [M_{us}] \quad (40)$$

$$[K]_e = [K_b] + [K_{wbs}] + [K_{wcb}] + [K_{uc}] + [K_{us}] + [K_s]. \quad (41)$$

The mass and stiffness matrices corresponding to  $\{q_e\}$  and the dissipation coordinates  $\{z\}$  are  $[K_{qz}]$  and  $[K_{zq}]$ . The matrices corresponding only to the dissipation coordinates  $\{z\}$  are  $[M_z]$ ,  $[C_z]$ , and  $[K_z]$ , where  $[C_z]$  is the viscoelastic damping matrix. In addition to the viscoelastic damping, structural damping is also included in the form of Rayleigh damping,

$$[C_{ea}] = \hat{a} \cdot [M_b] + \hat{b} \cdot [K_b]. \quad (42)$$

$\hat{a}$  and  $\hat{b}$  are generally obtained experimentally, but in this work were obtained from published values [2,13]. The elemental equations of motion for an ACLD treated section are therefore

$$\begin{bmatrix} [M]_e & [0]_{8 \times 2} \\ [0]_{2 \times 8} & [M]_z \end{bmatrix} \cdot \begin{bmatrix} \{\ddot{q}_e\} \\ \{\ddot{z}_e\} \end{bmatrix} + \begin{bmatrix} [C_{ea}] & [0]_{8 \times 2} \\ [0]_{2 \times 8} & [C_z] \end{bmatrix} \cdot \begin{bmatrix} \{\dot{q}_e\} \\ \{\dot{z}_e\} \end{bmatrix} + \begin{bmatrix} [K_e] & -[K_{qz}] \\ -[K_{zq}] & [K_z] \end{bmatrix} \cdot \begin{bmatrix} \{q_e\} \\ \{z_e\} \end{bmatrix} = \begin{bmatrix} \{Pc_e\} \cdot \{V(t)\}_e + \{fd\}_e \\ [0] \end{bmatrix}. \quad (43)$$

The resulting mass, damping, and stiffness matrices for the treated section of the beam are 10x10 matrices. The equations of motion for the entire beam can now be written as

$$[M] \cdot \{\ddot{q}\} + [C] \cdot \{\dot{q}\} + [K] \cdot \{q\} = \{Pc\} \cdot \{V(t)\} + \{Fd\}. \quad (44)$$

The dissipation coordinates,  $\{z\}$ , are included in the vector  $\{q\}$ , the total system mass matrix is  $[M]$ , the system damping matrix is  $[C]$ , and the system stiffness matrix is  $[K]$ .

To formulate the system finite element model, the beam was discretized into five elements of varying lengths, one finite element before the ACLD treatment that was 10% of the total length, one finite element for the ACLD treated section that was 30% of the total length, and three elements after the ACLD section that were cumulatively 60% of the total length as shown in Figure 3. The total number of finite elements and the lengths of each of the finite elements were chosen so that the most important dynamics were captured and so as to limit the time of numerical simulation. The first section had few dynamics of interest so only one element was used. The length of the first element was also minimized so that the ACLD section would be placed near the area with the largest moments. The length of the ACLD section was kept constant at 30% percent of the total length of the beam as in [2] for validation purposes. The final section of the beam had the greatest displacements and velocities and required more elements to be modeled accurately. The MATLAB routine FEASMBL [11] (refer to Appendix A) was utilized to assemble the three layers of the ACLD section and was also used to assemble the five elements that constitute the entire beam.

## 2.7 State Space Representation

After the system mass, damping, and stiffness matrices were obtained, the first three degrees of freedom were set to zero to impose the rigid boundary condition at the cantilevered section of the beam. A state space representation was then created from the reduced system matrices:

$$\{\dot{x}\} = [A] \cdot \{x\} + [B] \cdot V(t) + [\hat{B}] \quad (45)$$

$$\{y\} = [\hat{C}] \cdot \{x\}. \quad (46)$$

$\{\dot{x}\}$  was a 38x1 state vector for the assembled system,  $[A]$  a the 38x38 state matrix,  $[B]$  was a 38x1 control force matrix,  $[\hat{B}]$  was a 38x1 force matrix due to the tip load, and  $[\hat{C}]$  was a 1x38 output matrix. The values for the state space matrices were

$$[A] = \begin{bmatrix} [0]_{19 \times 19} & [I]_{19 \times 19} \\ -[M]^{-1} \cdot [K] & -[M]^{-1} \cdot [\hat{C}] \end{bmatrix}, \quad (47)$$

$$[B] = \begin{bmatrix} [0]_{19 \times 1} \\ [M]^{-1} \cdot \{Pc\} \end{bmatrix}, \quad (48)$$

$$[\hat{B}] = \begin{bmatrix} [0]_{19 \times 1} \\ [M]^{-1} \cdot [Fd] \end{bmatrix}. \quad (49)$$

$[\hat{C}]$  was a column vector of zeros with the exception of a one in the column that represented the tip deflection. In this case, the state space formulation resulted in the first 19 rows of the state vector to correspond to the generalized displacements of the system degrees of freedom. The degrees of freedom for the plain beam section were ordered with the axial displacement,  $u$ , first, the transverse displacement,  $w$ , second, and the rotation angle,  $\phi$ , third. Therefore, the eighteenth row of the state vector corresponded to the transverse displacement at the tip of the beam. The eighteenth column of the output matrix  $[\hat{C}]$  was set equal to one so that equation (46) produced the tip deflection.

## 2.8 Controllability and Observability

A state space model is state controllable at time  $t_0$  if it is possible by means of an unconstrained control vector to transfer the model from any initial state  $\{x(t_0)\}$  to any

other state in a finite interval of time [18]. The condition for complete state controllability is that the  $n \times n$  controllability matrix

$$[[B][A] \cdot [B] \dots [A]^{n-1} \cdot [B]]$$

is of full rank [18]. A system that is not completely state controllable contains states that are not affected by the input and are physically disconnected from the main system (i.e. states associated with dissipation coordinates) [18, 29, 30]. It is necessary to check the controllability of the state space model because if the system is not completely state controllable, it will not be possible to design a controller by arbitrarily selecting eigenvalues (pole placement) [18].

A state space model is completely observable if every state  $\{x(t)\}$  can be determined from the observation of the output  $\{y(t)\}$  over a finite interval of time [18]. The condition for complete state observability is that the  $(n \cdot m) \times n$  observation matrix

$$[[\hat{C}][\hat{C}] \cdot [A] \dots [C] \cdot [A]^{n-1}]^T$$

is of rank  $n$  [18]. It is necessary to check the observability of a model because it will not be possible to design an observer by arbitrarily selecting eigenvalues (pole placement) if the model is not completely state observable [18].

The state space model derived in section 2.7 was neither completely state controllable nor observable. Although it was not possible to design a controller or observer by arbitrarily selecting eigenvalues, it was still possible to design the controller and observer by utilizing an optimization approach. The optimization method for designing an observer and controller is outlined in chapters three and four, respectively.



## CHAPTER 3

### MODEL MODIFICATION

#### 3.1 Introduction to Model Modification

In this chapter the system state space model is modified to obtain a simplified, efficient, and more controllable model. To achieve this modified model, an observer is designed using an optimization approach [12] and model order reduction (MOR) is performed using a balanced realization method based on Schur decomposition [3]. The corresponding blocks in the control system diagram are highlighted below.

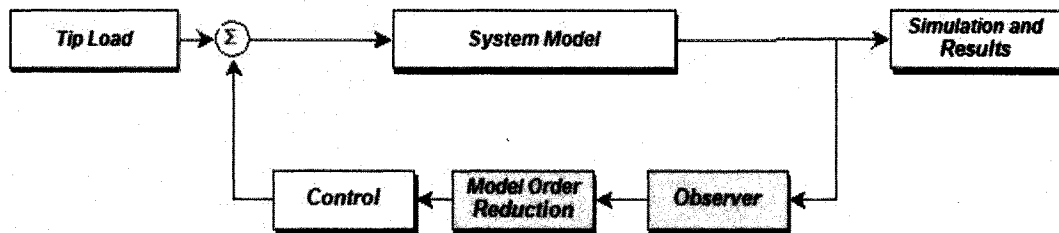


Figure 2: Control System Block Diagram

#### 3.2 Observer Design

In most dynamic systems rarely are all states available for feedback purposes. A state observer is needed to estimate the states  $\{x(t)\}$  based on the measurement of the available outputs  $\{y(t)\}$ . In this work, the tip deflection was used to estimate the 38 states of the state vector  $\{x(t)\}$ .

As detailed in [9, 12, 18], an observer is a dynamical system described by

$$\{\dot{x}_{ob}\} = [A] \cdot \{x_{ob}\} + [B] \cdot V(t) + [L] \cdot (y - [\hat{C}] \cdot \{x_{ob}\}). \quad (50)$$

$\{x_{ob}\}$  is the observed (estimated) state vector,  $[\hat{C}] \cdot \{x_{ob}\}$  is the estimated output, and  $[L]$  is the observer design matrix. The error between the actual states and the estimated states is defined as  $\{e\} = \{x\} - \{x_{ob}\}$ . The error dynamics are therefore

$$\begin{aligned} \{\dot{x}\} - \{\dot{x}_{ob}\} &= [A] \cdot \{x\} - [A] \cdot \{x_{ob}\} - [L] \cdot ([\hat{C}] \cdot \{x\} - [\hat{C}] \cdot \{x_{ob}\}) = \\ \dot{e} &= ([A] - L \cdot [\hat{C}]) \cdot e. \end{aligned} \quad (51)$$

If the error between the actual states and the estimated states is to vanish with time for any initial value of  $\{x_{ob}\}$ , then  $[A] - [L] \cdot [\hat{C}]$  should be asymptotically stable. The observer design problem therefore reduces to finding an observer gain matrix,  $[L]$ , such that  $[A] - [L] \cdot [\hat{C}]$  is asymptotically stable (i.e. eigenvalues have negative real parts) and the error vanishes suitably quickly.

Several pole placement methods exist to determine  $[L]$  such as the transformation approach, direct substitution approach, and Ackermann's formula [15]. Pole placement methods were not applicable in this work because the model was not completely state observable. An optimization method was therefore utilized to determine observer gain matrix  $[L]$ . In the optimization approach,  $[L]$  is determined so that  $u = -[L] \cdot \{x(t)\}$  minimizes the performance index

$$J = \int_0^{\infty} (\{x\}^T \cdot [Q_{obs}] \cdot \{x\} + \{u\}^T \cdot [R_{obs}] \cdot \{u\}) dt \quad (52)$$

where  $[Q_{obs}]$  is a positive-definite (or positive semi-definite) Hermitian or real symmetric matrix and  $[R_{obs}]$  is a positive-definite Hermitian or real symmetric matrix. To use the optimization framework, the transpose of  $[A] - [L] \cdot [\hat{C}]$  is first needed,

$$([A] - [L] \cdot [\hat{C}])^T = [A]^T - [\hat{C}]^T \cdot [L]^T. \quad (53)$$

$([A] - [L] \cdot [\hat{C}])^T$  and  $[A]^T - [\hat{C}]^T \cdot [L]^T$  have the same poles and it can be proven [18] that if  $([A] - [L] \cdot [\hat{C}])^T$  is stable, then  $[L] = ([R]^{-1} \cdot [\hat{C}] \cdot [P])^T$ , where  $[P]$  is the positive-definite matrix that solves the algebraic Riccati equation (ARE) [9],

$$[A] \cdot [P] + [P] \cdot [A]^T - [P] \cdot [\hat{C}]^T \cdot [R]^{-1} \cdot [\hat{C}] \cdot [P] + [Q]. \quad (54)$$

The observer was designed in the MATLAB m-file *MixedBeamModel.m* by using the MATLAB command `care` to return the values of the positive definite matrix  $[P]$ , the observer gain matrix  $[L]$ , and the vector of eigenvalues  $\{\lambda\}$  for the matrix  $[[A] - [L] \cdot [\hat{C}]]$ . The required inputs of the `care` command were the transpose of the state matrix  $[A]$ , the transpose of the output matrix  $[\hat{C}]$ , and the optimization matrices  $[Q_{obs}]$  and  $[R_{obs}]$ .  $[Q_{obs}]$  was chosen to be a  $38 \times 38$  identity matrix and  $[R_{obs}]$  was set equal to 1. The vector of eigenvalues  $\{\lambda\}$  contained elements with negative real parts and therefore the observer designed was dynamically stable and reduced the error between the actual and estimated states. The observer design matrix  $[L]$  and the state space matrices  $[A]$ ,  $[B]$ , and  $[\hat{C}]$  were input into the “Observer” block of the SIMULINK file *timoshenkobeam\_smc* (refer to Appendix B). The “Observer” block contained equation (50) in block diagram form and was used to implement the observer in the simulation.

### 3.3 Model Order Reduction

The purpose of model order reduction (MOR) is to approximate input-output behavior of high order systems with lower order models. MOR also serves to increase the controllability of the state space model because the lower order models are obtained

by removing uncontrollable states. In the case of the ACLD treated beam, 38 states modeled the input-output behavior between the tip load and tip deflection. MOR was utilized to approximate this relationship with fewer states, reduce simulation time, and increase the controllability of the model.

One of the most common approaches to MOR is to use projection methods such as the Krylov-subspace [26,27], truncated-balanced realization [3], and proper orthogonal decomposition (POD) methods [26, 27]. These methods reduce high order models to lower order models by projecting the high order models onto a lower subspace through transformation matrices. The transformation matrices are determined by different methods for each of the projection schemes. A balanced realization method based on Schur decomposition [3] was used in this work to reduce the beam model from  $n$  states to  $k$  states where  $k \ll n$ . In balanced realization methods, the projection subspaces are the dominant eigenspaces of the controllability and observability grammians  $[P]$  and  $[Q]$ . For a linear time invariant system, the controllability and observability grammians can be found by solving the Lyapunov equations,

$$[P] \cdot [A]^T + [A] \cdot [P] + [B] \cdot [B]^T = 0 \quad (55)$$

$$[Q] \cdot [A] + [A]^T \cdot [Q] + [\hat{C}]^T \cdot [\hat{C}] = 0. \quad (56)$$

The eigenvectors can then be found by placing the product grammian  $[P] \cdot [Q]$  into Schur form,  $[V] \cdot ([P] \cdot [Q]) \cdot [V]^T$ . The square root of each eigenvalue is the Hankel singular value for that state. The Hankel singular values denote the relative importance of each state and describe the effectiveness of the inputs translating to outputs [26]. The  $n$  states with the relatively largest Hankel singular values can therefore be used to determine the size of the reduced order model [26,27]. Orthogonal real transformations

$[V_A]$  and  $[V_D]$  are then calculated so that the Schur forms are ordered in ascending and descending order, respectively:

$$[V_A]^T \cdot ([P] \cdot [Q]) \cdot [V_A] = \begin{bmatrix} \lambda_1 & \dots & \dots \\ 0 & \dots & \dots \\ 0 & 0 & \lambda_n \end{bmatrix} \quad (57)$$

$$[V_D]^T \cdot ([P] \cdot [Q]) \cdot [V_D] = \begin{bmatrix} \lambda_n & \dots & \dots \\ 0 & \dots & \dots \\ 0 & 0 & \lambda_1 \end{bmatrix} \quad (58)$$

$[V_A]$  and  $[V_D]$  are then partitioned so that

$$[V_A] = \begin{bmatrix} [V_{R,SMALL}]_{n \times (n-k)} & [V_{L,BIG}]_{n \times k} \end{bmatrix} \quad (59)$$

$$[V_D] = \begin{bmatrix} [V_{R,BIG}]_{n \times k} & [V_{L,SMALL}]_{n \times (n-k)} \end{bmatrix}. \quad (60)$$

$[V_{R,BIG}]$  and  $[V_{R,SMALL}]$  form orthonormal bases for the right eigenspaces of

$[P] \cdot [Q]$  associated with the big eigenvalues  $\{\sigma_1, \dots, \sigma_k^2\}$  and the small eigenvalues  $\{\sigma_{k+1}, \dots, \sigma_n^2\}$ .

