ROBUST STATE ESTIMATION FOR THE CONTROL OF FLEXIBLE ROBOTIC MANIPULATORS

A Thesis
Presented to
The Academic Faculty

by

Brian Karl Post

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
The George Woodruff School of Mechanical Engineering

Georgia Institute of Technology
December 2013

ROBUST STATE ESTIMATION FOR THE CONTROL OF FLEXIBLE ROBOTIC MANIPULATORS

Approved by:

Wayne J. Book, Advisor
George W. Woodruff School of
Mechanical Engineering
Georgia Institute of Technology

William Singhose
George W. Woodruff School of
Mechanical Engineering
Georgia Institute of Technology

Magnus Egerstedt
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Patricio Vela
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Aldo Ferri
George W. Woodruff School of
Mechanical Engineering
Georgia Institute of Technology

Date Approved: July 15, 2013

To my wonderful wife Sarah

and my parents Susan and Mac

for their constant support and guidance.

Thank you.

ACKNOWLEDGEMENTS

Many people have supported me in this work, and without them its completion would not have been possible. First and foremost, I'd like to express my deepest appreciation to my advisor, Dr. Wayne J. Book, who has both provided financial support and offered me invaluable guidance and wisdom throughout the course of this work. He has been a true mentor to me and it has been a pleasure working for him.

I would like to thank Dr. Aldo Ferri, Dr. William Singhose, Dr. Magnus Egerstedt, and Dr. Patricio Vela for their service on my dissertation committee and for the helpful comments they have offered. Special thanks are given to Dr. Steve Dickerson and JD Huggins. Dr. Dickerson shared his expertise with me. J.D. Huggins offered his counsel, especially with regard to my experimental work. He has always been patient with me and has been a sustaining force in the lab. I well recognize that this project would not have been successfully completed without his help.

In addition I would like to thank CAMotion Inc. and National Instruments for their generous hardware donations.

I would also like to offer my sincere appreciation to all of the students of the Intelligent Machine Dynamics Laboratory including: Hannes Daepp, Dr. Mark Elton, Dr. Aaron Enes, Heather Humphreys, Dr. Longke Wang, and Dr. Ryder Winck for their friendship, camaraderie, and for being a continual sounding board for ideas, suggestions, or support whenever I needed it most.

Finally, I would like to thank my wife Sarah, my parents, and my friends for believing in me and sustaining me through this process with their unending love and support.

Contents

List of Tables

List of Figures

SUMMARY

In this thesis, a novel robust estimation strategy for observing the system state variables of robotic manipulators with distributed flexibility is established. Motivation for the derived approach stems from the observation that lightweight, high speed, and large workspace robotic manipulators often suffer performance degradation because of inherent structural compliance. This flexibility often results in persistent residual vibration, which must be damped before useful work can resume. Inherent flexibility in robotic manipulators, then, increases cycle times and shortens the operational lives of the robots. Traditional compensation techniques, those which are commonly used for the control of rigid manipulators, can only approach a fraction of the open-loop system bandwidth without inducing significant excitation of the resonant dynamics. To improve the performance of these systems, the structural flexibility cannot simply be ignored, as it is when the links are significantly stiff and approximate rigid bodies. One thus needs a model to design a suitable compensator for the vibration, but any model developed to correct this problem will contain parametric error. And in the case of very lightly damped systems, like flexible robotic manipulators, this error can lead to instability of the control system for even small errors in system parameters.

This work presents a systematic solution for the problem of robust state estimation for flexible manipulators in the presence of parametric modeling error. The solution includes: 1) a modeling strategy, 2) sensor selection and placement, and 3) a novel, multiple model estimator. Modeling of the FLASHMan flexible gantry manipulator is accomplished using a developed hybrid transfer matrix / assumed modes method (TMM/AMM) approach to determine an accurate low-order state space representation of the system dynamics. This model is utilized in a genetic algorithm optimization in determining the placement of MEMs accelerometers for robust estimation and observability of the systems flexible state variables.

The initial estimation method applied to the task of determining robust state estimates under conditions of parametric modeling error was of a sliding mode observer type. Evaluation of the method through analysis, simulations and experiments showed that the state estimates produced were inadequate. This led to the development of a novel, multiple model adaptive estimator. This estimator utilizes a bank of similarly designed sub-estimators and a selection algorithm to choose the true value from a given set of possible system parameter values as well as the correct state vector estimate. Simulation and experimental results are presented which demonstrate the applicability and effectiveness of the derived method for the task of state variable estimation for flexible robotic manipulators.

Chapter I

INTRODUCTION

1.1 Motivation for Research

While advances in control system design have enabled more efficient and accurate manipulation, industry has been reluctant to embrace lightweight and large workspace robotic manipulators. The demand for high precision in both static positioning and dynamic motion has driven manipulator design and resulted in massive, inefficient, expensive, and potentially dangerous robot arms. In making the structures of these robots lighter we could achieve higher speed manipulation, while reducing energy consumption, and providing a safer working environment for operators. Increased speed, workspace size and reduced purchasing and operating costs allow for interesting and appealing applications of lightweight manipulator technology.

Relatively few examples of lightweight manipulators exist in modern industry, despite the fact that this has been a subject of research interest for many years. Current lightweight manipulators are largely used in applications where external factors preclude excess weight.The space industry provides one example. Robots required to manipulate objects in space must be light enough to be launched by rocket into orbit. Though light in weight, these manipulators lack modern control methods that compensate for the inherent structural flexibility of the system. Instead, control bandwidth is limited and sufficient time is allowed for vibration to subside, up to 30% of total operation time [6].

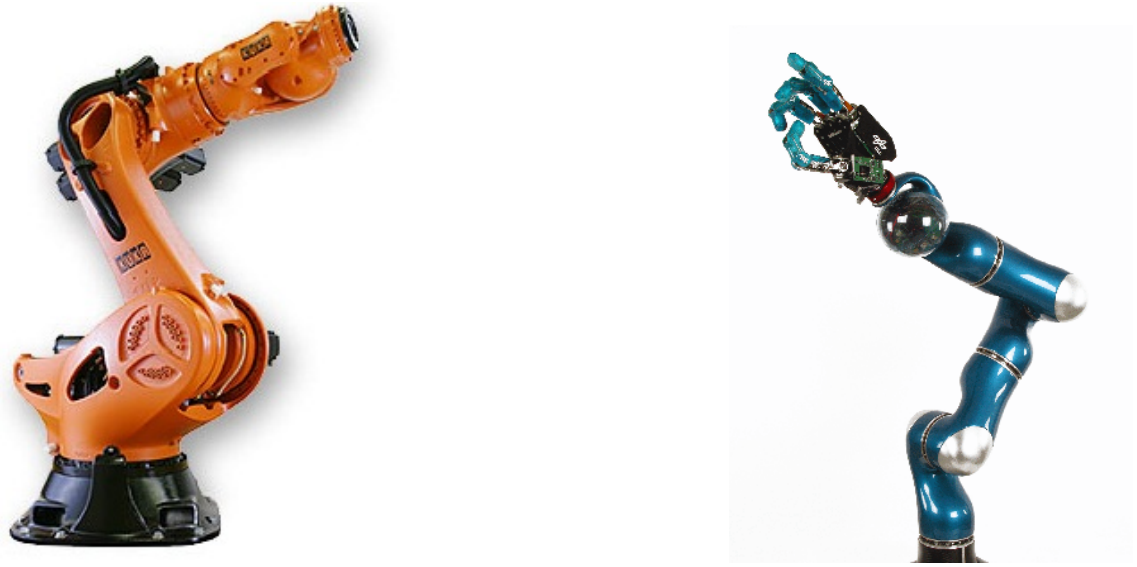
The modern control techniques for creating vibration limiting motion of flexible structures have not been widely applied in industry because they are impractical for the complexities of industrial implementation. Especially lacking is an adequate method for the reconstruction of system states from practical low cost sensing systems. Assuming this state information can be retrieved in an efficient, accurate, and practical manner, many of the already developed control algorithms could be used to enable highly capable lightweight

and long-reach robots. This is the goal of this dissertation project.

1.1.1 State of the Art Industrial Manipulators

Since their initial introduction in the 1960s, significant advancements in industrial manipulators have been made. However they still, with relatively few exceptions, rely primarily on joint sensing to determine their position and orientation in the workspace. Thus, in order to reliably position the end effector, the links which comprise the robot must be rigid, and the joints where actuation takes place are designed to exhibit very little compliance [14,101]. These conditions are obtained through structural design employing strong, heavy materials and reducing the link lengths.

Ultimately these design requirements necessitate that large powerful motors and drives be used to actuate these massive structures. These motors and drives not only require more energy to operate but come with larger initial costs [14]. Since a larger percentage of the torque produced from the drive is reserved for motion of the structure itself, less is allotted to the acceleration and motion of the payload. This results in reduced payload capacities.

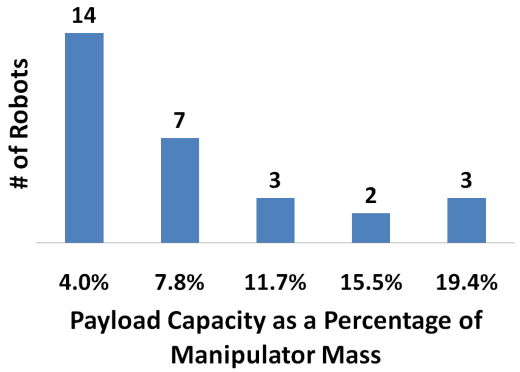


(a) Industrial Manipulator (b) Lightweight Manipulator

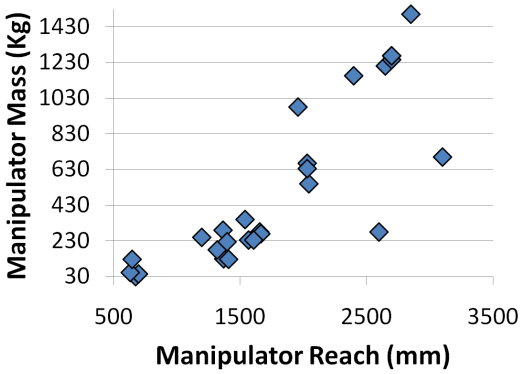
Figure 1: Traditional and Lightweight Manipulators[1]

[1]Sources: http://www.robots.com, http://www.scholarpedia.org

Examination of the current products (Figure 1(a)) of four major industrial robot manufacturers, Fanuc, ABB, Kuka, and MotoMan (including twenty nine specific robot models sampled from low to high payload capacity lines of each brand), indicates that most operate with payloads less than 10% of the total manipulator mass. A significant portion of these require payloads of 5% or less. The higher ratio robots in Figure 2(a) are found in heavy duty manipulator categories with payload capacities greater than 150kg. These represent a small but significant portion of the total market.



(a) Comparison of Payload/Total Manipulator Mass Ratios (upper-bound) for Standard Industrial Robots



(b) Comparison of Reach to Total Manipulator Mass

Figure 2: A Survey of Commercially Available Robotic Manipulators

Workspace size is also sacrificed for the sake of rigidity, as increases in link stiffness are necessary to prevent loss of accuracy and repeatability under static deflection and vibration during motion. As shown in Figure 2(b), increases in workspace size result in increased mass.

The excess mass means that these systems carry significant amounts of energy when in motion, and any collision with the environment or operator can certainly be destructive, if not life threatening. Therefore all human interaction with these massive manipulators is strictly prohibited and safety barriers are erected to separate the operators from the robots.

1.1.2 Benefits of Lightweight High-speed Manipulators

Reducing the mass and extending the reach of these robotic systems improves system performance by allowing more of the power generated by the motors to be transformed into

motion of the payload rather than the structure itself. This results in greater accelerations and reduced cycle times [14]. Extended reach also enables greater utility as workspace size increases allowing manipulation of larger objects and promoting more cost efficient automation solutions. With lighter weight components, energy consumption is reduced. Thus smaller motors and drives can achieve nearly identical performance metrics to that offered by the more massive manipulators. The need for expensive high performance actuators is thereby reduced along with the high energy costs.

While most industrial manipulators are certainly too heavy for mobile applications and require large sturdy mounting platforms, light weight manipulators have seen use in many applications where total mass is limited. Examples include explosive ordinance disposal robots, space robotics, and nuclear waste mitigation.

By reducing the mass of the manipulator, the system is made less rigid and more compliant, resulting in a manipulator more suited for robots in contact with fragile environments. Examples include a safer robot for washing windows or interacting with human operators [97]. While natural systems are well equipped for controlling compliant structures with extensive sensing systems and proprioception (e.g. positioning control of human arms via compliant muscles and flexible joints), it is much more difficult to accurately control a flexible robot than a rigid one.

1.1.2.1 Motivating Example: Large Scale Additive Manufacturing

Large scale additive manufacturing, an example of which is illustrated in Figure 1.1.2.1, refers to the automated process of incrementally building large parts. Akin to 3D printing on a small scale, forays into large scale additive manufacturing are aimed at producing large parts out of traditional and novel materials with nearly unlimited complexity. Beyond seemingly unlimited design freedom, additive manufacturing on large scales reduces waste, and labor and tooling costs when compared with traditional manufacturing methods.

Solid free form construction is a large scale application where a special mix of concrete is used to form a wall one layer at a time. A truly additive process, very little construction waste is created, making it environmentally preferable. Being a nearly automated

Figure 3: Large Scale Additive Manufacturing[2]

process,well suited for rapid construction, the technique could be used to provide temporary housing to disaster victims and permanent shelter for inhabitants of third world countries [51]. One of the primary barriers to development in this new and burgeoning field are suitable large workspace manipulators, with adequate performance characteristics. Build size, time, and quality are directly dependent on the workspace, speed, accuracy, and precision of the robotic manipulator used.

1.1.3 General Challenges

Structural flexibility in robotic systems comes in numerous forms and poses many control challenges. Usually concentrated in the joints, links, and/or base, this flexibility can result in a loss of accuracy and precision, excess vibration, or in some cases instability [14].

Using only internal joint sensing it is very difficult to determine the amount of deflection in each link. This results in inaccurate end effector position measurements. Since the actuators and rate sensors are typically collocated at the joints, the vibration resulting from motion will persist. No corrective compensation can be made with non-backdrivable actuators, because no significant error is observable at the joints. Estimates of the true flexible system states are necessary to correct for any errors between the desired and actual motion of the manipulator.

[2] http://www.artecreha.com/

While Proportional Derivative (PD) or (PID) control is typically used for rigid manipulators, the dynamics of a flexible manipulator exhibit non-minimum phase characteristics. Given a step command in position, the non-minimum phase behavior will result in a tip deflection in the opposite direction. Thus a suitable controller must be designed to compensate for this effect [14].

Aggressive trajectories, like trapezoidal velocity profiles, work well for rigid systems as they achieve the desired motion in a time optimal fashion. However, these aggressive trajectories are capable of exciting the resonant structural dynamics resulting in undesirable and potentially damaging vibration. Thus, the generation of vibration limiting commands is a necessary component of the control structure [80].

Assuming these challenges can be overcome, long-reach light-weight robots could improve current automation processes and enable robotic solutions for fields where the current state of technology makes automation impractical or even impossible.

1.2 Problem Statement

Stable and precise control of flexible manipulators requires accurate knowledge of flexible state variables, which are difficult to measure directly. Accurate estimates of these system states are predicated on the accuracy and reliability of the model used by the estimator. While certain system parameters are well known, many can be difficult to accurately measure or change with time. For example, in almost any use scenario, the manipulator payload will change over a cycle. For the the control system to remain stable and ensure reliable operation, the estimation routine must be robust to these errors and continue to provide accurate state information.

The ability to maintain accurate state estimates under conditions of parametric uncertainty and error is therefore important to the practical success and industrial application of light-weight, long-reach, high-speed, or otherwise flexible robotic manipulators. This research is targeted toward the solution of this problem by extending the utility of estimation algorithms to a broader range of parametric modeling error. The ultimate goal of the research is to have developed a systematic approach for modeling the flexible system,

selecting and placing sensors, and leveraging its modal properties such that an estimator design which preserves state accuracy over a large operational range is achieved

1.3 Research Overview

The proposed estimation strategy is the product of systematic approach, broken down into three separate research problems. First, one must select a suitable modeling strategy which adequately parametrizes the system dynamics in terms of useful state variables to be estimated. Second, selecting and placing sensors on the structure for optimal state reconstruction. Third, there must be effective utilization of the sensing and model information in the production of accurate state estimates.