The singular value decomposition of  $[V_{L,BIG}]^T \cdot [V_{R,BIG}]$  is taken to obtain

$$[V_{L,BIG}]^T \cdot [V_{R,BIG}] = [E_{BIG}] = [U_{E,BIG}] \cdot ([\Sigma_{E,BIG}]) \cdot [V_{E,BIG}]^T. \quad (61)$$

The left and right transformation matrices can be proven to be [3]

$$[S_{L,BIG}]_{n \times k} = [V_{L,BIG}] \cdot [U_{E,BIG}] \cdot [\Sigma_{E,BIG}]^{-1/2} \quad (62)$$

$$[S_{R,BIG}]_{n \times k} = [V_{R,BIG}] \cdot [V_{E,BIG}] \cdot [\Sigma_{E,BIG}]^{-1/2}. \quad (63)$$

The reduced system matrices are therefore

$$\begin{bmatrix} [A_r] & [B_r] \\ [C_r] & [D_r] \end{bmatrix} = \begin{bmatrix} [S_{L,BIG}]^T \cdot [A_r] \cdot [S_{R,BIG}] & [S_{L,BIG}]^T \cdot [B] \\ [\hat{C}] \cdot [S_{R,BIG}] & [D] \end{bmatrix}. \quad (64)$$

MOR was performed by utilizing a modified version of the `schurmr` command in the MATLAB m-file *MixedBeamModel* (refer to Appendix A). The modified command, `schurmrJAK`<sup>2</sup>, required the inputs `sys` and the desired size of the reduced order model. The variable `sys` was the original state space model and was obtained by using the MATLAB command `ss` ( $sys = ss(A, B, \hat{C}, D)$ ). If `sys` is the only input into `schurmrJAK`, the function will plot the Hankel singular values (refer to Figure 5) and prompt for the size of the reduced order model. As previously discussed, the  $n$  states with the relatively largest Hankel singular values will effectively describe the input-output behavior of the model and therefore a plot of the system Hankel singular values can be used to determine appropriate size of the reduced order model. In Figure 5, the first six Hankel singular values are relatively much larger than the rest, so the six states that correspond to these values are kept in the reduced order model. After the size of the reduced order model is entered, the function returns the structured arrays `G` and `redinfo`. `G` contains the reduced order state space model and `redinfo` contains the transformation matrices  $[S_{L,BIG}]$  and  $[S_{R,BIG}]$ . The transformation matrices are used in chapter four to project control optimization matrices onto the reduced subspace.

---

<sup>2</sup> File provided by Thesis advisor Dr. Javier Kypuros. The modified command included the transformation matrices  $[S_{L,BIG}]$  and  $[S_{R,BIG}]$  in the structure array `redinfo`.

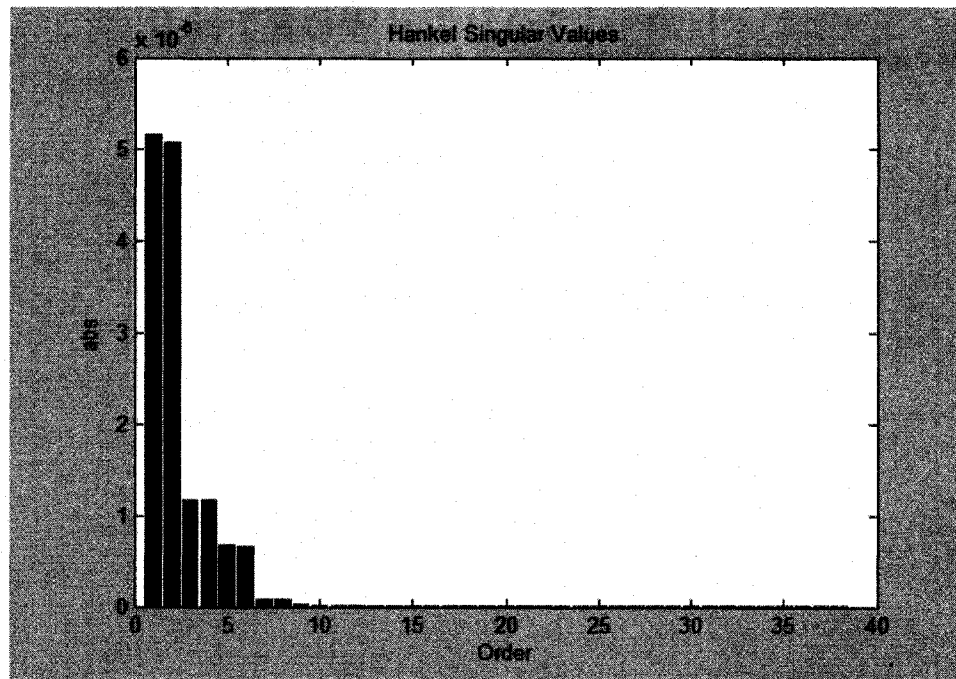


Figure 5: Hankel Singular Values for ACLD Treated Beam

## CHAPTER 4

### CONTROL METHODS

#### 4.1 Introduction to Control Methods

In this chapter a state-feedback control and a sliding mode control (SMC) are designed for the vibration attenuation of the reduced order ACLD beam model. The control gains of the state-feedback control and the sliding mode control are designed using an optimization method. To implement the optimization method in the reduced space, the optimization matrix  $[Q]$  is first designed in the full state space of the model and then projected onto the reduced space through the transformation matrix  $[S_{R,BIG}]$ . The control system block corresponding to the chapter content is highlighted below.

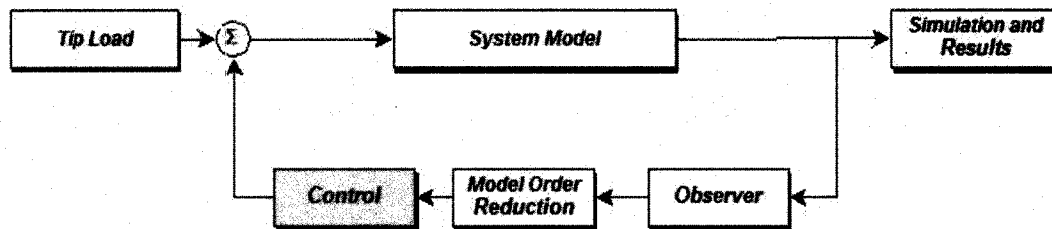


Figure 2: Control System Block Diagram

#### 4.2 LQR Control

State feedback control with control gains determined from linear quadratic regulator (LQR) theory is currently the most used control method for ACLD treated



beams [2, 13-15, 17, 27-28]. As in the design of the observer, the LQR problem is to find a gain vector  $[K]$  such that  $u = -[K] \cdot \{x(t)\}$  minimizes the performance index

$$J = \int_0^{\infty} (\{x\}^T \cdot [Q_{lqr}] \cdot \{x\} + \{u\}^T \cdot [R_{lqr}] \cdot \{u\}) dt, \quad (65)$$

where  $[Q_{lqr}]$  is a positive-definite (or positive semi-definite) Hermitian or real symmetric matrix and  $[R_{lqr}]$  is a positive-definite Hermitian or real symmetric matrix. The purpose of the performance index is to find an optimal balance between the control effort and the steady state error. The matrix  $[R_{lqr}]$  penalizes large values of the control effort  $u(t)$  and the matrix  $[Q_{lqr}]$  penalizes large values of  $\{x(t)\}$  and minimizes the steady state error [9, 12].

The optimal control gain matrix  $[K]$  was determined by first specifying  $[Q_{lqr}]$  in the full state space of the model and then projecting  $[Q_{lqr}]$  onto the reduced space by the following transformation

$$J = \int_0^{\infty} (\{x_r\}^T \cdot [S_{R,BIG}]^T \cdot [Q_{lqr}] \cdot [S_{R,BIG}] \cdot \{x_r\} + \{u\}^T \cdot [R_{lqr}] \cdot \{u\}) dt \quad (66)$$

$$[\hat{Q}_{lqr}] = [S_{R,BIG}]^T \cdot [Q_{lqr}] \cdot [S_{R,BIG}]. \quad (67)$$

The LQR control was designed in the MATLAB m-file *MixedBeamModel.m* (refer to Appendix A). The optimization matrices  $[Q_{lqr}]$  and  $[R_{lqr}]$  were chosen by testing values suggested in the literature [2, 13-15] and were then adjusted for further improvements. The values that produced the best results were  $[Q_{lqr}] = 10^{11} \cdot [I]$ , where  $[I]$  was a  $38 \times 38$  identity matrix, and  $[R_{lqr}] = 1$ . After  $[Q_{lqr}]$  was specified, the m-file *MixedBeamModel.m* was used to evaluate equation (67) and obtain the reduced optimization matrix  $[\hat{Q}_{lqr}]$ . The MATLAB command `lqr` was then used to perform the aforementioned optimization routine and determine the control gains  $[K]$ . The required

inputs were the reduced state space model  $G$  obtained in section 3.3, the reduced matrix  $[\hat{Q}_{lqr}]$ , and the optimization matrix  $[R_{lqr}]$ .

### 4.3 Sliding Mode Control

The effectiveness of sliding mode control (SMC) in ACLD applications was investigated in this research. A SMC is a robust control method that allows for both structured and unstructured uncertainties in the development of the model. Structured uncertainties arise from unknown plant parameters such as mass, stiffness, and modulus values while unstructured uncertainties involve simplified model dynamics such as neglecting structural modes in a reasonably rigid mechanical system, or estimating state values through an observer [1, 26]. Another form of unstructured uncertainties arises from disturbances that are unknown or unaccounted for. The beam model contained unstructured uncertainties due to the model simplifications made by observing (estimating) states and reducing the order of the system.

To design the SMC, a sliding surface was first designed in the reduced state space of the system. This sliding surface  $\sigma(\{x_r\})$  was the desired state trajectory of the system and contained the desired dynamics [4, 26]. The sliding surface can be nonlinear but is most commonly designed as a linear combination of the (reduced) states [4]

$$\sigma(\{x_r\}) = [S] \cdot \{x_r\}. \quad (68)$$

In this work, the resulting sliding surface was a 6-dimensional manifold in the reduced state space of system.

An optimal sliding surface matrix  $[S]$  was obtained by minimizing a performance index as with the design of the observer and LQR control [1, 4, 31]. To minimize the

performance index,  $[Q_{smc}]$  was first specified in the full state space of the model and then projected to the reduced space by the transformation matrix  $[S_{R,BIG}]$  as in equation (67). The values that produced the best results were  $[Q_{smc}] = 10^6 \cdot [I]$ , where  $[I]$  was a  $38 \times 38$  identity matrix, and  $[R_{smc}] = 1$ . The reduced  $[Q_{smc}]$  and  $[R_{smc}]$  were the values used in the MATLAB command `lqr` to obtain the optimal design matrix  $[S]$ . The required inputs were the reduced state space model “G” obtained in section 3.3, the reduced matrix  $[\hat{Q}_{lqr}]$ , and the optimization matrix  $[R_{lqr}]$  (refer to *MixedBeamModel.m* in Appendix A).

Once the sliding surface was obtained, a nonlinear switching feedback control,  $u(\{x_r\}, t)$ , was utilized to drive and maintain the system on the sliding surface. The control consisted of a linear portion,  $u_{eq}(\{x_r\})$ , and a nonlinear portion [1, 4]

$$u(\{x_r\}, t) = u_{eq}(\{x_r\}) + \eta \cdot \text{sgn}(\sigma(\{x_r\})). \quad (69)$$

The linear portion of the control drives the system state trajectory to the sliding surface and is referred to as the equivalent control force. The equivalent control force is obtained by using the fact that on the sliding mode  $\sigma(\{x\}) = \dot{\sigma}(\{x\}) = 0$  [1, 4, 6, 31]. Therefore,  $\dot{\sigma}(\{x_r\}) = [S] \cdot \{\dot{x}_r\} = 0$  and the equivalent control in equation (69) must equal

$$u_{eq}(\{x_r\}) = -([S] \cdot [B_r])^{-1} \cdot [S] \cdot [A_r] \cdot \{x_r\}. \quad (70)$$

The nonlinear portion of the control maintains the system on the sliding surface by utilizing a set of gains when the state trajectory is “above” the sliding surface and another set of gains when the state trajectory is “below” the sliding surface [4]. The parameter  $\eta$  imparts discontinuity to the control action across the sliding surface and must be chosen so that reachability and the existence of the sliding surface are guaranteed,  $\sigma(\{x_r\}) \cdot \dot{\sigma}(\{x_r\}) < 0$  [1, 4, 5, 28].  $\eta$  must therefore be

$$\eta \geq [S] \cdot [B_r]^{-1} \cdot [S] \cdot [\hat{B}]. \quad (71)$$

Before the control  $u(\{x_r\}, t)$  was implemented, modifications were necessary to the nonlinear portion of the control to eliminate the effects of chattering. Chattering is a result of imperfect control switching and is usually highly undesirable because it involves high control activity and may lead to the excitation of unmodeled structural modes [4, 26]. Chattering was eliminated by smoothing out the control discontinuity in a thin boundary layer,  $\epsilon$ , surrounding the switching surface [4, 27]. For illustrative purposes, the boundary layer is superimposed over the chattering response of a simplified 2-state model in Figure 6. By adding a boundary layer to the sliding surface  $\sigma$ , the sliding surface is effectively “thickened” and the desired state trajectory (dynamics) is reached with less control switching.

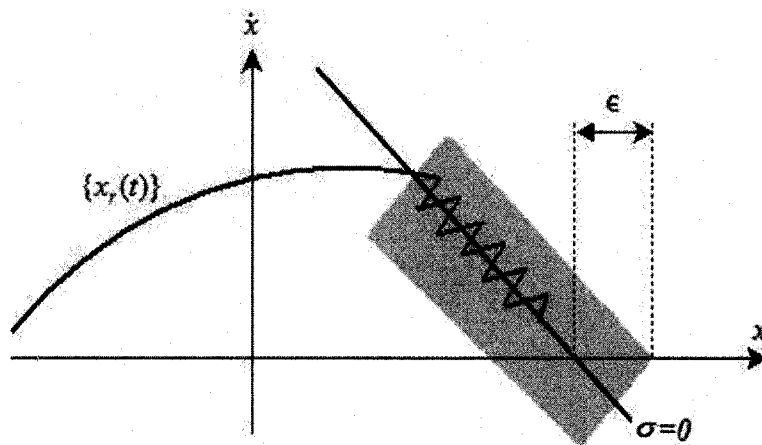


Figure 6: Chattering and Boundary Layer

To further smooth out the control discontinuity, the signum function was replaced by a saturation function with limits of  $-1$  and  $1$  as shown in Figure 7.