1.3.1 Dynamic Modeling of Flexible Manipulators

The structural compliance of an elastic system permits a material deformation under both external and inertial forces. Permitting bending without failure, the structural elasticity allows the system to return to its pre-loaded state when the forces are removed. This structural compliance also increases the mathematical complexity of models used to describe the dynamic behavior of the system. Since the elasticity in a flexible link manipulator is distributed along the individual beam elements which comprise the manipulator structure, distributed parameters are used to characterize the behavior [39]. Consisting of an infinite number of degrees of freedom, the partial differential equations which describe the motion of the structure are very difficult to solve in closed form. Thus, numerical methods are often used to approximate the system dynamics for control and estimation purposes.

The modeling approach taken in this work is a hybrid strategy of frequency and time domain modeling techniques. Natural frequencies and basis functions are determined through the frequency domain transfer matrix method (TMM), and the time domain assumed modes model (AMM) used for control and estimation is seeded with these basis functions to reduce the systemic error associated with the assumption of polynomial or trigonometric functions in the AMM approach. The result is an accurate, low order, time domain approximation of the system dynamics for a finite number of vibratory modes. Given the finite bandwidth

of practical control systems and the reduced contributions of higher order modes this truncation is often justified. The result is unwanted behavior including excess vibration and in some cases instability. Potential spillover is addressed in this work through redundancy and sensor placement.

1.3.2 Selection and Placement of Sensors

For rigid manipulators, position sensors at the joints are sufficient to determine the full state vector of the system. The addition of flexible links adds additional states to the model. Bending and torsional modes of vibration are a direct consequence of compliant links. Adding these modes of vibration to the system model results in two additional state variables for each mode considered. These states are neither directly measurable or observable from non-backdrivable joints. Therefore, extra sensors are necessary to recover these states.

Traditionally a strain gauge mounted at the base of the link is utilized for estimating the flexible state variables [42]. While effective for estimating the first mode of vibration, this placement is not optimal for multiple modes. In this work, the observability gramian and observability test matrix are utilized for developing an n-sensor, where n is the number of modes considered. This sensor placement strategy which maximizes the measurement/state energy correlation, ensuring maximizing signal to noise ratios for sensing each mode of vibration. A tradeoff between optimal and robust sensor placements is also considered. Low cost MEMs accelerometers are used as the primary sensing apparatus in this work, and while a large portion of the analysis is performed with respect to this sensor choice, the results can be generalized and applied to strain gauges, or other measurement devices.

1.3.3 Robust State Estimation

Parametric error is always present when constructing a model of any complex dynamic system. Excess modeling error can be detrimental or destabilizing when inaccurate parameters are used in control system design. Traditional estimators like Luenberger observers and Kalman filters have been shown to suffer dynamic and static state error when the models

used are errant. Sliding mode techniques have been reported as a solution to these problems, as they extend the bounds of tolerable parametric error. Predicated on a Lyapunov stability analysis, these methods use a large gain switching algorithm to ensure estimator stability in the presence of bounded nonlinearities and parameter uncertainty [103]. Sliding mode estimation is evaluated for use in flexible manipulator control and deemed a marginal improvement in estimation convergence rate at the expense of noisy state estimates. An alternative solution is proposed wherein multiple estimators built on parametrically dissimilar models are used and evaluated relative to each-other to determine the correct state vector.

1.4 Thesis Overview

The work presented in this thesis has been driven by fundamental work completed in the fields of flexible manipulator control and estimation theory. To that end, the presentation of completed work will begin with the fundamental concepts and available literature on the subject, and progress toward the development of systematic approach to the robust estimation of flexible state variables for flexible link manipulators.

Chapter II is a review of background material, a discussion of relevant fundamental literature, and presents the contributions of this work to the larger body of literature in the field. Chapter III details the development of the system models used throughout the subsequent chapters. Chapter IV presents a systematic approach to choosing sensors and their placements for optimal and robust performance in estimation routines. Chapter V is a discussion of the general estimation problem, the role of parametric modeling error, and an evaluation of robust observation techniques as tools to combat modeling error. A solution to the parametric modeling error problem is proposed and evaluated in chapter VI. Finally, chapter VII concludes the work with a summary and suggested future work.

Chapter II

BACKGROUND AND REVIEW OF PREVIOUS WORK

This review consists of a discussion of background material, foundational work, and variations of techniques applied in this thesis.

2.1 Flexible Manipulator Control

While feed forward methods work well for planning vibration limiting trajectories, external disturbances and modeling uncertainties will result in excitation of oscillatory dynamics. Feedback control can be used to compensate for these effects, thereby improving transient performance and reducing steady state error.

The appropriate choice of feedback control is imperative to the desired goal of achieving accurate high speed motion of flexible manipulators. The choice of feedback controllers also determines the specific form of feedback information that will be necessary to enact control(e.g. output vs. state feedback, and the performance targets for the estimator such as bandwidth, state error and robustness to noise). Thus the design of the state observer cannot be entirely divorced from the control design, necessitating the appropriate choice of control methodology.

2.1.1 PID Control

As previously described, PID control architectures are used in nearly all industrial manipulators. As described in [14], the control bandwidth achievable for a flexible arm using this method is approximately one third of the clamped joint natural frequency. As work-spaces increase, robot mass decreases, or payload mass increases, the system natural frequency becomes lower which limits the achievable control bandwidth using this method.

2.1.2 Modern Control Methods

A number of feedback compensation approaches are available for the control of flexible manipulators. Robust control methods like H_∞ [87] and sliding mode control [50] have been applied with relative success, as they maintain stability with minor reductions in achievable performance. Adaptive algorithms have also been applied to improve the controller performance given model uncertainties or time varying system parameters [34]. However, adaptive control with no compensation for the flexible behavior yields little improvement. More recently, fuzzy logic [18] and neural network based control schemes [85] have been applied to the problem. Singular perturbation techniques have been shown effective for a class of manipulators by separating the control problem into two distinct time scale problems, a slow trajectory tracking problem and a fast vibration suppression problem [14]. Linear state feedback control is, perhaps, the most well developed and promising approach, but requires modeling simplifications of the continuum and nonlinear structure of the flexible system. All techniques developed to date, however, rely on estimates of the flexible system states or end effector position measurements to provide the appropriate control effort.

2.2 Modeling Methods for Flexible Robots

State estimation for the control of flexible manipulators is predicated on the availability of an accurate system model with a complete characterization of the internal states of the flexible system. For traditional robotic manipulators the internal states associated with deflection of the link members are small and can be neglected in most applications, given that speeds are low and manipulator stiffness is high.

2.2.1 Traditional Rigid Robotic Manipulators

For rigid manipulators, the equations of motion (1), in the joint space form, can be represented as a function of the joint variables and their derivatives which are often easily measurable through encoders and tachometers:

$$M(q)\ddot{q} + C(q,\dot{q}) + \tau_g(q) = \tau \tag{1}$$

Derived in [86, 93] $M(q)$ is the inertia matrix, $C(q, \dot{q})$ is the dissipation matrix, q is the vector of generalized coordinate vector composed entirely of joint variables, $\tau_g(q)$ is the vector of joint torques caused by gravity, and τ is the vector of input torques at the joints. In reality, however, no structure is truly rigid and (1) is merely an approximation that is valid under limited conditions. Cetinkunt and Book [20, 22] outlined procedures for determining when this approximation is valid for joint feedback control systems. A critical ratio of undamped natural frequency of the controlled rigid system and lowest clamped structural natural frequency ω_R/ω_C is used to confirm or invalidate the rigid system model. For controller bandwidths much less than one third of the clamped natural frequency, the assumption holds, but for significant increases in control gains, the critical ratio increases beyond an acceptable level and the achievable damping ratio decreases and the closed loop system eigenvalues no longer reach the real axis, which is indicative persistent vibration of the end effector.

2.2.2 Joint vs. Link Flexibility

Flexibility can be considered as concentrated in the joints or distributed throughout the structure a robotic manipulator [27]. Joint flexibility originates from elastic elements in the drive train of a manipulator. In recent years, series-elastic actuators have been utilized in robot arms intended for human robot interaction to enhance the compliance of a manipulator in contact with delicate objects. Increased joint flexibility requires additional generalized coordinates associated with the deflection of the flexible elements (θ - motor displacement, q - link displacement). The dynamic equations of motion are, therefore:

$$\begin{aligned} M(q)\ddot{q} + C(q, \dot{q}) + \tau_g(q) + K(q - \theta) &= 0 \\ B\ddot{\theta} + K(\theta - q) &= \tau \end{aligned} \tag{2}$$

where K is a matrix of joint stiffnesses and B is a matrix of rotor inertias from the drive motors [26, 27]. De Luca and Flacco [28] designed and simulated a PD regulation control algorithm that utilizes the additional generalized coordinates. In the case of flexible joints, these additional variables are easily measured with additional encoders and tachometers.

In contrast to flexible joint manipulators, the additional generalized coordinates associated with distributed link flexibility arise from an infinite length series expansion, and are only indirectly measurable. In practice this infinite series is truncated to a finite number of modes. Alberts et al. [2, 3] studied the addition of viscoelastic damping layers to the link structures as a mechanism of improving the achievable performance of flexible link manipulators. The approach significantly reduced the contributions of higher order modes in the system response and further reduced control requirements to the active damping of only one or two modes of vibration. Even with applied damping treatment, the residual vibration associated with the controlled motion of flexible manipulators must be measured in order to affect active vibration control. State observers are, thus, needed to recover the required state variables [42].

2.2.3 Time Domain Models

Time domain models in the form of time dependent ordinary differential equations are desirable for estimation of flexible robotic manipulators, as they can be directly converted to state space form and implemented on deterministic real-time systems.

2.2.3.1 Field Descriptions of Vibrating Beams

Flexible link manipulators are, essentially, collections of beams connected through actuated joints. Therefore, the study of the manipulator's structural vibration can be directly linked to the study of vibrating beams. Links are often assumed to be Euler-Bernoulli beams as in [25] because of their large length to diameter ratios. However researchers, to a lesser extent, have also applied Timoshenko beam theory in an attempt to characterize flexible "stubby" links [13]. Exact displacement field descriptions of the structure can only be obtained through analysis of a continuous system with an infinite number of degrees of freedom. This requires the solution of a fourth-order boundary-value problem in the case of a bending bar.

Let x be the location of a point along the axial coordinate of a bar, $w(x,t)$ be the transverse displacement of that point, and f(x,t) be the transverse force at that location. The system parameters are $m(x)$, mass per unit length, and the flexural rigidity EI, where

E is Young's Modulus and I is the area moment of inertia. Then the deflection at any point along the beam satisfies:

$$-\frac{\partial^2}{\partial x^2}\left[EI(x)\frac{\partial^2 w(x,t)}{\partial x^2}\right] + f(x,t) = m(x)\frac{\partial^2 w(x,t)}{\partial t^2} \tag{3}$$

.

From [39, 71], a general solution to the unforced system differential equation obeys the form:

$$W(x) = C_1 \sin\beta x + C_2 \cos\beta x + C_3 \sinh\beta x + C_4 \cosh\beta x \tag{4}$$

,

where $\beta^4 = \frac{\omega^2 m}{EI}$

The natural modes can be determined, by applying the four geometric and natural boundary conditions. While providing the greatest accuracy, this modeling approach is often unfeasible for the study of flexible manipulators. More often than not, no closed form solution like (4) is available due to nonuniform mass and stiffness distributions [71], violations of the thin beam assumption, damping, or other effects. Therefore, most solutions for practical systems are computationally burdensome, if solvable at all.

2.2.3.2 Lumped Parameter Approximations

The distributed nature of the mass and stiffness of a flexible manipulator link has been approximated in significantly different ways. First, there is the lumped parameter approximation where the mass and stiffness elements are assumed to be uniformly distributed along the link. Here each lump represents a point mass with attached elastic elements. Book and Majette [16] compared the natural frequencies predicted through lumped parameter approximations versus continuous methods. They found that the series length has a large impact on the accuracy of this model. Book [11] developed 4×4 transformation matrices using the lumped parameter approximation to efficiently describe the link deflections due to forces at the end of arm. Lumped parameters are also easier to identify experimental, Yoshikawa [110] used a machine vision system and discrete targets to identify a lumped parameter model of a single link flexible manipulator for use in state estimation. However,

as described in [16] this method is only accurate for longer series approximations.

2.2.3.3 Raleigh-Ritz / Assumed Modes Method (AMM)

The assumed modes approach is an energy method that requires the specification of appropriate basis functions which approximate the displacement field of the flexible element through a Ritz series expansion. The number of terms in the series corresponds to the number of modes considered. As in the lumped parameter approximation, the approximation improves with increases in the series length until converging to the true displacement field for an infinite series. In contrast to the lumped parameter approximations, the mass and stiffness are not assigned uniformly but associated with each mode of vibration. Combined with the material stiffness and inertial properties, Lagrange's equations can be used to develop an equation of motion for a finite number of vibratory modes [39, 71].

This approach was modified by Asada in [4] using a virtual link coordinate system to allow up to n serially connected flexible links. Lee [58] applied an assumed modes approach to flexible manipulators with closed kinematic chains by introducing Lagrange multipliers to determine constraint forces and in [59] proposed a computationally efficient method for deriving the mass and gravity matrices by expressing the velocity vector as a function of a Jacobian matrix and generalized coordinates. Hastings and Book [43] derived a method for developing simple state space models, using the assumed modes approach, in this case, by assuming orthogonal mode functions. Book [12] used assumed mode shapes to develop 4×4 transformation matrices for representing manipulators with rotary joints. In all of these cases, as the number of links increases, the number of state variables necessary to represent the system increases. The result is increasing model complexity.

2.2.4 Frequency Domain Models

Transfer Matrix Method (TMM) [10, 15, 27, 53] is a useful extension to the continuous method, which uses 4x4 matrices, derived in [77], to describe both the joint motion and link deflection of a flexible manipulator. Directly utilizing the solution to the partial differential equations governing the bending of the flexible elements and including the rigid bodies, lumped masses, and joint control variables, the TMM provides the exact approach for

determining the natural frequencies of the manipulator. In establishing transfer matrices for each individual element of the manipulator the displacement, deflection, moment, and shear at any point along the manipulator arm can be expressed as a function of frequency in a 4×4 matrix relating the current station to another. By expressing the global transfer matrix from the base of the arm to the end of the arm and applying appropriate boundary conditions, a homogeneous matrix equation can be determined. The natural frequencies of the system consists of the frequencies for which there exists a non-trivial null space of the matrix equation [11, 15]. Krauss [53] extended this method to three dimensional flexure, and [17] developed state space models using the transfer matrix method via an iterative procedure.

2.2.5 Determining Modeshapes and Natural Frequencies

While the TMM method has been used to develop state space models suitable for control and estimation, physical intuition is lost. In practice most researchers use assumed modes approaches, but the selection of basis functions varies among researchers. A majority use continuously differentiable polynomials to represent the mode shapes of the system, but these basis functions are a poor fit for the true mode-shapes and result in inaccurate models. Some researchers have used finite element analyses to determine the natural frequencies and mode-shapes used in the AMM approach. This results in accurate models, given that a sufficiently large number of elements are used. Similarly mode-shapes computed through the TMM method were used by Book and Majette in [17] to determine assumed modes models for control. This approach results in very accurate low order approximations of the flexible system which can be easily transformed into state space form in a manner described in chapter 3.

2.3 Sensing for Reconstructing Flexible State Variables

Sensors used for feedback and determining flexible system states include vision systems [34,75,108], optical deflection sensors [33,40,74,75], strain sensors [2,35,42,61,100,108]

(by far the most popular), and accelerometers [76,79,94]. Li and Wang [63] compared combinations of the sensing elements (joint displacement, joint velocity, root strain, tip acceleration, and tip deflection) for estimating the states of a flexible manipulator with a Kalman filter. They found that joint measurement combined with another sensor significantly improved the quality of estimates from the Kalman filter over any of the measurements by themselves. The recent advancements in and resulting ubiquity of low cost MEMS accelerometers make them a natural choice for this application when combined with a joint position measurement. However, the effects of modeling error on state reconstruction using accelerometer measurements and how to improve the robustness of these estimates remain open questions in the study of flexible manipulators.