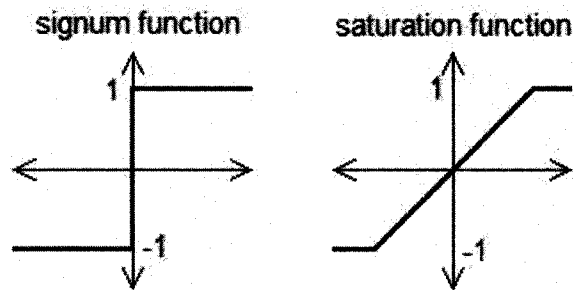


Figure 7: Signum and Saturation Functions

The final nonlinear switching control was

$$u(\{x_r\}, t) = u_{eq}(\{x_r\}) + \eta \cdot \text{sat}\left(\sigma\left(\frac{\{x_r\}}{\varepsilon}\right)\right). \quad (72)$$

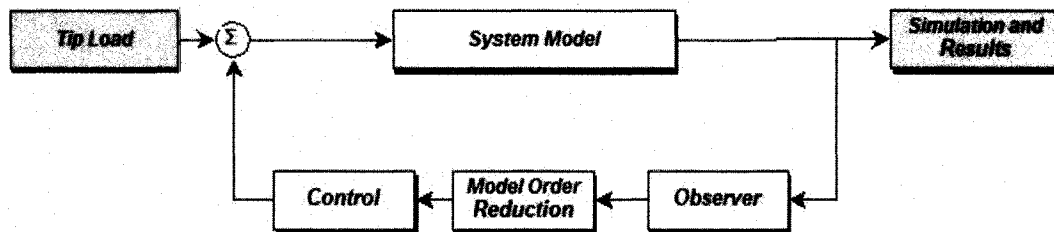
The SMC was designed in the MATLAB m-file *MixedBeamModel.m* in Appendix A. The file used the `lqr` command to determine the optimal sliding surface, evaluated the coefficient of  $\{x_r\}$  in equation (70), and calculated equation (71) to determine  $\varepsilon$ . The calculated values were input into the “SMC” block of the SIMULINK file *timoshenko\_beam\_smc* in Appendix B. The “SMC” block contained equation (72) in block diagram form and was used to implement the control for the simulation.

## CHAPTER 5

### SIMULATION AND NUMERICAL RESULTS

#### 5.1 Introduction to Simulation and Results

In this chapter the vibration damping achieved by a state-feedback linear quadratic regulator (LQR) is compared to the vibration damping achieved by a sliding mode control (SMC). Several loading conditions and saturation limits are tested to accurately compare the control methods. The blocks corresponding to the chapter content is highlighted below.



#### 5.2 Experimental Setup

An SCLD-treated cantilevered beam with dimensions of 300 mm × 15 mm × 3 mm was utilized in the comparison of the SMC and LQR control schemes. The viscoelastic layer and piezoelectric cover sheet had dimensions of 100 mm × 15 mm and were placed 30 mm from the fixed end. Several loading conditions with varying control saturation limits were simulated to compare the vibration control produced by the

controllers. Table 1 indicates the model parameters used throughout the case studies unless otherwise stated.

TABLE 1

<b>Material Properties and Other System Parameters</b>		
$E_b$	$7.0 \times 10^{10} \text{ N/m}^2$	Beam elastic modulus
$E_c$	$6.49 \times 10^{10} \text{ N/m}^2$	PZT elastic modulus
$G_b$	$2.6 \times 10^{10} \text{ N/m}^2$	Beam shear modulus
$G_c$	$2.49 \times 10^{10} \text{ N/m}^2$	PZT shear modulus
$\rho_b$	$2700 \text{ kg/m}^3$	Beam density
$\rho_c$	$7600 \text{ kg/m}^3$	PZT density
$\rho_s$	$1250 \text{ kg/m}^3$	VEM density
$t_b$	0.003 m	Beam thickness
$t_c$	0.00025 m	PZT thickness
$t_s$	0.001 m	VEM thickness
$b$	0.015 m	Beam, PZT, and VEM Thickness
$L$	0.3 m	Length of Beam
$\kappa$	$10^6 \text{ N/m}^2$	VEM equilibrium modulus
$\alpha_r$	1.0	VEM modulus parameter
$\omega_r$	1000 rad/s	VEM modulus parameter
$\zeta$	4.0	VEM modulus parameter
$k_a$	5/6	Shear correction factor
$\hat{a}$	0.64	Coefficient for structural damping
$\hat{b}$	$1.2 \times 10^{-6}$	Coefficient for structural damping
$d_{31}$	$-175 \times 10^{-12} \text{ m/V}$	PZT constant
$[Q_{smc}]$	$10^6 [I]_{38 \times 38}$	SMC optimization matrix
$[Q_{lqr}]$	$10^{11} [I]_{38 \times 38}$	LQR optimization matrix
$R$	1	SMC and LQR optimization matrix
$\varepsilon$	1	Chattering coefficient

### 5.3 Case 1: 1-N Pulse

A 1-N pulse load with a duration of 1 ms was applied to the tip of the finite element beam model at  $t = 0$  as shown in Figure 8.

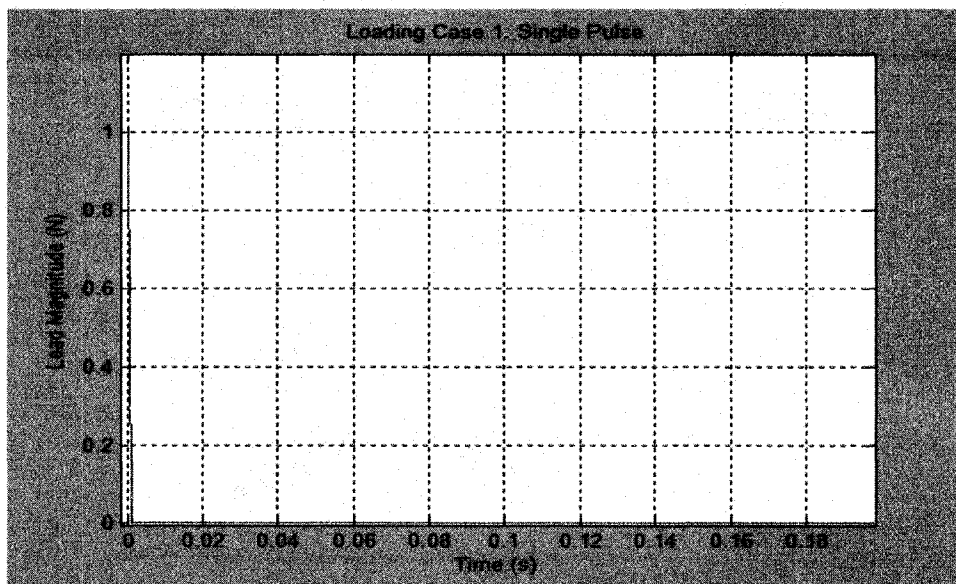


Figure 8: Single 1-N Pulse

To demonstrate the effectiveness of the ACLD treatment, the load was first applied to a beam treated with only a PCLD treatment. Implementation of the PCLD treatment resulted in a maximum displacement of 0.675 mm and a settling time well over 1 s as shown in Figure 9. It should also be noted that the passive response matched the results from [2].

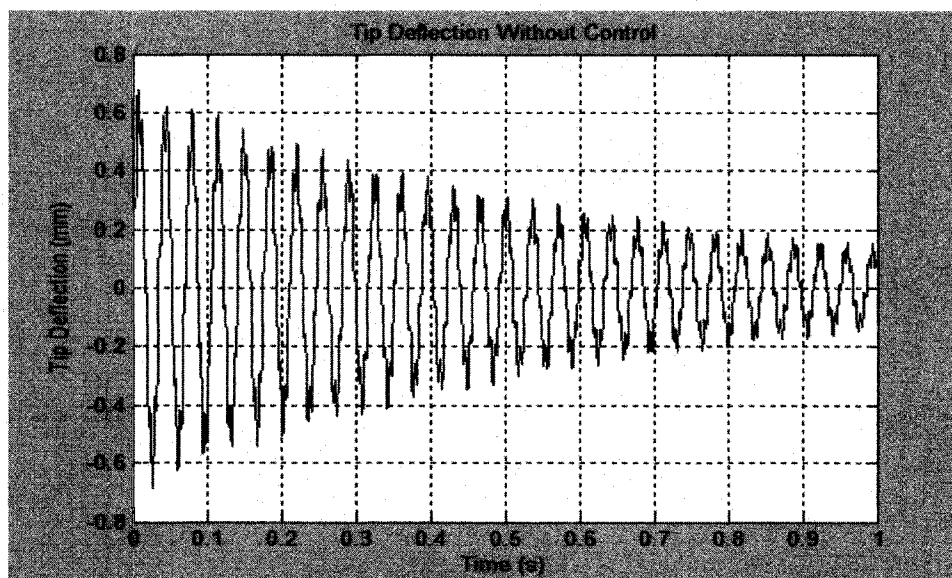


Figure 9: 1-N Pulse without Active Control



The pulse load was then applied to an ACLD treated beam model for both a LQR and a SMC control scheme. Saturation limits of 120V to -120V were enforced to replicate realistic voltage constraints.

Implementation of the ACLD treatment with a LQR control resulted in a maximum deflection of 0.666 mm and a settling time of 0.340 s (refer to Figure 10). Application of the SMC resulted in a maximum deflection of 0.672 mm and a settling time of 0.335 s (refer to Figure 11). The two control schemes produced a lower maximum deflection and a much shorter settling time when compared to the model with only a PCLD treatment, but only a marginally different vibration response, less than 1.5% difference for both maximum deflection and settling time, when compared to each other. When saturation limits are imposed, the amplitude and frequency of the beam, not the control algorithm, dictate the control activity. The LQR and SMC request the same maximum negative voltage as the beam approaches its peaks in amplitude and both control methods plateau at the maximum negative voltage until the beam begins to oscillate in the opposite direction.

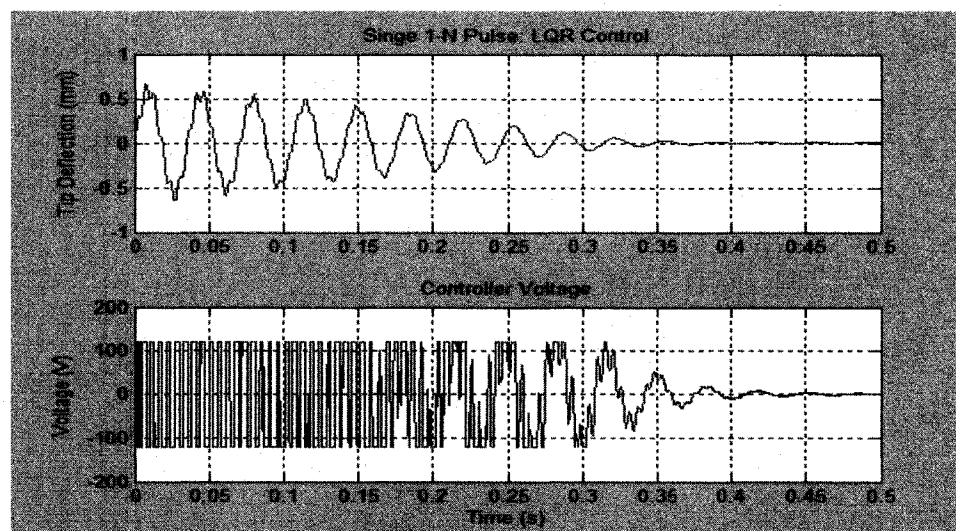


Figure 10: Single 1-N Pulse: LQR with Saturation

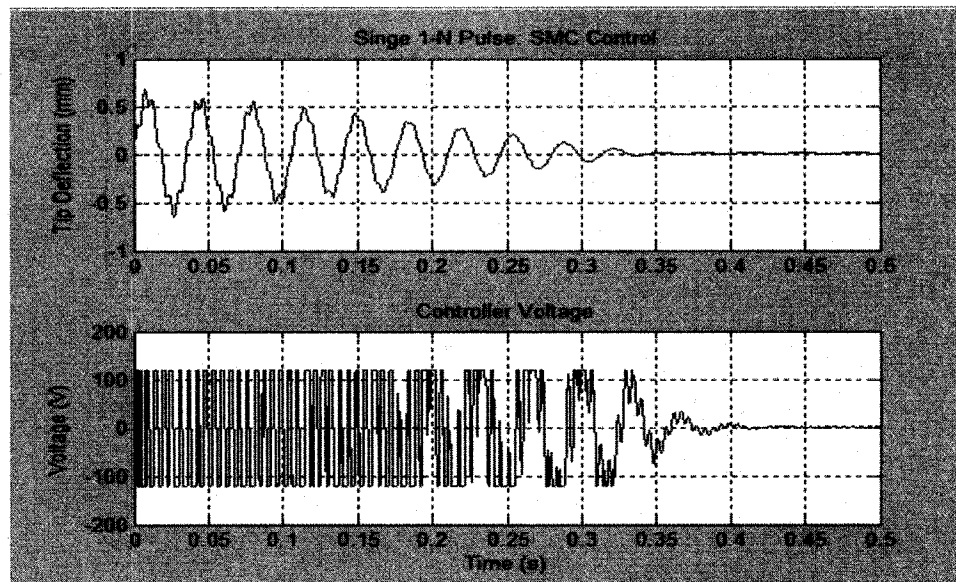


Figure 11: Single 1-N Pulse: SMC with Saturation

SMC is a high speed switching control that achieves good performance at the cost of high control activity. To further compare the effectiveness of the LQR and SMC, the 1-N pulse load was applied to the beam model for both control methods without saturation limits. Removing the saturation limits allowed the LQR and SMC to operate under the ideal condition where an infinite amount of voltage was available and a resulting force was applicable. As shown in Figure 12, Application of the LQR control without saturation limits resulted in a maximum deflection of 0.594 mm, a settling time of 0.1125 s, and required a maximum of 2380 V.

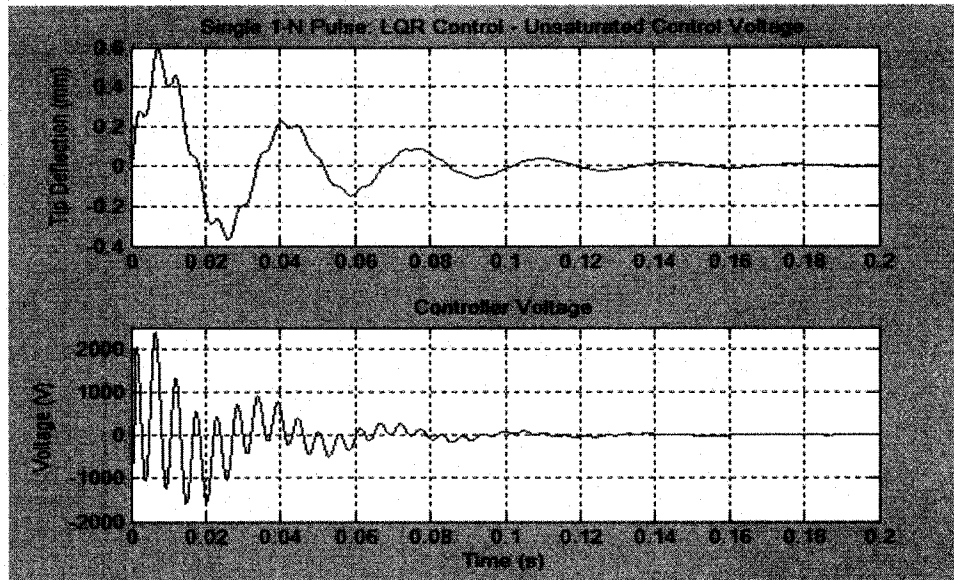


Figure 12: Single 1-N Pulse: LQR without Saturation

Application of a SMC without saturation produced a maximum deflection of 0.561 mm, a settling time of 0.0755 s, and required a maximum voltage of 2258 V (refer to Figure 13).

When compared to the LQR without saturation, the maximum deflection was decreased by 5.5%, the settling time was decreased by 32.9%, and the maximum voltage required was lowered by 122V.

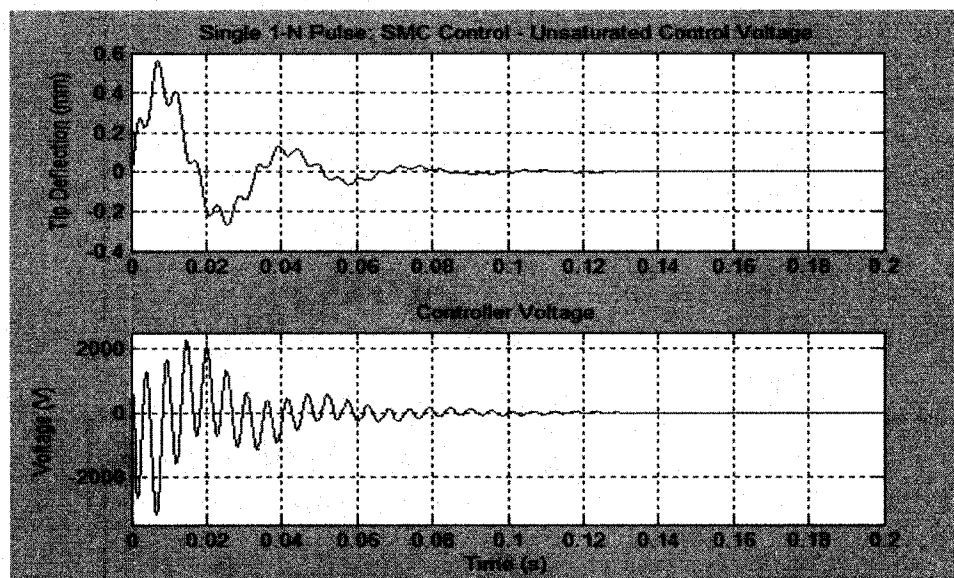


Figure 13: Single 1-N Pulse: SMC without Saturation

The improvement in performance demonstrates the effectiveness of the SMC control scheme when the saturation limits are removed and the control algorithm is the foremost factor in determining the control action.

#### 5.4 Case 2: Double 1-N Pulse

For the second case, an additional 1-N pulse load was applied to the beam 100 ms after the first pulse as shown in Figure 14. The second pulse was applied at  $t = 100$  ms so that neither control scheme had sufficient time to completely dampen the vibration from the first pulse load. This loading condition was tested to investigate the vibration control achieved for a beam that is impacted when already in motion.

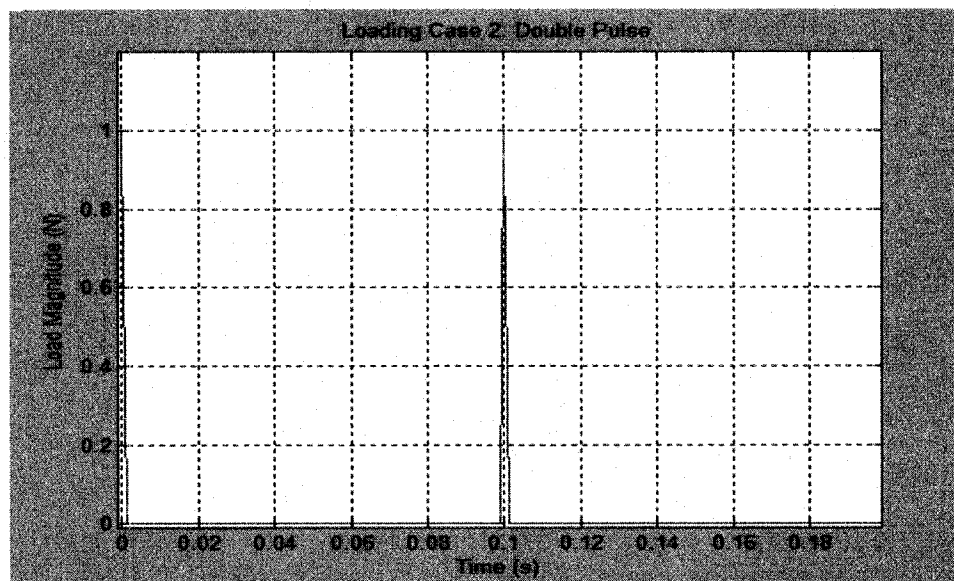


Figure 14: Double 1-N Pulse

As shown in Figure 15, the LQR control with saturation resulted in maximum amplitude of 0.955 mm and a settling time of 0.475 s. The load and saturation limits were then applied to a SMC with  $\epsilon = 0.2$  to eliminate chattering. Implementation of the SMC resulted in maximum amplitude of 0.952 mm and a settling time of 0.461 s as illustrated

in Figure 16. As in the first case, the vibration response is only marginally better for the SMC when the saturation limits are enforced.

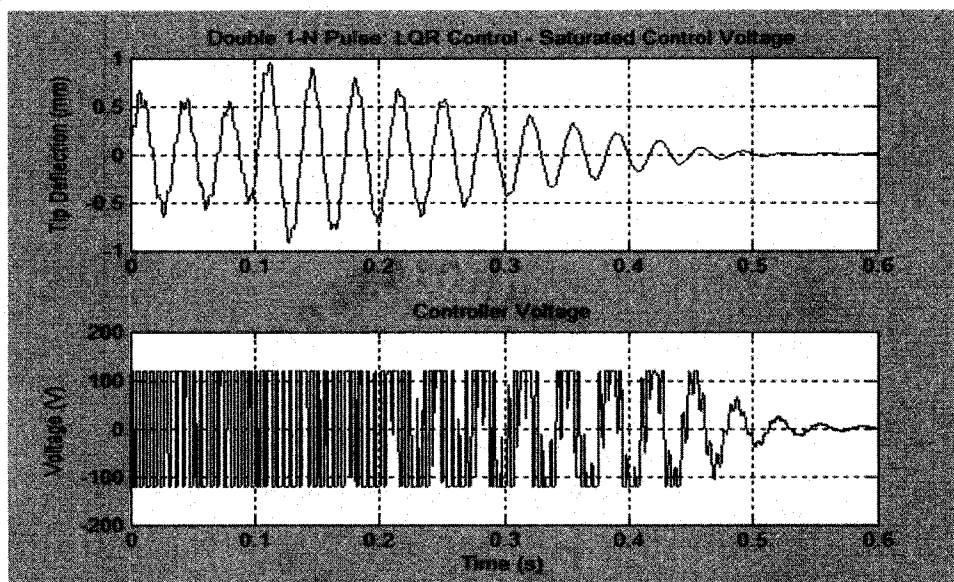


Figure 15: Double 1-N Pulse: LQR with Saturation

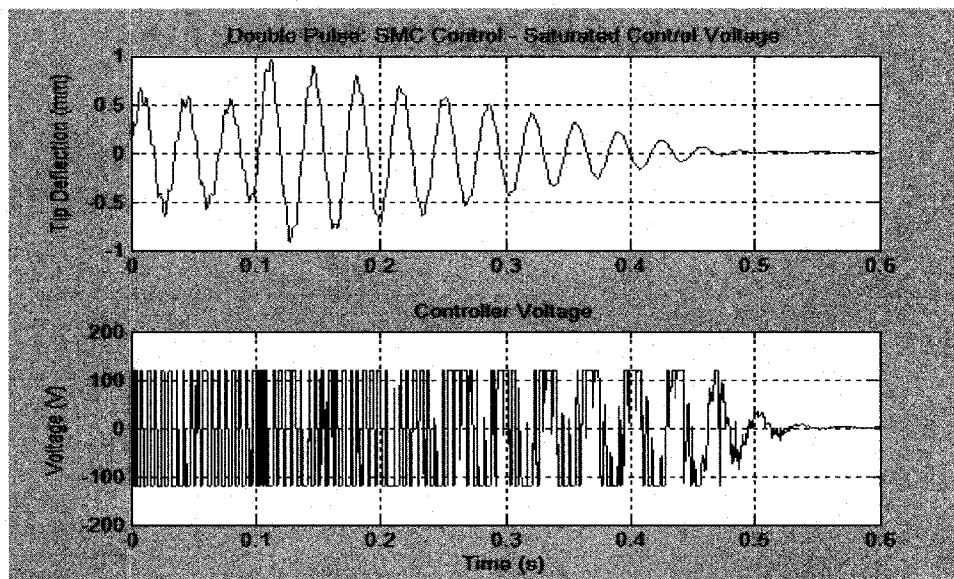


Figure 16: Double 1-N Pulse: SMC with Saturation

The saturation limits were then removed to test the controls under the ideal situation where the controllers used all of their requested voltage and the PZT was able to apply a corresponding force. Figure 17 illustrates that the LQR control without saturation

produced maximum amplitude of 0.624 mm, settling time of 0.225 s, and a maximum voltage of 2445 V.

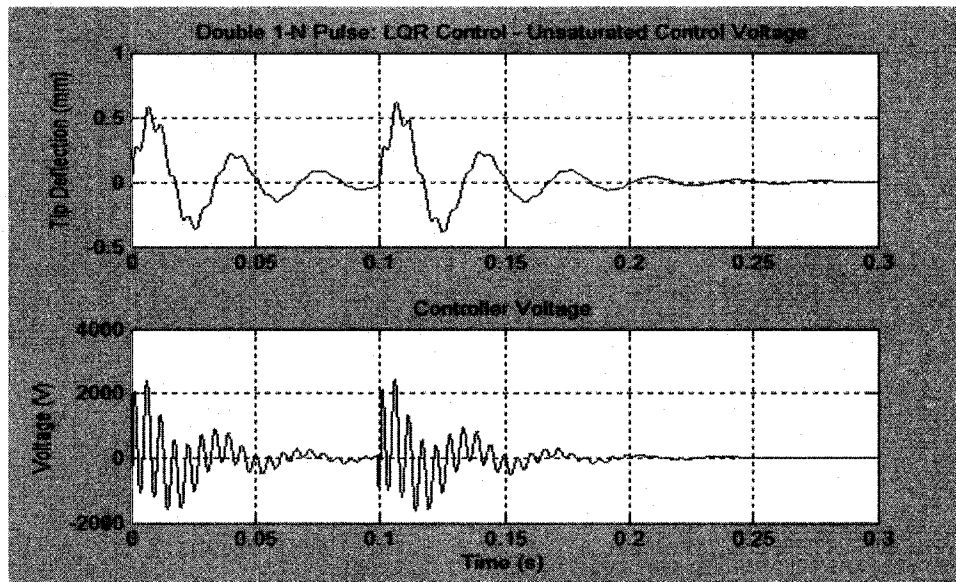


Figure 17: Double 1-N Pulse: LQR without Saturation

Application of the SMC without saturation limits resulted in a maximum deflection of 0.562 mm, settling time of 0.172 s, and a maximum voltage of 2258 V as shown in Figure 18.

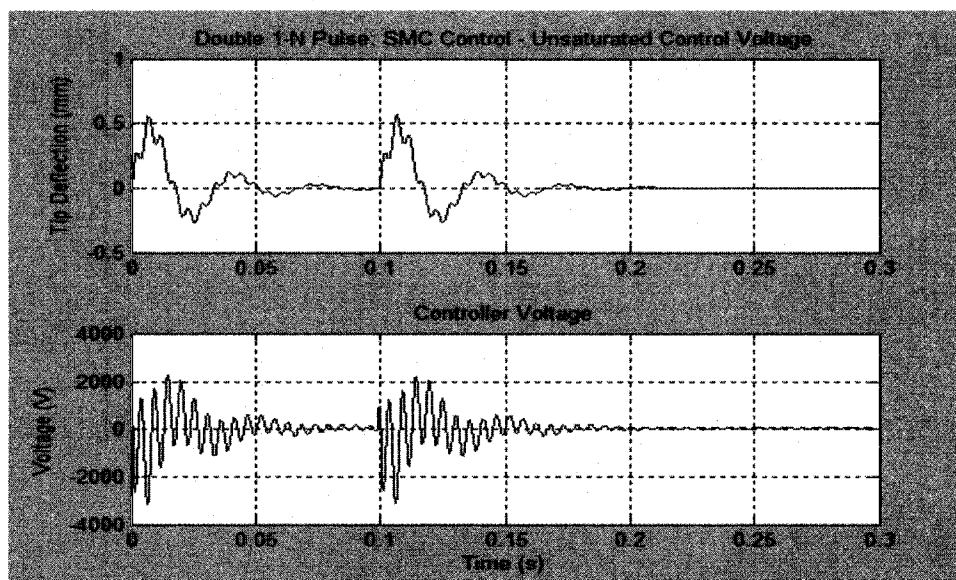


Figure 18: Double 1-N Pulse: SMC without Saturation

When compared to the LQR without saturation limits, the maximum amplitude was decreased by 9.93% and the settling time was decreased by 23.56%. As in the first case, the SMC control scheme greatly out performed the LQR control scheme when saturation limits were removed.

### 5.5 Case 3: Harmonic Load

For the third case, a harmonic load of  $f(t) = 0.1 \cdot \sin(179 \cdot t)$  N was applied to the tip of the beam for 0.193 s as shown in Figure 19. The frequency of the harmonic load corresponded to a natural frequency of the full order model and was chosen to investigate the effectiveness of the control schemes during resonance as illustrated in Figure 20. The duration of the load was chosen so that the sinusoid completed 5.5 cycles and terminated at  $f(t) = 0$ . The amplitude of the control force was designed so that the maximum deflection of the beam model would not exceed 5mm during the application of the load.

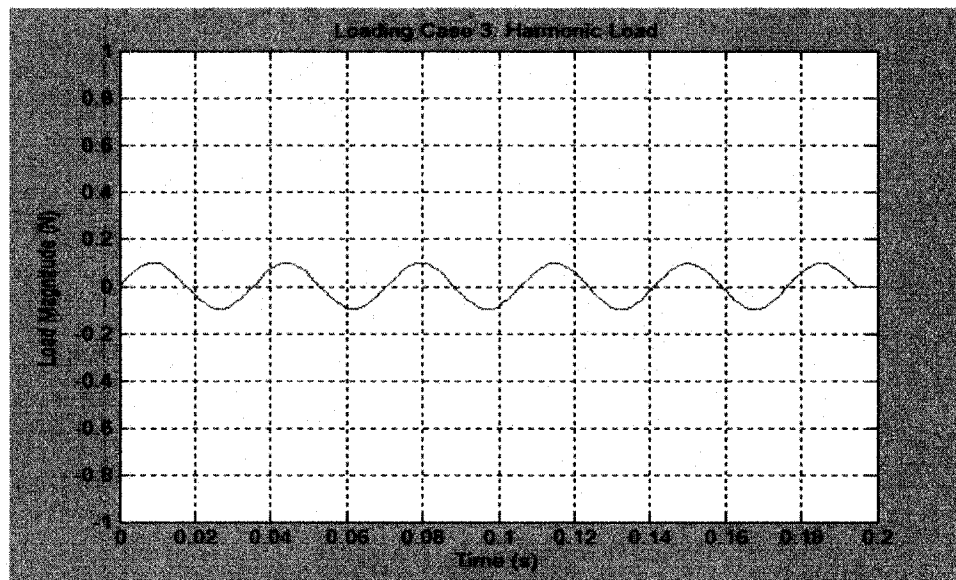


Figure 19: Harmonic Load

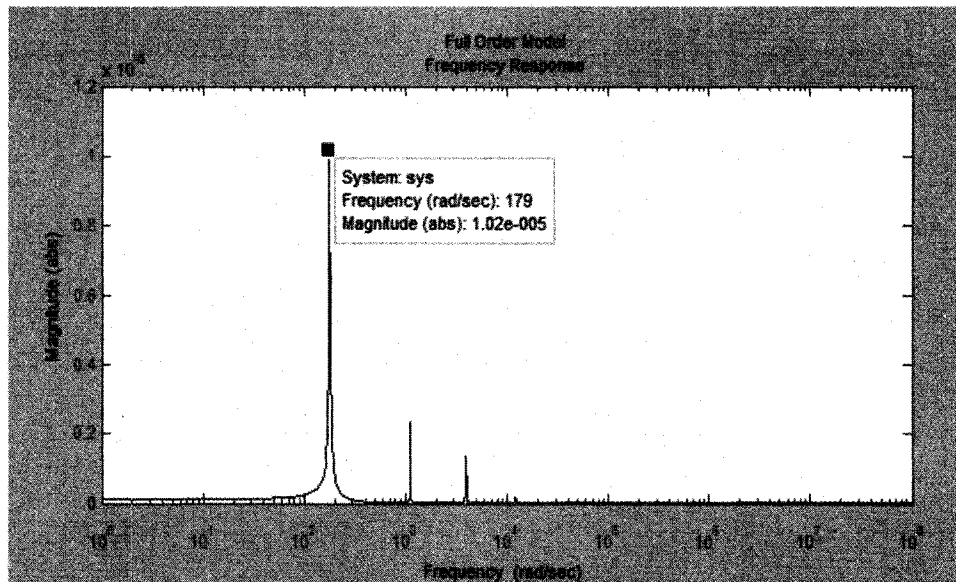


Figure 20: Frequency Response of Full Order Model

As with the first two cases, the vibration response was compared for a LQR and SMC with saturation and then a LQR and SMC without saturation. Implementation of the LQR with saturation limits produced a maximum deflection of 4.480 mm and a settling time of 0.95 s as shown in Figure 21.