2.3.1 Sensor Placement for State Estimation of Flexible Robotic Manipulators

While the choice of sensor is often well justified in the flexible manipulator literature, very little analysis of the desired number of sensors and sensor locations for controlling flexible robotic manipulators is presented. Statements such as, tip position is measured, strain gage placed at the base of the link, and vision feedback is utilized, are made with reference to the type of sensing used and sensor location. Given the complexity of these systems, parameter changes, modeling error, and nonlinear effects can result in unobservability of one or more system modes necessary for control and lead to instability of the control system if the sensing strategy is treated as an afterthought rather than an fundamental part of the estimation strategy.

In the late 80's and early 90's large truss-like space structures were being developed. Because of the flexible nature of these large structures active vibration control approaches were sought to eliminate persistent vibrations. Once one of these structures are in place, the sensors and actuators become difficult and expensive, if not impossible to move [49]. There was, thus, an interest generated in developing systematic methods that would determine optimal actuator and sensor placements. Trial and error placement of the sensors was simply unsatisfactory.

Kammer and Yao [49] used accurate finite element analyses to determine the mode-shapes of a structure. Placements were initially assumed at candidate locations and the Fisher information matrix composed of measurement noise covariances and mode-shapes was used to eliminate placements with low "effective independence." The result was optimal sensor placement for a limited set of target modes. In one of the few experimental applications in the literature, Toncgo and Meldrum [98] applied a form of the Fisher information matrix in an optimization routine to determine sensor placements for a 63 DOF planar truss structure (one of the few experimental applications in the literature).

Meanwhile, in [48] errors in the pre-launch finite element analysis are considered to determine the detrimental effects on the effective independence sensor placement strategy. This remains one of the few works which have considered any modeling error in the placement of sensors for flexible structures. Maghami and Joshi [68] developed a nonlinear programming strategy which placed sensors and actuators to move the transmission zeros of the control system further into the left half plane and avoid pole-zero cancellation to enhance the performance of the control system. Gawronski and Lim [37] approached optimal sensor/actuator placement using a balanced flexible realization where approximations of the controllability and observability gramians are equal and diagonal and where the diagonal values are the Hankel singular values of the system. The Hankel singular values are then used to quantify the joint observability and controllability properties and serve as a metric for determining the actuator and sensor locations. Georges [38] used the maximization of the minimum eigenvalue of the observability gramian and the minimization of the maximum eigenvalue of the controllability gramian as conditions for placing a single actuator and sensor independently. Hiramoto et al. [44] combined the problems of collocated sensor/actuator placement and control design using an explicit solution of the algebraic Riccati equation. They simulated their method for the determination of two pairs of sensor and actuator placements which minimized the H_2 norm of the closed loop system for a simply supported beam.

Rao et al. [82] used a genetic algorithm to determine optimal actuator placements for a flexible truss structure by maximizing the energy dissipation of the active controllers. A

genetic algorithm approach is interesting as it allows for the generation of fitness functions with no closed form solution, a trait that will be exploited in chapter 4.

More recently, smart structures research has been the impetus for the pursuit of optimal sensor/actuator locations. Mounting PZT actuators on thin beams and plates allows collocated sensing and actuation, and many of the methods developed for placing actuators and sensors on large space structures have found their way into these applications. Liu and Rao [64] placed sensors according to a maximum signal to noise ratio including both measurement noise and spillover of uncontrolled modes. Bruant et al. used a genetic algorithm for placing sensors on a vibrating plate according to a fitness function related to the observability gramian.

In this dissertation a general approach will be developed using the singular value decomposition of the observability gramian and a genetic algorithm optimization to determine optimal (with respect to signal to noise ratio) and robust (with respect to observability) sensor placements for a finite number of modes, as discussed in chapter 4.

2.4 Robust Flexible State Variable Estimation

Practical flexible robotic manipulators are rarely encountered in the literature. Often only simulated results are presented and effects like changes in payload, nonlinear deformation, saturation, and modeling error are neglected. Even fewer examples of experimental analysis with imperfect models exist. A large reason for this gap is the lack of methods for recovering useful estimates of flexible state variables when the parameters of the system vary or are incorrectly identified. Without accurate state estimates, any control algorithm's performance would suffer. Thus, a search for a suitable state estimator which exhibits invariance to bounded parametric uncertainty and nonlinearities was executed.

While a plethora of "robust" observers exist in the literature, (e.g robust Kalman filters [62, 107, 112], neural network observers [1], and H_∞ [46] observers to name a few), sliding mode observers generated the most interest. Sliding mode control has been implemented on several occasions to flexible manipulators [8, 50, 109] with great success, but in all cases the state vector is assumed to be known. And direct measurements of the flexible state

vector, as established, are very difficult to obtain. A natural extension therefore is to apply sliding mode control theory to the observation problem. Originally developed in the late 80's by Walcott and Zak [104], sliding mode observers have been gaining momentum in recent years, especially as estimators for flexible manipulators.

Utilizing the techniques from variable structures theory (VSS) Walcott and Zak [104] developed the first sliding mode observer. It uses a discontinuous switching gain in addition to a traditional linear gain in providing an observation strategy which does not require exact knowledge of the nonlinearities or uncertainties of the system. They present a proof in which a term associated with parametric error is bounded by a scalar gain to enforce a negative definite condition on the derivative of a Lyapunov candidate. This ensures stability of the estimate error. In doing so it is asserted that the estimate error $\rightarrow 0$ as $t \rightarrow \infty$. As is argued in this dissertation, this is often not the case and, instead, the error will approach some bounded value. In [103] the rate of convergence was studied and a more general observer formulation offered, including invariance to input as well as state disturbances.

2.4.1 Applications of Sliding Mode Estimation to Flexible Robotic Manipulators

Wit and Slotine [105] applied sliding mode observers to simulations of robotic manipulators with and without joint friction. Comparisons of the sliding mode estimates with sampled and differentiated joint variable measurements were drawn with favorable results. Chalhoub and Kfoury [23] developed a sliding mode estimator of the form in [105] for a flexible link manipulator using the joint displacement variable to estimate the flexible states. The authors noted that poor estimator performance was achieved in the presence of unstrucured uncertainty, but good performance was found when the model was a good match for the plant. Elberheiry and Elmaraghy [32] derived a combination Luenberger and sliding mode estimator and applied the result to a two link flexible joint manipulator. In their work, the switching gain derived is active until the transient time interval ends and a high gain Luenberger observer takes over. The switch between the sliding mode and Luenberger gains was used in order to avoid the large overshoots produced by high gain observers under transient conditions. Zaki et al. [111] applied a variable structure observer design for the

control of a two link flexible manipulator and showed better performance than a "quasi-linear" estimator (a linear estimator applied to the nonlinear system but designed using a linearized system model). Martinez et al. [69, 70] combined a sliding mode estimator and sliding mode controller for vibration suppression control of a flexible arm, using only the motor position as feedback (in simulation) for a nominal model of the plant dynamics.

2.4.2 Other Developments in Sliding Mode Observers of Note

Furuhashi et al. [36] designed a control algorithm for brushless DC motors using an adaptive sliding mode observer to estimate the position and velocity of the rotor from the voltages induced by the rotation of the motor. An adaptive scheme was also presented to correct the velocity estimate from the effect of parameter errors. Kim and Inman [52] proposed a sliding mode observer to reduce the effect of observation spillover from residual modes in the state estimation of a vibrating cantilever beam, again using a Lyapunov analysis to demonstrate that the residual modes (treated as a disturbance) will not destabilize the estimator, whereas a Kalman filter was shown to produce divergent state estimates. Chen and Dunnigan [24] developed a sliding observer with six design rules to avoid chattering phenomena and compared the results of estimates from the sliding mode observer to estimates from both the Kalman filter and extended Kalman filter for an induction machine simulation and experiment. Results were presented that indicate that while the extended Kalman filter outperformed the sliding mode observer, the design and implementation was more complex. Su et al [95] develop a sliding observer for determining unmeasured velocity signals from measured position signals, but requiring only the degrees of freedom of the mechanical system, but no other detailed modeling information. Veluvolo et al. [102] developed a sliding mode observer for a bio-reactor and used it to estimate the unknown (scalar valued) input from an equivalent control term (a byproduct of the estimation algorithm) while in the sliding mode. Dian et al. [31] applied a sliding mode observer to the application of positioning a precision motion stage. The error compensation approach used an estimator function based on the equivalent output injection signal (i.e. the effort to maintain the motion on the sliding surface), which can be directly calculated from the sliding surface

dynamics and approximates the state and input dependent uncertainty.