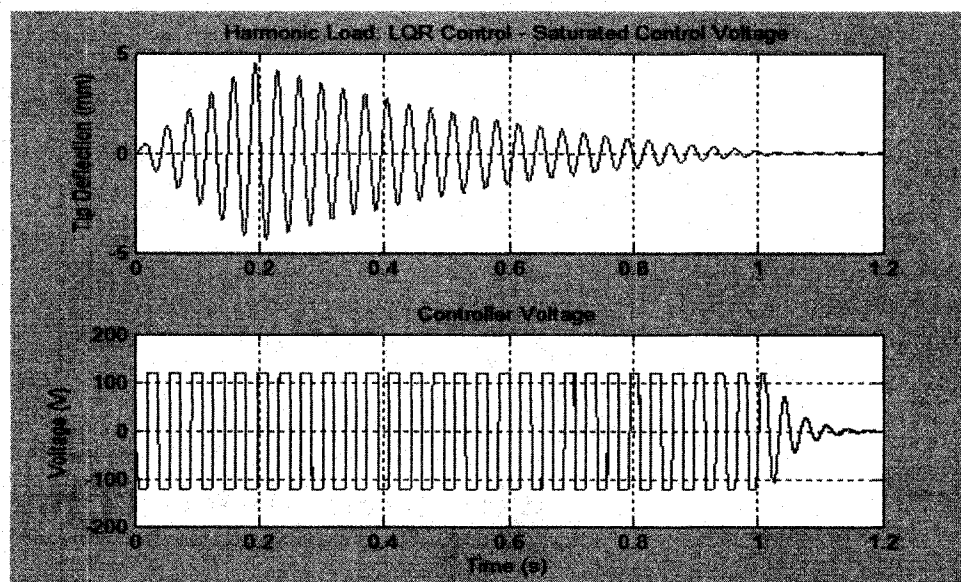


Figure 21: Harmonic Load: LQR with Saturation



The SMC with saturation limits resulted in a maximum deflection of 4.475 mm a settling time of 0.93 s as shown in Figure 22.

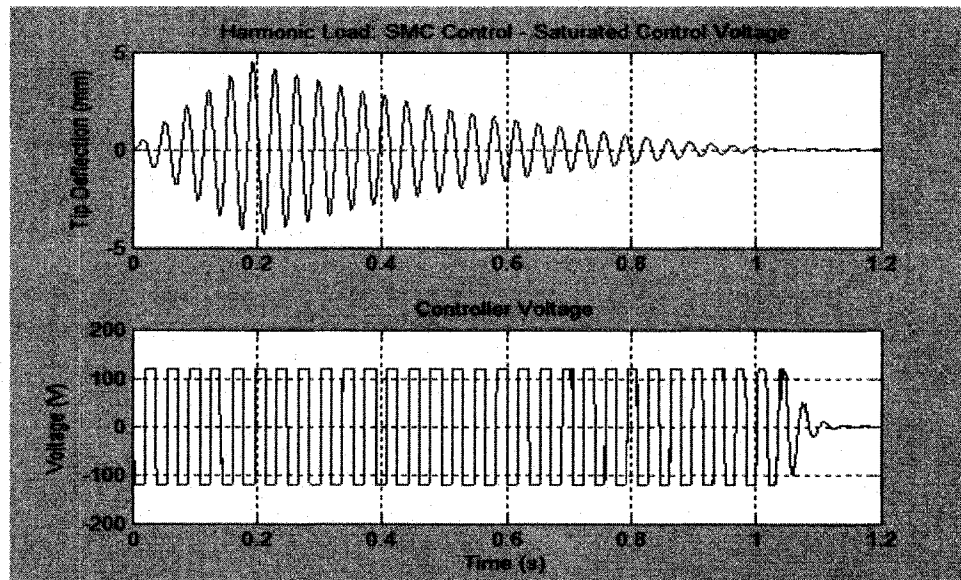


Figure 22: Harmonic Load: SMC with Saturation

As in the first two cases, the vibration response of the beam when the SMC was implemented was only slightly improved when saturation limits were imposed. In this case the dependence of the control activity on the frequency and amplitude of the vibration response is more evident. As the beam deflection increases the voltage requested by the controller decreases and then plateaus until the beam begins to oscillate in the opposite direction.

The load was then applied to a LQR and SMC without saturation limits to investigate the ideal response of the controllers. The LQR without saturation limits produced a vibration response with maximum amplitude of 1.1445 mm, settling time of 0.2980 s, and a maximum voltage of 1622 V as shown in Figure 23. Implementation of a SMC without saturation limits resulted in a maximum amplitude of 0.716 mm, settling time of 0.2485 s, and a maximum voltage of 1675 V as shown in Figure 24. When

compared to the LQR without saturation limits, the amplitude of the vibration was decreased by 37.44% and the settling time was decreased by 16.61%.

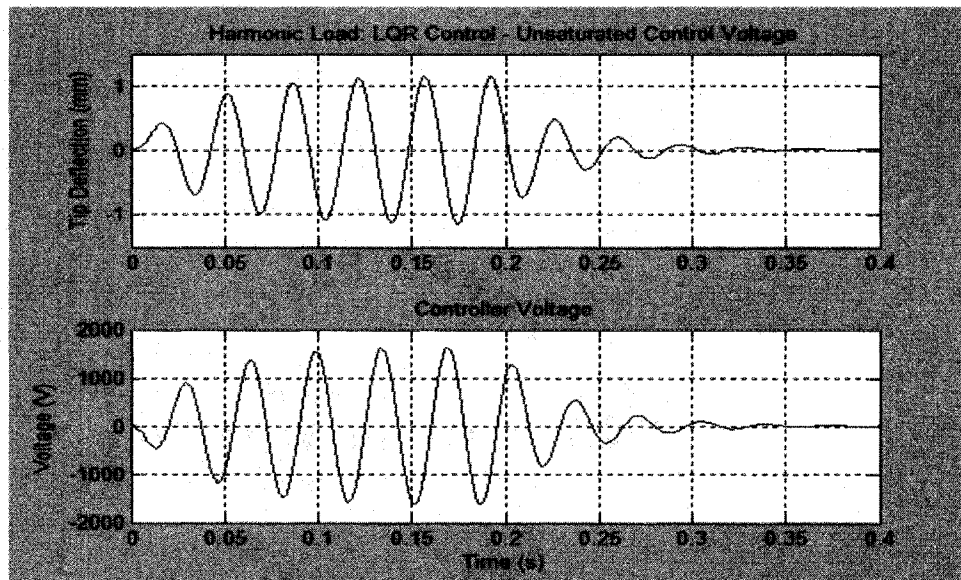


Figure 23: Harmonic Load: LQR without Saturation

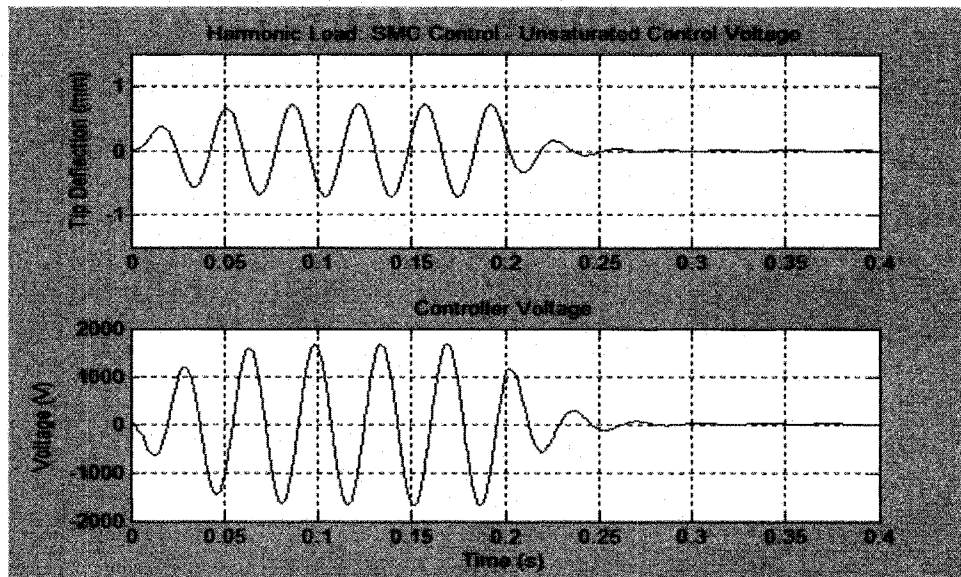


Figure 24: Harmonic Load: SMC without Saturation

The implementation of the SMC produced a smaller maximum deflection and a shorter settling time when saturation limits were removed for all three loading cases. The improvement in vibration control when compared to the LQR was a result of the increase

in the allowed control activity and the utilization of a robust control scheme in the presence of unstructured modeling errors. The results are summarized in Table 2.

TABLE 2

<b>Summary of Results</b>						
<b>Loading Case</b>	<b>Control Method</b>	<b>Saturation Limit</b>	<b>Max Deflection (mm)</b>	<b>Settling Time (s)</b>	<b>% Decrease of md Relative to SMC</b>	<b>% Decrease of <math>t_s</math> Relative to SMC</b>
1	LQR	On	0.666	0.3400	-0.90	1.47
	SMC	On	0.672	0.3350		
	LQR	Off	0.594	0.1125	5.56	32.89
	SMC	Off	0.561	0.0755		
2	LQR	On	0.955	0.4750	0.31	2.95
	SMC	On	0.952	0.4610		
	LQR	Off	0.624	0.2250	9.94	23.56
	SMC	Off	0.562	0.1720		
3	LQR	On	4.480	0.9500	0.11	2.11
	SMC	On	4.475	0.9300		
	LQR	Off	1.145	0.2980	37.44	16.61
	SMC	Off	0.716	0.2485		

## CHAPTER 6

### CONCLUSIONS AND FUTURE WORK

In this work an actively constrained layer damping (ACLD) treatment was applied to a beam for vibration suppression. The beam was modeled using Timoshenko beam theory and the viscoelastic material in the ACLD treatment was modeled using the GHM method. Energy methods were utilized to obtain the mass, damping, and stiffness matrices for the discrete elements of the beam and a MATLAB algorithm was created to assemble these segments into a finite element model for the entire beam. The finite element model was neither completely state controllable nor observable. An observer was designed and model order reduction (MOR) was implemented to obtain a simplified, efficient, and more controllable model. The process of observation and MOR introduced modeling errors into the system in the form of unstructured uncertainties.

The most common control method utilized in the control of ACLD treatments is state feedback control with control gains determined using linear quadratic (LQR) theory. Sliding mode control (SMC) is a robust control method and was utilized in this research to determine if a robust control method produced better vibration attenuation in ACLD treated beam models containing unstructured uncertainties.

The vibration control for a SMC and LQR were compared for several loading conditions. SMC produced only a marginally better vibration response for all loading conditions when saturation limits were imposed. The similarity in vibration control was a

direct result of the limitations on control activity placed on the SMC. Saturation limits were removed to test effectiveness of the control schemes for an ideal scenario where the controllers were allowed an infinite amount of voltage and a corresponding force was applicable. When compared to the LQR without saturation limits, the vibration control for a SMC was greatly improved. This improvement in response was due to the increase in control activity and the utilization of a robust control scheme in the presence of unstructured uncertainties.

Further studies can be performed to include modeling errors due to structured uncertainties, random forces, and combinations of both to achieve increasingly realistic models. In this work, it was assumed the material properties of the beam, VEM, and PZT were known exactly. This is rarely the case especially with viscoelastic materials where the material properties are estimated experimentally, highly dependent on the frequency of the vibration, and extremely sensitive to temperature. The tip loads were also assumed exactly known before the simulation. A more realistic model would employ a bounded random tip load to model the uncertainty in loads. The forces can also be applied to varying sections of the beam during the simulation to represent uncertainty in the application of the load. Additional studies can also be performed to include tracking applications and other applications that require much less control effort. Applications that require less control effort will not cause the SMC to reach saturation limits as often and will therefore increase the control effectiveness.

This research has contributed several advancements in the area of ACLD research. First, ACLD model modifications were introduced through the implementation of an observer and model order reduction. These modifications produced a simplified,

efficient, and more controllable state space model. Secondly, a sliding mode control (SMC) was implemented and compared to the conventional state-feedback control with gains determined using linear quadratic regulator (LQR) theory. SMC was shown to be effective and produced greater vibration control in the absence of saturation limits.

## REFERENCES

1. Adhikari, R., Yamaguchi, H., "Sliding Mode Control of Buildings with ATMD," *Earthquake Engineering and Structural Dynamics*, Vol. 26, pp. 409-422, 1997.
2. Balamurugan, V., Narayanan, S., "Finite Element Formulation and Active Vibration Control Study on Beams Using Smart Constrained Layer Damping (SCLD) Treatment," *Journal of Sound and Vibration*, Vol. 249(2), pp. 227-250, 2002.
3. Chiang R., Safonov M "A Schur Method for Balanced Truncation Model Reduction," *IEEE Transactions on Automatic Control*, Vol. 34(7), July 1989.
4. DeCarlo, R., Matthews, G., Zak, S., "Variable Structure of Nonlinear Multivariable Systems: A Tutorial," *Proceedings of the IEEE*, Vol.76, No.3 1998.
5. Felippa, C., Nonlinear Finite Element Method, 2006, Colorado University, 18 Mar. 2006, <<http://www.colorado.edu/engineering/CAS/courses.d/MFEM.d>>
6. Fernandez, B., "Generalized Sliding Mode Control for MIMO Nonlinear Systems," Massachusetts Institute of Technology, 1988.
7. Golla, D., Hughes, P., "Dynamics of Viscoelastic Structures-A Time-Domain, Finite Element Formulation", *Journal of Applied Mechanics*, Vol.52, Dec. 1985.
8. Han, J., Rew, K., Lee, I., 'An Experimental Study of Active Vibration Control of Composite Structures with a Piezo-Ceramic Actuator and a Piezo-Film Sensor", *Smart Materials and Structures*, Vol.6, pp. 549-558, 1997.

9. Healy, Observer Design. Naval Post Graduate School.  
<[web.nps.navy.mil/~me/healey/ME4811/notes3.pdf](http://web.nps.navy.mil/~me/healey/ME4811/notes3.pdf)>
10. Hirschorn, R., Lewis, A., "Geometric Sliding Mode Control: The Linear and Linearised Theory", <[http://www.mast.queensu.ca/~ron/psfiles/hirschorn\\_lewis.pdf](http://www.mast.queensu.ca/~ron/psfiles/hirschorn_lewis.pdf)>
11. Kwon, Y., The Finite Element Method Using MATLAB, *CRC Press*, 1997
12. Lewis, F., Symons, V., Optimal Control, *John Wiley and Sons*, 1995.
13. Liao, W., Wang, K., "Characteristics of Enhanced Active Constrained Layer Damping Treatments with Edge Elements, Part 1: Finite Element Model Development and Validation", *Transactions of the ASME*, Vol. 120, Oct. 1998.
14. Liao, W., Wang, K., "On the Analysis of Viscoelastic Materials for Active Constrained Layer Damping Treatments," *Journal of Sound and Vibration*, Vol. 207(3), pp. 319-334, 1997.
15. Lim, Y., Varadan, Va., Varadan, Vi., "Closed Loop Finite-Element Modeling of Active Constrained Layer Damping In the Time Domain Analysis," *Smart Materials and Structures*, Vol. 11, pp. 89-97, 2002.
16. Mattman, Shear Locking, Eng-Tips Forums. <<http://www.eng-tips.com/viewthread.cfm?qid=30464>>.
17. Margaretha, J. "Hybrid Active/Passive Damping Models with Frequency Dependent Damping," Oct. 1997, Dissertation, Virginia Polytechnic Institute and State University.
18. Ogata, K., Modern Control Engineering, fourth edition, *Prentice Hall*, NJ, 2002.
19. Park, S., Modeling of Viscoelastic Energy Dissipation Units For Seismic and Wind Mitigation. Georgia Institute of Technology. Nov 1999.



20. Prathap, G., Finite Element as Analysis, CSIR Centre for Mathematical Modeling and Computer Simulation. <<http://www.cmmacs.ernet.in/cmmacs/pdf/Ch06>>.
21. Rao, M., "Recent Applications of Viscoelastic Damping for Noise Control in Automobiles and Commercial Airplanes," May 2001. India-USA Symposium on Emerging Trends in Vibration and Noise Engineering.
22. Rao, S., Mechanical Vibrations, Fourth edition. *Pearson Prentice Hall*, 2004.
23. Rao, S., Sunar, M., "Piezoelectricity and its Use in Disturbance Sensing and Control of Flexible Structures: A Survey," *ASME Journal of Applied Mechanics Review*, 47(4), pp 113-121.
24. Rao, S., Veley, D., "A Comparison of Active, Passive and Hybrid Damping in Structural Design," *Smart Material. Structures*, Vol. 5, pp.660-671, 1996.
25. Reddy J., An Introduction to the Finite Element Method, second edition, *McGraw-Hill*, 1993.
26. Slotine J., Li W., Applied Nonlinear Control, *Prentice Hall*, NJ, 1991.
27. Trinade, M., Benjeddou A., Onayon R., "Finite Element Analysis of Frequency- and Temperature-Dependent Hybrid Active-Passive Vibration Damping,"
28. Trinade, M., Benjeddou A., Onayon R., "Finite Element Modeling of Hybrid Active-Passive Damping of Multilayer Piezoelectric Sandwich Beams. Part 1: Formulation,"
29. White J., Model Order Reduction Site at MIT. Nov. 2005. Massachusetts Institute of Technology. <[web.mit.edu/mor](http://web.mit.edu/mor)>.
30. Wilcox K., "An Introduction to Model Reduction for Large Scale Applications". Massachusetts Institute of Technology.  
<[http://web.mit.edu/mor/papers/ADCL\\_Sept05.pdf](http://web.mit.edu/mor/papers/ADCL_Sept05.pdf)>

31. Young K., Özgüner Ü., "Sliding-Mode for Robust Linear Optimal Control",  
*Automatica*, Vol. 33(7), pp. 1313-1323, 1997.

**APPENDIX A**  
**MATLAB FILES**

```

% Nodes and Degrees of Freedom -----

nelL = 1;           % number of elements left of ACLD
nelC = 1;           % number of elements for ACLD
nelR = 3;           % number of elements right of ACLD
nel = nelL+nelC+nelR; % number of elements
mnel = 2;           % number of nodes per element
ndof1 = 3;          % number of dofs per node for beam element
ndof2 = 4;          % number of dofs per node for SCLD elements
nnode = (mnel-1)*nel+1; % total number of nodes

PLL = 0.1;          % percent length of beam segment left of ACLD
PLC = 0.3;          % percent length of ACLD beam segment
PLR = 0.6;          % percent length of beam segment right of ACLD

% total system dofs

sdof = nelL*ndof1 + ((nelC+1)*ndof2+nelC*2) + nelR*ndof1;

% System matrices -----

M = zeros(sdof,sdof); % initialization of system mass matrix
C = zeros(sdof,sdof); % initialization of system damping matrix
K = zeros(sdof,sdof); % initialization of system stiffness matrix
Fd = zeros(sdof,1);   % initialization of Fd vector
index = zeros(1,6);   % initialization of index vector
Fd(sdof-1) = 1.0;     % tip load magnitude
source = 1;           % 1 for impulse, 2 for periodic, and 3 for step
Pc = zeros(sdof,1);   % initialization of control vector

% Compute system mass, damping, and stiffness matrices
[M,C,K,Pc] = SystemMatricies03(M,C,K,Pc,nelL,nelC,nelR,nel,PLL,PLC,✓
PLR);

% State Space Representation -----
% Apply constraints
% The following lines remove rows and columns corresponding to
% displacements specified by boundary conditions.

```

```

M = M(4:sdof,4:sdof);
K = K(4:sdof,4:sdof);
C = C(4:sdof,4:sdof);
Fd = Fd(4:sdof,1);
Pc = Pc(4:sdof,1);

AA = [zeros(sdof-3) eye(sdof-3); -inv(M)*K -inv(M)*C];
BB = [zeros(sdof-3,1); inv(M)*Pc];
CC = zeros(1,(sdof-3)*2);
CC(18) = 1;
DD = 0;

sys = ss(AA,BB,CC,DD);

% Observability and Controllability of FOM -----

C1 = [BB AA*BB AA^2*BB AA^3*BB AA^4*BB AA^5*BB AA^6*BB...
      AA^7*BB AA^8*BB AA^9*BB AA^10*BB AA^11*BB AA^12*BB...
      AA^13*BB AA^14*BB AA^15*BB AA^16*BB AA^17*BB AA^18*BB...
      AA^19*BB AA^20*BB AA^21*BB AA^22*BB AA^23*BB AA^24*BB...
      AA^25*BB AA^26*BB AA^27*BB AA^28*BB AA^29*BB AA^30*BB...
      AA^31*BB AA^32*BB AA^33*BB AA^34*BB AA^35*BB AA^36*BB...
      AA^37*BB];

rank(C1);

O1 = [CC; CC*AA; CC*AA^2; CC*AA^3; CC*AA^4; CC*AA^5;...
      CC*AA^6; CC*AA^7; CC*AA^8; CC*AA^9; CC*AA^10; CC*AA^11;...
      CC*AA^12; CC*AA^13; CC*AA^14; CC*AA^15; CC*AA^16;...
      CC*AA^17; CC*AA^18; CC*AA^19; CC*AA^20; CC*AA^21;...
      CC*AA^22; CC*AA^23; CC*AA^24; CC*AA^25; CC*AA^26;...
      CC*AA^27; CC*AA^28; CC*AA^29; CC*AA^30; CC*AA^31;...
      CC*AA^32; CC*AA^33; CC*AA^34; CC*AA^35; CC*AA^36;...
      CC*AA^37];

rank(O1);

% Observer Design -----

Qob = 1*eye(size(AA));

[P,Eig,L] = care(AA',CC',Qob);           % Solves Algebraic Ricatti Equation

L = L';                                   % Observer gain matrix

```

```

Eig;                                % Must Have Negative Real Parts To Be Stable

% Model Order Reduction -----
[G, redinfo] = schurmrJAK(sys,6);

SLBIG = redinfo.SLBIG;                % Transformation Matrix
SRBIG = redinfo.SRBIG;

T = SLBIG';

% Check for Controllability and Observability in ROM -----

CM = [G.b G.a*G.b G.a^2*G.b G.a^3*G.b G.a^4*G.b G.a^5*G.b];

rank(CM);                            % Rank of Controlability Matrix

OM = [G.c; G.c*G.a; G.c*G.a^2; G.c*G.a^3; G.c*G.a^4; G.c*G.a^5];

rank(OM);                            % Rank of Observability Matrix

% LQR Control Design -----

[rM,cM] = size(M);

BBhat = [zeros(rM,1); inv(M)*Fd];    % Vector Nedded for Tip Load

Qct = 1e11*eye(size(AA));           % Q specified in the full state space

Qct = SRBIG'*Qct*SRBIG;             % Transfrmion of Q to Control Reduced States

R = 1;

sysrom = ss(G.a,G.b,G.c,G.d);

[Kc,P1,E1] = lqr(G.a,G.b,Qct,R);     % Control Gain based on LQR design

%Kc = [0 0 0 0 0 0];                % No control

```

```

% Sliding Mode Control Design -----

Qs = 1*10^6*eye(size(AA));          % Q specified in the full state space
Qs = SRBIG'*Qs*SRBIG;              % Transformation of Q to Control Reduced States
Rs = 1;

[S,P2,E2] = lqr(G,Qs,Rs);          % S = Optimal sliding surface
Eta = 1.5*abs(inv(S*G.b)*(S*T*BBhat));
KUeq = -inv(S*G.b)*S*G.a;          % Equivalent Control Force
e = 1;                             % Boundary Layer Thickness of Sliding Surface

```

```
function [M,C,K,Pc] = SystemMatrices03(M,C,K,Pc,nelL,nelC,nelR,nel,PLL, PLC,PLR)
```

```
%-----  
% SystemMatrices03.m (created by jak 032305; updated by jak 032305)  
% This file calculates system matrices (M, C, K, and P) for the ACLD  
% problem. This accounts for a partially ACLD treated beam.  
%  
%-----
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Parameters %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%  
% Variabe subscripts: b = beam, c = PZT, s = VEM
```

```
% Elastic Modulus (N/m^2)  
Eb = 7.0e10;  
Ec = 6.49e10;  
Es = 0*14e6;
```

```
% Shear Modulus  
Gb = 2.6e10;  
%Gb = 3.8*10^6;  
Gc = 2.496e10;  
Gs = 0*3.8*10^6;
```

```
% Width of Beam, PZT, and VEM (m)  
b = .015;
```

```
% Thickness (m)  
tb = .003;  
ts = .00025;  
tc = .001;
```

```
% Cross sectional Area (m^2)  
Ab = tb*b;  
Ac = ts*b;  
As = tc*b;
```

```
% Moment of Inertia (m^4)  
Ib = b*tb^3/12;  
Ic = b*ts^3/12;  
Is = b*tc^3/12;
```

```
% Density (kg/m^3)  
rhob = 2700;  
rhoc = 7600;
```



```

rhos = 1250;

% shear correction factor
k_alpha = 5/6;

% VEM Layer Parameters
kappa = 1*10^6; % Equilibrium Value of Modulus (N/m^2)
alpha = 1.0;
omega = 1000;
chi = 4.0;

% Total Length of Beam (m)
L = .3;

% Coefficients for Structural/Rayleigh Damping
a_hat = .64;
b_hat = 1.2e-6;
InternalDamping = 1;

% Piezoelectric constant (m/V)
d31 = -175e-12;

BLP = [Eb Gb rhob tb a_hat b_hat]; % Beam Layer Parameters
VLP = [Es rhos ts alpha omega chi kappa]; %Viscoelastic Layer Parameters
CLP = [Ec Gc rhoc tc d31]; % Constraining Layer Parameters

Le1 = PLL*L/nelL; % element 1 length
Le2 = PLC*L/nelC; % element 2 length
Le3 = PLR*L/nelR; % element 3 length

[Me1,Ce1,Ke1,Pce1] = TimoshenkoFEM2 (BLP,k_alpha,b,Le1,InternalDamping);
[Me2,Ce2,Ke2,Pce2] = BalamuruganNarayananFEM2 (BLP,k_alpha,VLP,CLP,b,Le2,InternalDamping);
[Me3,Ce3,Ke3,Pce3] = TimoshenkoFEM2 (BLP,k_alpha,b,Le3,InternalDamping);

for iel = 1:nel
    if (iel <= nelL)
        % Timoshenko beam elements
        Me = Me1; Ce = Ce1; Ke = Ke1; Pce = Pce1;
    end
end

```

```

    if (iel == 1)
        index = [1:6];
    else
        index = [index(4:6) index(4:6)+3];
    end
elseif (iel > nell) & (iel <= nell+nelC)
    % ACLD elements
    Me = Me2; Ce = Ce2; Ke = Ke2; Pce = Pce2;
    if (iel == nell+1)
        index = [index(4:6) index(4:6)+3 index(4:6)+6 index(6)+7];
    else
        index = [index(5:8) index(5:10)+6];
    end
else
    % Timoshenko beam elements
    Me = Me3; Ce = Ce3; Ke = Ke3; Pce = Pce3;
    if (iel == nell+nelC+1)
        index = [index(5:7) index(5:7)+6];
    else
        index = [index(4:6) index(4:6)+3];
    end
end
M = FEASMBL(M,Me,index);           % assembles system mass matrix
C = FEASMBL(C,Ce,index);         % assembles system damping matrix
K = FEASMBL(K,Ke,index);         % assembles system stiffness matrix
Pc = FEASMBL2(Pc,Pce,index);     % assembles system control vector
end

```

```

function [Me,Ce,Ke,Pe] = TimoshenkoFEM2(BLP,k_alpha,b,Le,
InternalDamping)
% -----
% TimoshenkoFEM2.m (created by jak 021704; updated by jak 021904)
% This is the same as the Balamurugan & Narayanan FEM without the
damping
% and constraining layers. This has be corrected from TimoshenkoFEM
where
% there was an error with Mwb.
% qe = [u1 w1 theta1 u2 w2 theta2]'
% -----

% Beam layer parameters
Eb = BLP(1);
Gb = BLP(2);
rhob = BLP(3);
tb = BLP(4);
a_hat = BLP(5);
b_hat = BLP(6);

Ab = b*tb;
Ib = b*tb^3/12;

% beam stiffness bending term
Kwbb = Eb*Ib/Le*[...
    zeros(2,6);...
    0    0    1    0    0    -1;...
    zeros(2,6);...
    0    0    -1    0    0    1];

% beam stiffness transverse shear term
% Must use a quadrature rule to correct the shear term and prevent
% shear locking. This matrix matches that in Kwon & Bang, THE FINITE
% ELEMENT METHOD, p. 244, Eq. 8.2.11.
Kwbs = Gb*Ab*k_alpha/4/Le*[...
    0    0    0    0    0    0;...
    0    4    2*Le    0    -4    2*Le;...
    0    2*Le    Le^2    0    -2*Le    Le^2;...
    0    0    0    0    0    0;...
    0    -4    -2*Le    0    4    -2*Le;...
    0    2*Le    Le^2    0    -2*Le    Le^2];

% beam stiffness extension term
Kub = Eb*Ab/Le*[...
    1    0    0    -1    0    0;...
    zeros(2,6);...
    -1    0    0    1    0    0;...