2.4.3 Multiple Model Adaptive State Estimators

Given the complex problem of modeling error in flexible state variable estimation it makes intuitive sense that if the correct model could be recovered, an estimator based on the accurate model would provide better state estimates than one based on an inaccurate model. This notion is the basic premise of a multiple model adaptive state estimator (MMAE). Essentially the general objective is to operate a set of independent estimators based on different parameter variations which correspond to the likely range of true parameter values. Therefore, determining the best fit state vector estimate simply requires identification of the best model in the set for the given operating regime. Multiple model approaches to control and estimation grew out of the need to handle complex problems and computing improvements of the 1970s and 80s but fell out of favor for the most part because of high computational burden. However, because of the growing parallel computing sector, multiple model estimation is making a comeback. Each estimator can be run independently on low performance processors or embedded systems and a central agent is only used to select the best estimate.

Athans and Chang [5] provide one of the earliest analyses of the multiple model adaptive state estimation approach. In their work, they derive the evolution of the posterior probability density and weighting functions used to select state estimates from a bank of Kalman filters using the residual output error. Lashlee and Maybeck [56] applied multiple model adaptive estimation to space structure control using a moving bank which was used to select a small set of representative models is selected from a large bank of 100 models by a decision mechanism. The benefit of the moving bank approach is the reduced computational burden, but it comes at the expense of performance if the best model lies outside of the moving bank. Leahy and Tellman [57] developed an approach for multiple model-based control of rigid robotic manipulators with varying payloads. Inclusion of the Kalman filter bank MMAE allows the estimation of the full payload vector for a rigid robot. The payload vector estimate was then used to switch joint PD gains and feed-forward compensation.

Simulations were performed of a PUMA robot and results indicated significant performance improvements over single model based controllers. Menke and Maybeck [72] used a multiple model adaptive estimation approach to determine sensor/actuator failure detection in the Vista F-16 aircraft with the ability to detect hard single and multiple failures with good convergence properties. Debus et al. [29] determined a multiple model adaptive estimator to estimate contact states for rigid robot/environment contact. Applied to a peg-in-hole manipulation task, a hidden markov model was used as the decision test. Providing the kinematics of the rigid robot and a description of the contact states, object properties were estimated successfully and to a level exceeding what could be achieved by the operator. Rong and Vesselin [83] performed a survey of maneuvering target tracking including applications of multiple model approaches. They identified three separate generations of multiple model estimators. In the first generation each filter operates independently and in the second the filters cooperate to achieve better performance. The third is different in that it uses a variable size bank of estimator models. Multiple Kalman filters were used by Quinlan and Middleton in [81] to determine the state of a robo-cup robot. In this variation of the multiple model estimator, models were allowed to split and merge. The developed approach is compared with a particle filter and shown to be significantly faster in terms of execution time.

Of the developed approaches, a modified first generation method (with a fixed bank of parallel estimators) will be used in this dissertation as described in chapter 6.

2.5 Thesis Contributions

The primary contribution of this work is the creation, fundamental analysis, and experimental verification of a robust strategy for state estimation for general flexible manipulators. This strategy encompasses generation of an appropriate model of the dynamic behavior, selection and appropriate placement of sensors for flexible state estimation via a derived optimal procedure, and the establishment of a robust multiple model adaptive estimator that takes into account modeling error and parametric variation.

A significant effort was put toward the creation of an accurate low order system model

of the FLASHMan testbed. A combined TMM/AMM model approach has been reported in the literature (see [17]). My original contributions are, in part, in the application to flexible gantry systems and in the use of joint flexibility to change the control domain from force/torque inputs to displacement inputs. The latter improves flexible state estimation (i.e. using a readily measured joint variable as the input to the state observer rather than the estimated force on the base of a flexible link). Furthermore, the analysis and experimental evaluation of low-cost MEMs acceleration sensors presented herein is unique in flexible robotic manipulator applications. While accelerometers have been used in previous works, the effects of direct feed-through behavior and dynamic errors introduced through parameter bias have not been studied in this domain.

The use of the singular values of the observability gramian associated with the modes of the flexible system for maintaining robustness with respect to observability, and the novel procedure for determining multiple sensor placements for flexible link manipulators are valuable additions to the flexible manipulator control literature.

Analyses and experimental results of sliding mode estimation approaches for the control of flexible manipulators, under conditions of parametric uncertainties which corrupt the output and direct feed through matrices, are presented. Simulation and experimental results are provided comparing the performance of sliding mode estimation vs both Kalman filter and traditional Luenberger observers in flexible manipulator applications.

Finally, the development and experimental verification of a multiple model adaptive estimator for controlling flexible manipulators is presented and serves as a novel approach for the determination of robust state estimates in the flexible manipulator domain. Unlike the works presented in the literature from other domains, the weight assignment and selection algorithms developed are uncorrelated with Kalman filter estimate probability density functions. Instead, they are directly related to the parametric bias in the modal model, by utilizing multiple sensor measurements and multiple sub-estimators, regardless of the chosen estimation strategy.

Chapter III

MODELS FOR ESTIMATION AND CONTROL

The modeling of flexible manipulators has been the subject of investigation over several decades. Single and multi-link manipulator models have been presented in both the time and frequency domain. At issue in this work is the determination of a low order model which adequately characterizes the behavior of the flexible system. Modal models are a common choice among researchers for estimation and control of flexible motion systems [6] and offer an elegant transition into state space form described by (5) and (6).

$$\dot{z}(t) = Az(t) + Bu(t) \tag{5}$$

$$y(t) = Cz(t) + Du(t) \tag{6}$$

The task then becomes relating the rigid and flexible system dynamics in the state space form. A hybrid approach of both frequency and time domain modeling methods is used to form modal equations of motion that are transformed into state space form. In doing so, an accurate system model of low order is generated that is suitable for real time control and estimation.

3.1 Frequency Domain Modeling for Basis Function Generation

Complex partial differential equations are required to model complicated manipulator structures. These equations range from difficult to impossible to solve in closed form. If we set up the problem at an arbitrary pose, transfer matrices can be used to combine individual elements into a representation of the dynamics of the complete system. The roots of the resulting boundary value problem are the natural frequencies of the individual modes of vibration.

Let the TMM state variables be chosen to adequately describe the planar flexure of a

beam as follows.

$$\upsilon = \begin{bmatrix} -w \\ \psi \\ M \\ V \end{bmatrix} = \begin{bmatrix} -displacement \\ angle \\ moment \\ shear force \end{bmatrix} \tag{7}$$

Note that these states are not the same as those used to define the state space representation in (5). Furthermore, let a transfer matrix U describe the evolution of these state variables over an element of the structure, e.g. a joint, a beam, or a rigid body.

$$\upsilon_i = U_i \upsilon_{i+1} \tag{8}$$

Then, for example, a system composed of three components would have the total transfer matrix U_{tot} as described in [77].

$$\upsilon_0 = U_0 U_1 U_2 \upsilon_3 = U_{tot} \upsilon_3 \tag{9}$$

3.1.1 Controlled Joint

Controlled joints which enforce a manipulator's pose share similar behavior with stiffness and damping elements. For example, for a PD controlled joint illustrated in figure 4

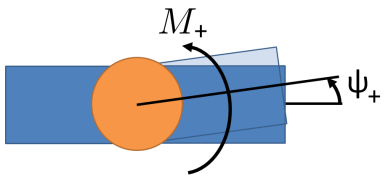


Figure 4: Controlled Joint

the torque produced in response to an error of $e = \psi_+ - \psi_-$ is

$$M_+ = -K_p e - K_d \dot{e} \tag{10}$$

.

Thus, in the frequency domain, $-\psi_- = \psi_+ + \frac{M_+}{K_p+K_d s}$ and therefore the transfer matrix associated with a controlled joint is

$$ControlledJoint(C) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1/k(s) & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11}$$

where $k(s)$ is the transfer function of the joint controller and $s = j\omega$.

3.1.2 Planar Rotation

The second important transfer matrix is for static rotation in a plane of angle θ as illustrated in figure 5.

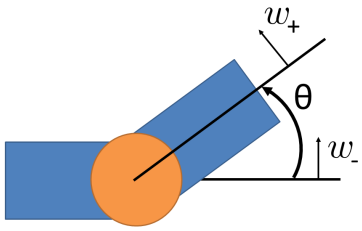


Figure 5: Planar Rotation

Reproduced from [27] the associated transfer matrix is A as described by (12) when compressive effects are assumed to be negligible.

$$PlanarRotation(A) = \begin{bmatrix} 1/\cos\theta & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ m_s\omega^2 \sin\theta \tan\theta & 0 & 0 & \cos\theta \end{bmatrix} \tag{12}$$

where $m_s = \sum_{j=i}^{n} m_j$ is the outboard mass from the joint i to the end of arm or the next planar rotation.