```

```

        zeros(2,6)];
% beam tranverse mass term
Mwb = rhob*Le/6*[ ...
    0      0      0      0      0      0;...
    0     2*Ab    0      0     Ab    0;...
    0      0     2*Ib    0      0    Ib;...
    0      0      0      0      0    0;...
    0     Ab     0      0     2*Ab  0;...
    0      0     Ib     0      0    2*Ib];
% beam axial mass term
Mub = rhob*Ab*Le/6*[...
    2      0      0      1      0      0;...
    zeros(2,6);...
    1      0      0      2      0      0;...
    zeros(2,6)];

Me = Mwb + Mub;
Ke = Kwbb + Kwbs + Kub;

% Has been modified to allow for Rayleigh damping.
if InternalDamping == 1
    Ce = a_hat*Me + b_hat*Ke;
else
    Ce = zeros(6,6);
end
Pe = zeros(6,1);

```

```

% BalamuraganNarayanaDerivation.m
% (created by jak 071603; updated by jak 021804)
% qe = [u1 w1 theta1 gama1 u2 w2 theta2 gama2]'
% ze = [z1 z2]'
syms Ab As Ac tb ts tc Ib Is Ic Gb Gs Gc rhob rhos rhoc
syms Eb Es Ec k_alpha alpha kappa omega chi Le x z h

Nu = [1-x/Le, 0, 0, 0, x/Le, 0, 0, 0];
Nw = [0, 1-x/Le, 0, 0, 0, x/Le, 0, 0];
Ntheta = [0, 0, 1-x/Le, 0, 0, 0, x/Le, 0];
Ngamma = [0, 0, 0, 1-x/Le, 0, 0, 0, x/Le];
% Corrected this on 021404
Nz = [1-x/Le, x/Le];
dNu = diff(Nu, 'x', 1);
dNw = diff(Nw, 'x', 1);
dNtheta = diff(Ntheta, 'x', 1);
dNgamma = diff(Ngamma, 'x', 1);

%h = ts+(tb+tc)/2;

% Beam -----
Kwbb = Eb*Ib*int(dNtheta.*dNtheta,x,0,Le);
Kwbs = Gb*Ab*k_alpha*int([Ntheta; dNw].'*[1 -1]'.*[1 -1]*[Ntheta;
dNw],...
x,0,Le);
Kub = Eb*Ab*int(dNu.*dNu,x,0,Le);
Mwb = rhob*int(Ab*Nw.*Nw+Ib*Ntheta.*Ntheta,x,0,Le);
Mub = rhob*Ab*int(Nu.*Nu,x,0,Le);
% -----

% Piezoelectric layer -----
Kwcb = Ec*Ic*int(dNtheta.*dNtheta,x,0,Le);
Kwcs = Gc*Ac*k_alpha*int([Ntheta; dNw].'*[1 -1]'.*[1 -1]*[Ntheta;
dNw],...
x,0,Le);
Kuc = Ec*Ac*int([dNu; dNtheta; dNgamma].'*[1 -h ts].*[1 -h ts]*...
[dNu; dNtheta; dNgamma],x,0,Le);
Mwc = rhoc*int(Ac*Nw.*Nw+Ic*Ntheta.*Ntheta,x,0,Le);
Muc = rhoc*Ac*int([Nu; Ntheta; Ngamma].'*[1 -h ts].*[1 -h ts]*...
[Nu; Ntheta; Ngamma],x,0,Le);
% -----

% Viscoelastic layer -----
Kwsb = Es*Is*int(dNtheta.*dNtheta,x,0,Le);
Kus = Es*As*int([dNu; dNtheta; dNgamma].'*[1 -(tb+ts)/2 ts/2].'*...
[1 -(tb+ts)/2 ts/2]*[dNu; dNtheta; dNgamma],x,0,Le);
Ks = As*(kappa+alpha*kappa)*int(Ngamma.*Ngamma,x,0,Le);

```

```

Mws = rhos*int(As*Nw.*Nw+Is*Ntheta.*Ntheta,x,0,Le);
Mus = rhos*As*int([Nu; Ntheta; Ngamma].*[1 -(tb+ts)/2 ts/2].'*...
    [1 -(tb+ts)/2 ts/2]*[Nu; Ntheta; Ngamma],x,0,Le);
Kqz = As*alpha*kappa*int(Ngamma.*Nz,x,0,Le);
Kzq = Kqz.';
Kz = As*alpha*kappa*int(Nz.*Nz,x,0,Le);
Mz = As*alpha*kappa/omega^2*int(Nz.*Nz,x,0,Le);
Cz = As*alpha*kappa*2*chi/omega*int(Nz.*Nz,x,0,Le);
% -----

Me = Mwb + Mwc + Mub + Muc + Mus + Mws;
Ke = Kwbb + Kwbs + Kwcb + Kwcs + Kub + Kuc + Kus + Ks + Kwsb;

```

```
function [Mea,Cea,Kea,Pea] = BalamuruganNarayananFEM2(BLP,k_alpha,VLP,CLP,b,Le,InternalDamping);
```

```
% BalamuruganNarayananFEM2.m (created by jak 092803; updated by jak 021904)
```

```
% Refer to Balamurugan, V. and Narayanan, S., 2002, "Finite element formulation and active vibration control study on beams using smart constrained layer damping (SCLD) treatment," Journal of Sound and Vibration, 249(2), pp. 227-250.
```

```
% Beam layer parameters
```

```
Eb = BLP(1);
Gb = BLP(2);
rhob = BLP(3);
tb = BLP(4);
a_hat = BLP(5);
b_hat = BLP(6);
```

```
% Viscoelastic layer parameters
```

```
Es = VLP(1);
rhos = VLP(2);
ts = VLP(3);
alpha = VLP(4);
omega = VLP(5);
chi = VLP(6);
kappa = VLP(7);
```

```
% Constraining layer parameters
```

```
Ec = CLP(1);
Gc = CLP(2);
rhoc = CLP(3);
tc = CLP(4);
d31 = CLP(5);
```

```
Ab = b*tb;
As = b*ts;
Ac = b*tc;
Ib = b*tb^3/12;
Is = b*ts^3/12;
Ic = b*tc^3/12;
h = ts+(tb+tc)/2;
```

```
% Beam -----
Kwbb = Eb*Ib/Le*[zeros(2,8); ...
    0      0      1      0      0      0      -1      0; ...
    zeros(3,8); ...
    0      0     -1      0      0      0      1      0; ...
    zeros(1,8)];
```

```
%Must use a quadrature rule to correct the shear term and prevent shear
```

```

% locking. This matrix matches that in Kwon & Bang, THE FINITE ELEMENT
% METHOD, p. 244, Eq. 8.2.11.
Kwbs = Gb*Ab*k_alpha/4/Le*[zeros(1,8);...
    0      4      2*Le    0      0      -4      2*Le    0;...
    0      2*Le    Le^2    0      0      -2*Le    Le^2    0;...
    zeros(2,8);...
    0      -4      -2*Le    0      0      4       -2*Le    0;...
    0      2*Le    Le^2    0      0      -2*Le    Le^2    0;...
    zeros(1,8)];
%Kwbs = Gb*Ab*k_alpha*[zeros(1,8); ...
%      0      1/Le    1/2     0      0      -1/Le    1/2     0; ...
%      0      1/2     Le/3    0      0      -1/2     Le/6    0; ...
%      zeros(2,8); ...
%      0      -1/Le   -1/2    0      0      1/Le    -1/2    0; ...
%      0      1/2     Le/6    0      0      -1/2     Le/3    0; ...
%      zeros(1,8)];
Kub = Eb*Ab/Le*[ ...
    1      0      0      0      -1     0      0      0; ...
    zeros(3,8); ...
    -1     0      0      0      1      0      0      0; ...
    zeros(3,8)];
Mwb = rhob*Le*[zeros(1,8); ...
    0      Ab/3    0      0      0      Ab/6    0      0; ...
    0      0      Ib/3   0      0      0      Ib/6    0; ...
    zeros(2,8); ...
    0      Ab/6    0      0      0      Ab/3    0      0; ...
    0      0      Ib/6    0      0      0      Ib/3    0; ...
    zeros(1,8)];
Mub = rhob*Ab*Le*[ ...
    1/3    0      0      0      1/6    0      0      0; ...
    zeros(3,8); ...
    1/6    0      0      0      1/3    0      0      0; ...
    zeros(3,8)];
% -----

% Piezoelectric layer -----
Kwcb = Ec*Ic/Le*[zeros(2,8); ...
    0      0      1      0      0      0      -1     0; ...
    zeros(3,8); ...
    0      0      -1     0      0      0      1      0; ...
    zeros(1,8)];
%Must use a quadrature rule to correct the shear term and prevent shear
% locking. This matrix matches that in Kwon & Bang, THE FINITE ELEMENT
% METHOD, p. 244, Eq. 8.2.11.
Kwcs = Gc*Ac*k_alpha/4/Le*[zeros(1,8);...
    0      4      2*Le    0      0      -4      2*Le    0;...
    0      2*Le    Le^2    0      0      -2*Le    Le^2    0;...
    zeros(2,8);...
    0      -4      -2*Le    0      0      4       -2*Le    0;...
    0      2*Le    Le^2    0      0      -2*Le    Le^2    0;...
    zeros(1,8)];

```



```

%Kwcs = Gc*Ac*k_alpha*[zeros(1,8); ...
%      0      1/Le    1/2    0      0      -1/Le    1/2    0; ...
%      0      1/2     Le/3    0      0      -1/2     Le/6    0; ...
%      zeros(2,8); ...
%      0      -1/Le   -1/2    0      0      1/Le    -1/2    0; ...
%      0      1/2     Le/6    0      0      -1/2     Le/3    0; ...
%      zeros(1,8)];
Kuc = Ec*Ac/Le*[ ...
      1      0      -h      ts      -1      0      h      -ts;↙
...
      zeros(1,8); ...
      -h      0      h^2    -h*ts   h      0      -h^2↙
h*ts;...
      ts      0      -h*ts   ts^2    -ts     0      h*ts    -ts^2;↙
...
      -1      0      h      -ts     1      0      -h      ts; ...
      zeros(1,8);...
      h      0      -h^2    h*ts   -h      0      h^2     -h*ts;↙
...
      -ts     0      h*ts   -ts^2   ts      0      -h*ts   ts^2];
Mwc = rhoc*Le*[zeros(1,8); ...
      0      Ac/3    0      0      0      Ac/6    0      0; ...
      0      0      Ic/3    0      0      0      Ic/6    0; ...
      zeros(2,8); ...
      0      Ac/6    0      0      0      Ac/3    0      0; ...
      0      0      Ic/6    0      0      0      Ic/3    0; ...
      zeros(1,8)];
Muc = rhoc*Ac*Le*[ ...
      1/3     0      -h/3    ts/3    1/6     0      -h/6    ts/6;↙
...
      zeros(1,8); ...
      -h/3    0      h^2/3   -h*ts/3 -h/6     0      h^2/6   -↙
h*ts/6; ...
      ts/3    0      -h*ts/3 ts^2/3   ts/6     0      -h*ts/6 ts^2/6;↙
...
      1/6     0      -h/6    ts/6    1/3     0      -h/3    ts/3;↙
...
      zeros(1,8); ...
      -h/6    0      h^2/6   -h*ts/6 -h/3     0      h^2/3   -↙
h*ts/3; ...
      ts/6    0      -h*ts/6 ts^2/6   ts/3     0      -h*ts/3↙
ts^2/3];
Pc = Ec*d31*b*[-1 0 h -ts 1 0 -h ts]';
% -----

% Viscoelastic layer -----
Kwsb = Es*Is/Le*[zeros(2,8); ...
      0      0      1      0      0      0      -1      0; ...
      zeros(3,8); ...
      0      0      -1     0      0      0      1      0; ...
      zeros(1,8)];

```

```

Kus = Es*As/Le*[ ...
      1          0  -(tb+ts)/2    ts/2          -1          0
(tb+ts)/2      -ts/2; ...
      0          0  0              0              0          0
0              0; ...
      -(tb+ts)/2  0  (tb+ts)^2/4  -(tb+ts)*ts/4  (tb+ts)/2  0
-(tb+ts)^2/4  (tb+ts)*ts/4; ...
      ts/2          0  -(tb+ts)*ts/4  ts^2/4          -ts/2          0
(tb+ts)*ts/4  -ts^2/4; ...
      -1          0  (tb+ts)/2    -ts/2          1          0
-(tb+ts)/2    ts/2; ...
      0          0  0              0              0          0
0              0; ...
      (tb+ts)/2  0  -(tb+ts)^2/4  (tb+ts)*ts/4  -(tb+ts)/2  0
(tb+ts)^2/4  -(tb+ts)*ts/4; ...
      -ts/2          0  (tb+ts)*ts/4  -ts^2/4          ts/2          0
-(tb+ts)*ts/4  ts^2/4];
Ks = As*(kappa+alpha*kappa)*Le*[zeros(3,8); ...
      0          0          0          1/3          0          0          0          1/6;
...
      zeros(3,8); ...
      0          0          0          1/6          0          0          0          1/3];
Mws = rhos*Le*[zeros(1,8); ...
      0          As/3          0          0          0          As/6          0          0; ...
      0          0          Is/3          0          0          0          Is/6          0; ...
      zeros(2,8); ...
      0          As/6          0          0          0          As/3          0          0; ...
      0          0          Is/6          0          0          0          Is/3          0; ...
      zeros(1,8)];
Mus = rhos*As*Le*[ ...
      1/3          0  -(tb+ts)/6    ts/6          1/6          0
-(tb+ts)/12    ts/12; ...
      zeros(1,8); ...
      -(tb+ts)/6  0  (tb+ts)^2/12  -(tb+ts)*ts/12  -(tb+ts)/12  0
(tb+ts)^2/24  -(tb+ts)*ts/24; ...
      ts/6          0  -(tb+ts)*ts/12  ts^2/12          ts/12          0
-(tb+ts)*ts/24  ts^2/24; ...
      1/6          0  -(tb+ts)/12    ts/12          1/3          0
-(tb+ts)/6    ts/6; ...
      zeros(1,8); ...
      -(tb+ts)/12  0  (tb+ts)^2/24  -(tb+ts)*ts/24  -(tb+ts)/6  0
(tb+ts)^2/12  -(tb+ts)*ts/12; ...
      ts/12          0  -(tb+ts)*ts/24  ts^2/24          ts/6          0
-(tb+ts)*ts/12  ts^2/12];
Kqz = As*alpha*kappa*Le*[zeros(3,2); ...
      1/3          1/6; ...
      zeros(3,2); ...
      1/6          1/3];
Kzq = Kqz';
Kz = As*alpha*kappa*Le*[ ...
      1/3          1/6; ...

```

```

        1/6      1/3];
Mz = As*alpha*kappa*Le/omega^2*[ ...
        1/3      1/3; ...
        1/6      1/3];
Cz = 2*As*alpha*kappa*Le*chi/omega*[ ...
        1/3      1/6; ...
        1/6      1/3];
% -----

% Element Matrices -----
Me = Mwb + Mwc + Mws + Mub + Muc + Mus;
Ke = Kwbb + Kwbs + Kwcb + Kwcs + Kub + Kuc + Kwsb + Kus + Ks;
% -----

% Augmented Element Matrices and Control Matrix -----
Mea = [Me zeros(8,2); zeros(2,8) Mz];
% Has been modified to allow for Rayleigh damping.
if InternalDamping == 1
    Cea = [(a_hat*Me + b_hat*Ke) zeros(8,2); zeros(2,8) Cz];
else
    Cea = [zeros(8,10); zeros(2,8) Cz];
end
Kea = [Ke -Kqz; -Kzq Kz];
Pea = [Pc; zeros(2,1)];
% -----

```

```
function [kk] = FEASMBL(kk,k,index)
%-----
% Purpose:
%   Assembly of element matrices into the system matrix
%
% Note:
%   This has not be changed from the original. This is OK
%   because the dofs associated with dissipation coordinates
%   are not added between elements due to indices.
%
% Synopsis:
%   [kk] = feasmb11(kk,k,index)
%
% Variable Description:
%   kk - system matrix
%   k  - element matrix
%   index - d.o.f. vector associated with an element
%-----

edof = length(index);
for i = 1:edof
    ii = index(i);
    for j = 1:edof
        jj = index(j);
        kk(ii,jj) = kk(ii,jj) + k(i,j); % adds the element matrix to
the system matrix
    end
end
end
```

```
function [ff] = FEASMBL2(ff,f,index)
%-----
% Purpose:
%   Assembly of element vector into the system vector
%
% Synopsis:
%   [ff]=feasmb2(ff,f,index)
%
% Variable Description:
%   ff - system vector
%   f  - element vector
%   index - d.o.f. vector associated with an element
%-----

edof = length(index);
for i = 1:edof
    ii = index(i);
    ff(ii) = ff(ii)+f(i);
end
```

% This file is the same as schurmr with the exception of matrices  
 % [SLBIG] and [SRBIG] included at the end.  
 % Only the description of the command and the modifications made to  
 % schurmr.m are printed out.

function [Gred,info] = schurmrJAK(G,varargin)

```

%SCHURMR Schur balanced truncation model reduction for
continuous/discrete
%         & stable/unstable plant.
%
%[GRED,REDINFO] = SCHURMR(G,ORDER,KEY1,VALUE1,KEY2,VALUE2,...) performs
% Schur balanced truncation model reduction on LTI system G such
that the
% sigma plot of the error (GRED - G) < 2*(the sum of the discarded
% states' Hankel singular values). For unstable G the algorithm
works
% by first splitting G into a sum of stable and antistable part,
% reduces the stable part, then add the unstable part back for
% the final output G. The unstable part is kept as default.
%
% Inputs:
% G - LTI system to be reduced
%
% Optional inputs:
% ORDER - an integer array with desired order of reduced model.
% A batch run of [m:n] can be specified for a pack of reduced
% order models to be generated.
%
% KEY      | VALUE      | MEANING
% -----|-----|-----
% 'MaxError' | real no. or | Reduce to achieve H-infinity error
%            | vector      | If present, 'MaxError' overrides ORDER
input.
% -----|-----|-----
% 'Weights'  | {Wout,Win}  | Optional 1x2 cell array of LTI weights
Wout
%            | cell        | (output)and Win (input); default is both
%            |            | identity. Weights have to be invertable.
% -----|-----|-----
% 'Display'  | 'off' or 'on'| Display HANKELSV plots (default 'off')
%
% Outputs:
% GRED      - LTI reduced order system
% REDINFO   - a struct of REDINFO.ErrorBound, REDINFO.StabSV,
% REDINFO.UnstabSV, REDINFO.SRBIG, and REDINFO.SLBIG

% R. Y. Chiang & M. G. Safonov 4/15/02
% Copyright 1988-2004 The MathWorks, Inc.

```

```

    Type = Typeoriginal;
end

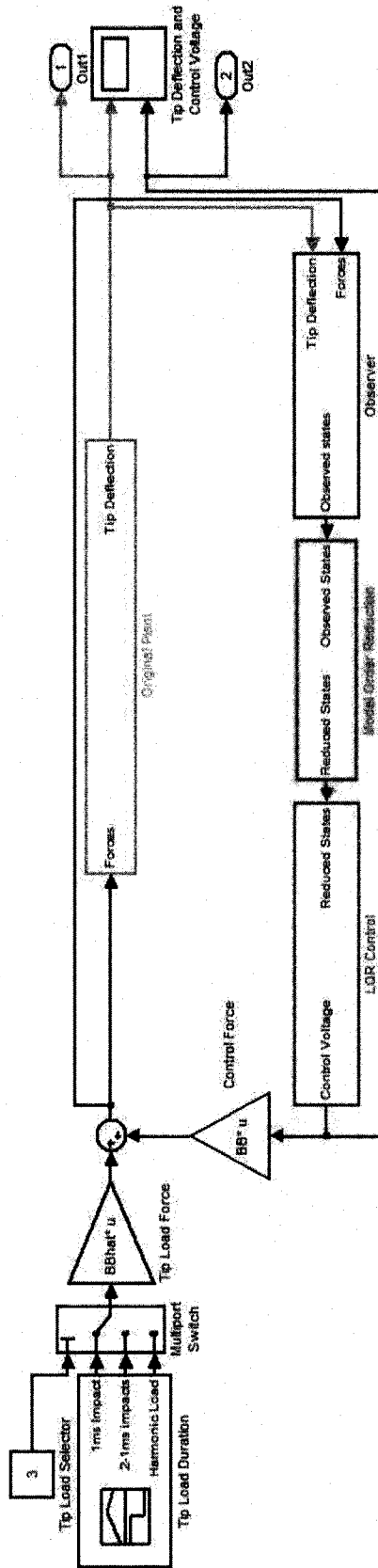
stabexe = length(redval{1,1});
Ebound = zeros(stabexe,1);
for i = 1:stabexe
    if redval{1,1}(i) < 0
        warning('desired order(Gred) < order(Gunstable+Gjw), assign
(Gunstable+Gjw) to Gred.')
        redval{1,1}(i) = 0;
    end
    A = G.a;
    B = G.b;
    C = G.c;
    D = G.d;
    [Ar,Br,Cr,Dr,aug,hhsv,SLBIG,SRBIG] = schbal(A,B,C,D,Type,redval
{1,1}(i));
    gr = ss(Ar,Br,Cr,Dr);
    if wtflag % exists weighting
        ggrr = wtval{1,1}'*(gr+Gus)*wtval{1,2}';
        [gr,gtemp] = stabproj(ggrr);
        gr = gr + get(gtemp,'d');
    end
    gr = gr+P+Gjw; % add back the unstable part of G
    if Ts, % discrete case
        gr = bilin(gr,1,'S_Tust',[Ts,1-paaz]);
        gr.Ts = Ts0;
    end
    Gred(:, :, i) = gr;
    Ebound(i) = aug(2);
    info.ErrorBound = Ebound;
    info.StabSV = [inf*ones(no_jw_pole,1);hhsv];
    [rA,cA] = size(A);
    if (prod(size(info.StabSV)) < rA)
        info.UnstabSV = hksv(P);
    else
        info.UnstabSV = [];
    end
    info.SRBIG = SRBIG;
    info.SLBIG = SLBIG;
end

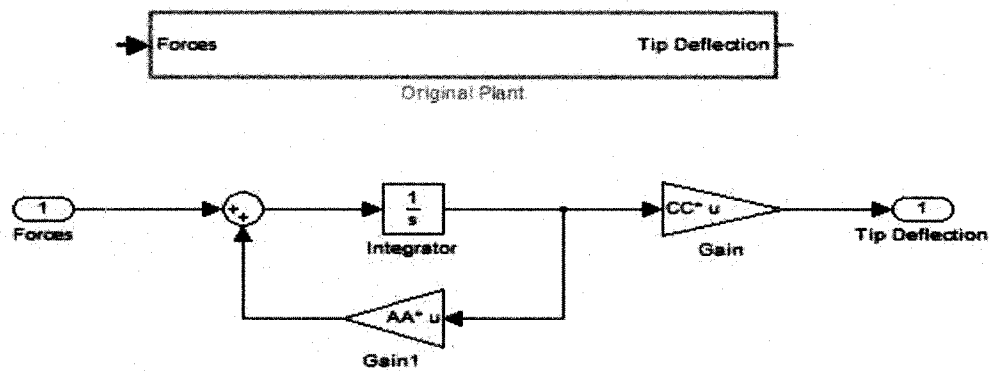
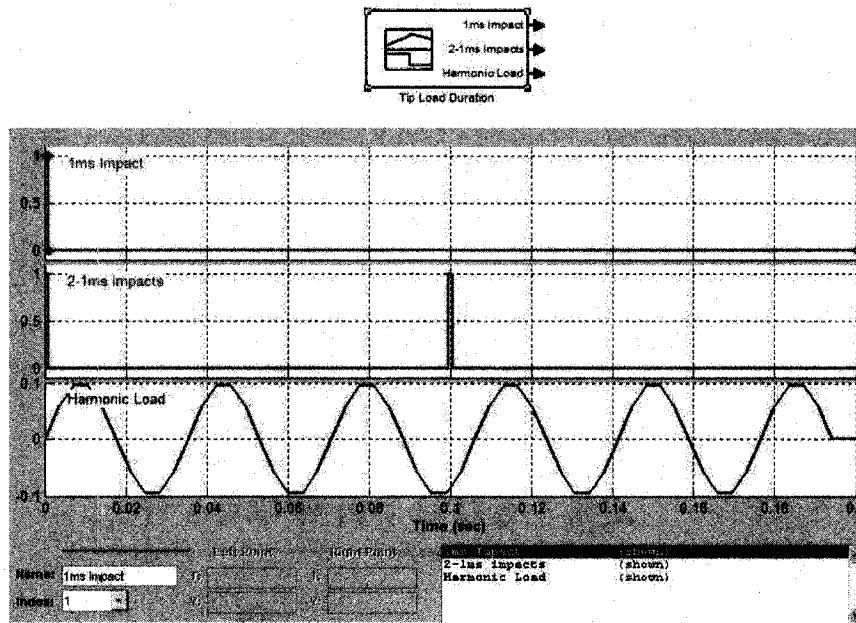
% ----- End of SCHURMR.M % 10/26/02

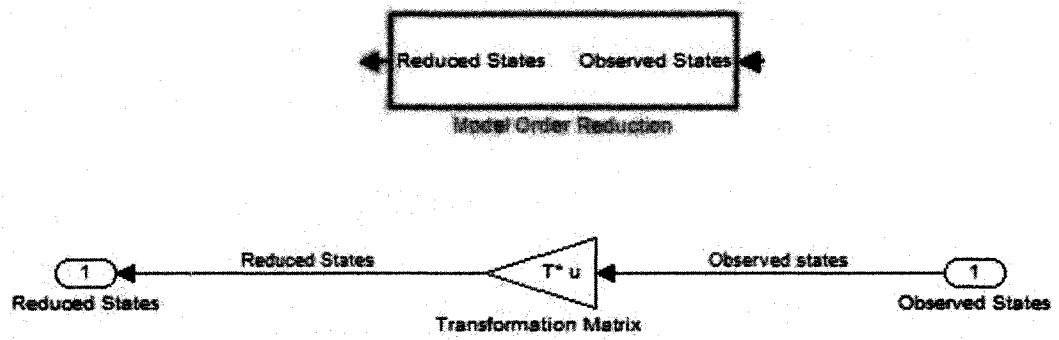
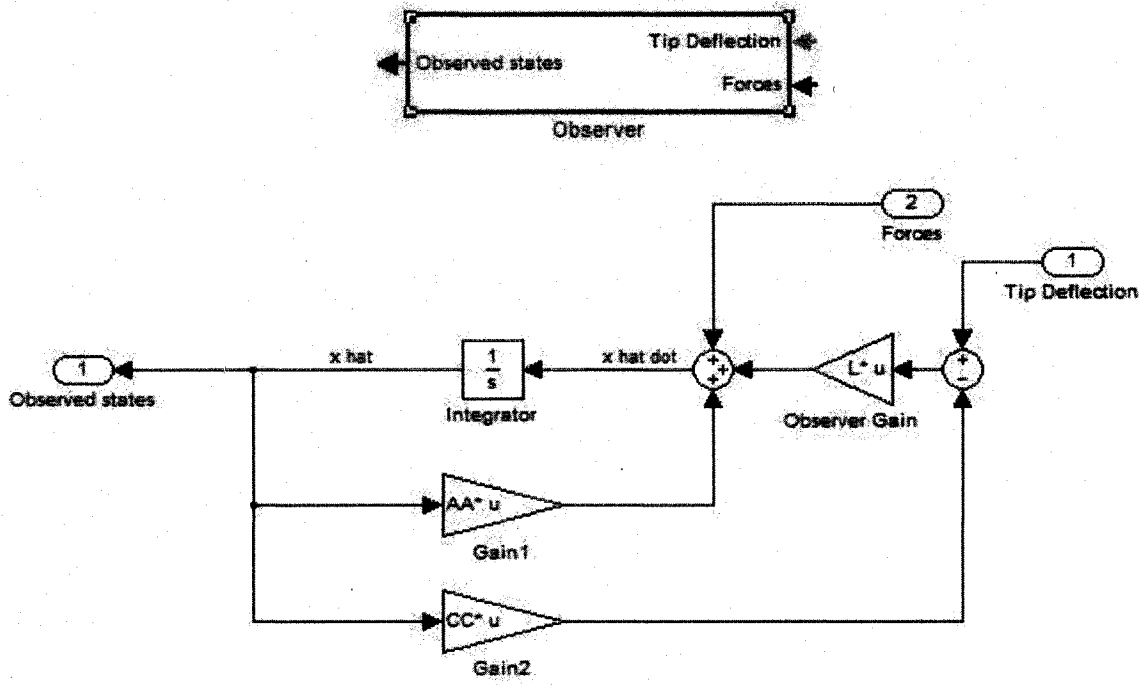
```

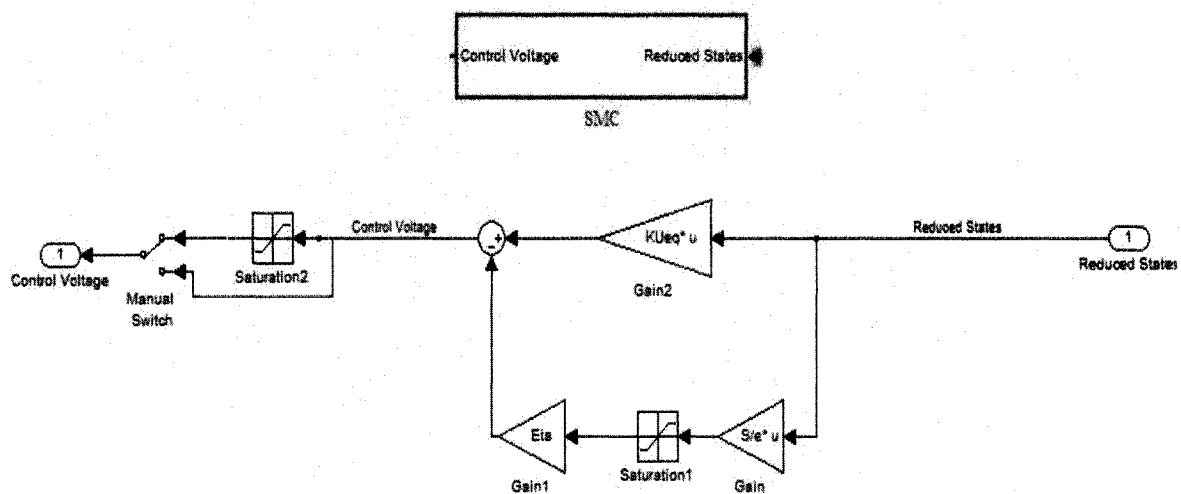
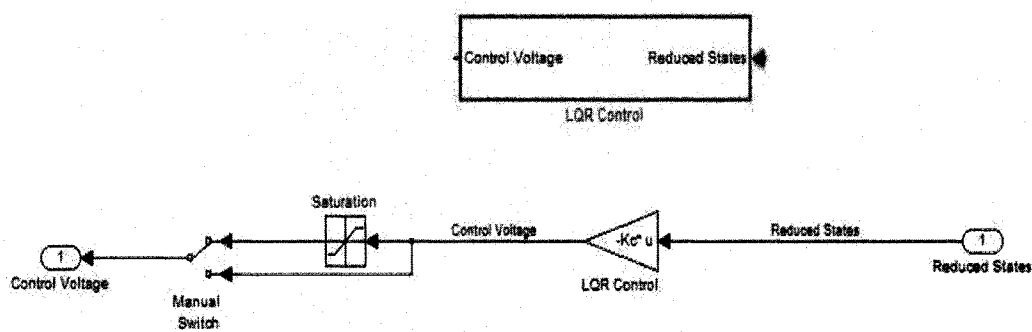
**APPENDIX B**  
**SIMULINK FILES**











## VITA

Jose J. Rodriguez was born on December 12, 1980, to John and Norma Rodriguez in Baytown, Texas. Jose was raised in San Juan, Texas and graduated from P.S.J.A North high school as a Texas Scholar in 1999. Following high school, Jose received his Bachelor of Science in Mechanical Engineering from the University of Texas-Pan American, graduating cum laude in 2003. Jose enrolled into the graduate program at the University of Texas-Pan American in January of 2004.

### Education:

University of Texas - Pan American, Edinburg, Texas (August 1999 - December 2003)

Bachelor of Science in Mechanical Engineering

Major: Mechanical Engineering Minor: Mathematics

GPA: Overall 3.48, Engineering Curriculum 3.88

University of Texas - Pan American, Edinburg, Texas (January 2004 – December 2005)

Master of Science in Mechanical Engineering

Major: Mechanical Engineering

Thesis: Comparison of Sliding Mode Control (SMC) and State-Feedback Control Applied to a Partially Treated Actively Constrained Layer Damped (ACLD) Beam

GPA: 3.91