3.1.3 Rigid Body

The transfer matrix associated with rigid bodies, an example of which is shown in Figure 6, can be derived by: 1) summing moments about the center of mass of the rigid body; and 2) balancing forces in the normal direction (y).

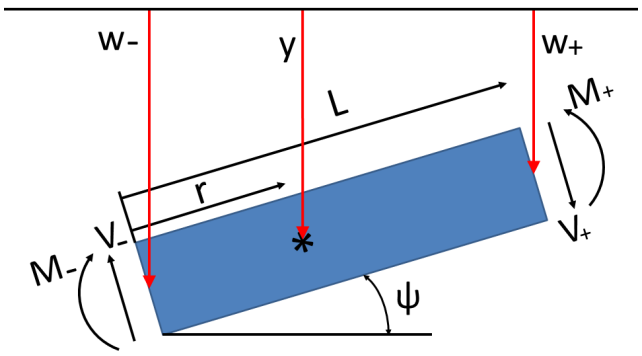


Figure 6: Rigid Body

$$\sum F_y = m\ddot{y} = V_- - V_+ \tag{13}$$

where $\ddot{y} = \ddot{w}_- + r\ddot{\psi}_-$, $\psi_+ = \psi_-$, and $w_+ = w_- + L\psi_- \Rightarrow \ddot{y} = \ddot{w}_+ - L\ddot{\psi}_+ + r\ddot{\psi}_+$. Meaning that, in the frequency domain, the shear force on the left end of the rigid body is

$$\begin{aligned} V_- &= ms^2(-w_+) + ms^2(L-r)\psi_+ + V_+ \\ &= \begin{bmatrix} ms^2 & ms^2(L-r) & 0 & 1 \end{bmatrix} v_+ \end{aligned} \tag{14}$$

Similarly, summing moments about the center of gravity results in an expression for the left moment in terms of the state vector v.

$$\sum_{\circlearrowleft +} M_g = I_g(-\ddot{\psi}_+) = V_- r + V_+(L-r) - M_+ + M_- \tag{15}$$

Inserting the definition of V_- from (14).

$$M_- = I_g s^2 \psi_+ - ms^2 r(-w_+) - ms^2 r(L-r)\psi_+ - V_+ L + M_+ \tag{16}$$

Therefore, the complete transfer matrix for a rigid body element is R as described in (17)

$$RigidBody(R) = \begin{bmatrix} 1 & L & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -ms^2(L-r) & s^2 I - ms^2 r(L-r) & 1 & -L \\ ms^2 & ms^2(L-r) & 0 & 1 \end{bmatrix} \tag{17}$$

3.1.4 Beam Element with Distributed Flexibility

Most importantly the transfer matrix for a flexible beam element can be derived from the four state equations in the spatial direction, x.

$$\psi = \frac{dw}{dx} \tag{18}$$

$$M = EI\frac{d^2 w}{dx^2} \tag{19}$$

$$V = -\frac{\partial M}{\partial x} \tag{20}$$

$$\frac{\partial V}{\partial x} = \mu \frac{\partial^2 w}{\partial t^2} \tag{21}$$

where (18) comes from the small angle approximation of $tan(\frac{d\psi}{dx})$ as displayed in figure 7, (19) comes from the mechanics of a beam in bending, and (20) and (21) come from the application of Newtons second law to a differential beam element, and where $V_+ = V_- + \frac{\partial V}{\partial x}dx$ and $M_+ = M_- + \frac{\partial M}{\partial x}dx$. Combining equations (19), (20), and (21) yeilds the partial differential equation governing the flexure of the beam in bending.

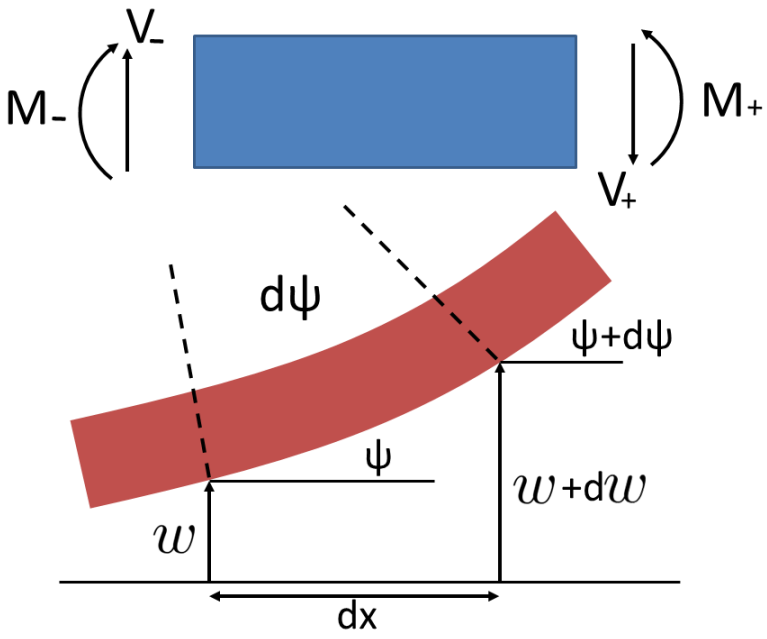


Figure 7: Beam Element

$$-EI\frac{\partial^4 w}{\partial x^4} = \mu\frac{\partial^2 w}{\partial t^2} \tag{22}$$

This equation can be solved by separating the time and space dependent terms of $w(x,t)$, i.e.

$$w(x,t) = W(x)T(t) \tag{23}$$

Therefore the resulting spatial DE is

$$W'''' - \left(\frac{\beta}{L}\right)^4 W = 0 \tag{24}$$

where

$$\left(\frac{\beta}{L}\right)^4 = -\omega^2\frac{\mu}{EI} \tag{25}$$

.

A general solution to (24) is given by

$$W = A_1 sin(\frac{\beta x}{L}) + A_2 cos(\frac{\beta x}{L}) + A_3 sinh(\frac{\beta x}{L}) + A_4 cosh(\frac{\beta x}{L}) \tag{26}$$

which can be substituted into equations (18)-(20) for w. In combination with (26) these equations can express the state vector υ in terms of a coefficient vector A such that

$$\upsilon = U(x, \beta)A \tag{27}$$

Evaluating at each of the end points $x = 0, x = L$ and solving for A

$$U(0, \beta)^{-1}\upsilon(0) = A \tag{28}$$

and therefore

$$\upsilon(L) = U(L, \beta)U(0, \beta)^{-1}\upsilon(0) \tag{29}$$

or

$$\upsilon(0) = U(0, \beta)U(L, \beta)^{-1}\upsilon(L) \tag{30}$$

where the transfer matrix $B = U(0, \beta)U(L, \beta)^{-1}$, or in closed form:

$$FlexibleBeam(B) = \begin{bmatrix} c_0 & Lc_1 & ac_2 & aLc_3 \\ \beta^4 c_3/L & c_0 & ac_1/L & ac_2 \\ \beta^4 c_2/a & \beta^4 Lc_3/a & c_0 & Lc_1 \\ \beta^4 c_1/aL & \beta^4 c_2/a & \beta^4 c_3/L & c_0 \end{bmatrix} \tag{31}$$

where,

$$a = \frac{L^2}{EI} \quad , \quad \beta^4 = \omega^2 L^4 \mu/(EI) \tag{32}$$

and

$$c = \begin{Bmatrix} (\cos\beta + \cosh\beta)/2 \\ (\sin\beta + \sinh\beta)/(2\beta) \\ (\cosh\beta - \cos\beta)/(2\beta^2) \\ (\sinh\beta - \sin\beta)/(2\beta^3) \end{Bmatrix} \tag{33}$$

for a more detailed derivation see [9, 53].

3.1.5 Transfer Matrix Modeling

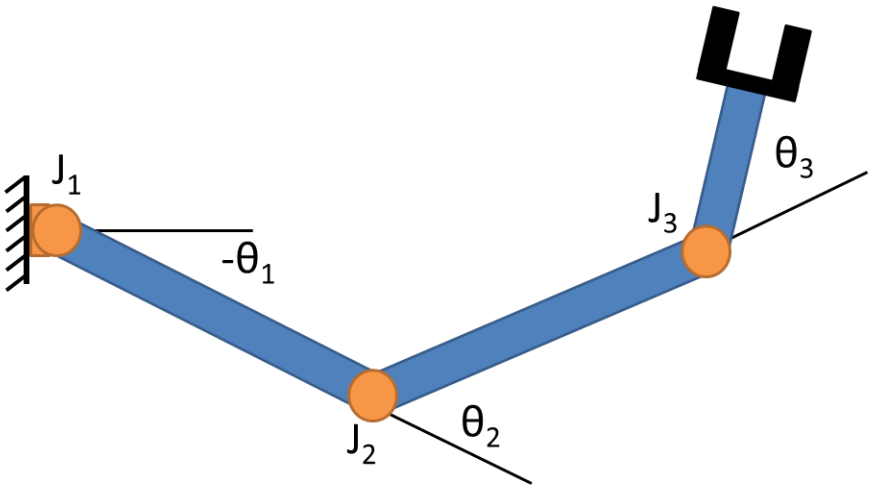


Figure 8: Three Link Flexible Manipulator

For complex robot structures, the global transfer matrix is simply the product of several individual transfer matrices. For example, the global transfer matrix for the robot in Figure 8 fixed in a particular pose with three flexible links and rigid bodies at the end of each link would simply be:

$$U_G = R_1 A_1 B_1 R_2 A_2 B_2 R_3 A_3 B_3 R_4 \tag{34}$$

Therefore $\upsilon_0 = U_G(s)\upsilon_4$ provides the relationship for the state variables at the robot base

given the conditions at the end effector. By applying the appropriate boundary conditions, a reduced order homogeneous system $U_{Gh}(s)$ can be obtained whose determinant, when evaluated for varying frequencies for zero crossings, provides the eigenvalues of the flexible system. For example in the case of the manipulator in figure 8 with clamped - free boundary conditions in a static pose,

$$\begin{bmatrix} 0 \\ 0 \\ M \\ V \end{bmatrix}_0 = \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} & u_{1,4} \\ u_{2,1} & u_{2,2} & u_{2,3} & u_{2,4} \\ u_{3,1} & u_{3,2} & u_{3,3} & u_{3,4} \\ u_{4,1} & u_{4,2} & u_{4,3} & u_{4,4} \end{bmatrix} \begin{bmatrix} w \\ \psi \\ 0 \\ 0 \end{bmatrix}_4 \tag{35}$$

from which the homogeneous reduced order system (36) can be extracted.

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}_0 = \begin{bmatrix} u_{1,1} & u_{1,2} \\ u_{2,1} & u_{2,2} \end{bmatrix} \begin{bmatrix} w \\ \psi \end{bmatrix}_4 \tag{36}$$

Therefore, values of s for which (36) has a non-trivial solution are the natural frequencies of the system. These values can be determined directly by evaluating the determinant of the subsystem as a function of s and searching for the zero crossings.

As a direct solution to the PDE governing the elastic deformation of the flexible structure, this method is a very accurate approach to determining the natural frequencies. As will be discussed in Section 3.3, it also is useful in determining the mode shapes of a flexible manipulator. It is not, however, a trivial matter to extract a meaningful state space model from the frequency domain representation [17].

3.2 Assumed Modes Modelling of Flexible Manipulators

The assumed modes modeling (AMM) approach for flexible structures is a Lagrangian approach for expressing the equations of motion, in the time domain, of systems with distributed mass and elasticity. Assumptions are made about the mode shapes of the system which are in turn used to formulate the structural dynamics. Kinetic energy, potential energy, and energy dissipation equations are formulated using these assumed mode shapes as

basis functions. The equation of motion can then the be determined through the application of the non-conservative Lagranges equations

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial \mathscr{D}}{\partial \dot{q}_i} + \frac{\partial V}{\partial q_i} = Q \tag{37}$$

which, for a linear system, results in the equations of motion (38).

$$\Rightarrow \quad M\ddot{q} + C\dot{q} + Kq = Q \tag{38}$$

3.2.1 AMM Example - Generic Single Link Manipulator

To best illustrate the AMM procedure for general flexible manipulator systems, consider the planar single link flexible manipulator in figure 9.

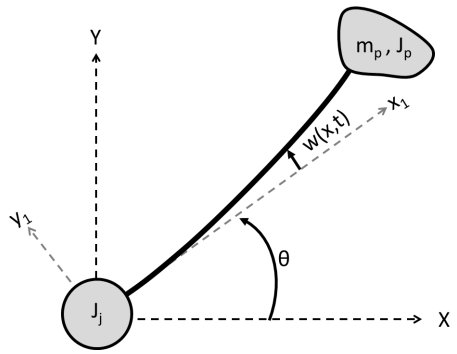


Figure 9: Flexible Link Manipulator

The kinetic energy of the system must first be expressed before determining the equations of motion.

$$T = \frac{1}{2}\int_m \dot{R}_P \cdot \dot{R}_P \, dm \tag{39}$$

where $\dot{R}_P$ is the velocity of a mass element dm at point P expressed in the global coordinate system (X, Y). From figure 9

$$R_P = (x_1 C\theta - w\,(x_1, t)\, S\theta)\hat{i} + (x_1 S\theta + w\,(x_1, t)\, C\theta)\hat{j} \tag{40}$$

where $w\,(x_1, t)$ is the deflection of the link in the y_1 direction and $\hat{i}$ and $\hat{j}$ are unit vectors

in the global X and Y directions. The deflection $w(x_1, t)$ can be separated into spatial and time dependent variables according to

$$w(x,t) = \sum_{i=1}^{\infty} \psi_i(x_1) q_{f_i}(t) \tag{41}$$

where $q_{f_i}(t)$ is the generalized coordinate associated with the i^{th} mode of vibration and $\psi_i(x_1)$ is the basis function approximating the ith mode shape of the link. Taking the derivative with respect to time of (41) gives the velocity of point P.

$$\begin{aligned} \dot{R}_P = &\left[-S\theta \sum_{i=1}^{N} \psi_i(x_1) \dot{q_{f_i}} - \dot{\theta} \left(x_1 S\theta + C\theta \sum_{i=1}^{N} q_{f_i} \psi_i(x_1) \right) \right] \hat{X} \\ &+ \left[C\theta \sum_{i=1}^{N} \psi_i(x_1) \dot{q_{f_i}} + \dot{\theta} \left(x_1 C\theta - S\theta \sum_{i=1}^{N} q_{f_i} \psi_i(x_1) \right) \right] \hat{Y} \end{aligned} \tag{42}$$

and therefore

$$\dot{R}_P \cdot \dot{R}_P = \dot{\theta}^2 \left(\sum_{i=1}^{N} \sum_{j=1}^{N} q_{f_i} q_{f_j} \psi_i(x_1) \psi_j(x_1) + x_1^2 \right) + 2x_1 \dot{\theta} \sum_{i=1}^{N} \dot{q_{f_i}} \psi_i(x_1) + \sum_{i=1}^{N} \sum_{j=1}^{N} \dot{q_{f_i}} \dot{q_{f_j}} \psi_i(x_1) \psi_j(x_1) \tag{43}$$

The kinetic energy can be divided into separate integrals over each mass element. With simplification this result is

$$T = \frac{1}{2} J_j \dot{\theta}^2 + \frac{1}{2} \int_{m_1} \dot{R}_P \cdot \dot{R}_P dm + \frac{1}{2} m_{eff} \dot{R_{eff}} \cdot \dot{R_{eff}} + \frac{1}{2} J_{eff} \left(\dot{\theta} + \left. \frac{\partial \dot{w}(x_1, t)}{\partial x_1} \right|_{x_1 = L} \right) \tag{44}$$

After determining the first two terms of (37), and sparing the tedious manipulation of the resulting equations, the mass matrix of the equation of motion is determined. The total system inertia term is,

$$J_{Total} = \left[J_j + \frac{1}{3} \rho A L^3 + m_{eff} L^2 + J_{eff} \right] \tag{45}$$

the terms coupling the rigid body coodinate θ to the flexible coordinates q_{f_i} are

$$W_i = \rho AL \int_0^L \psi_i(x_1) x_1 \, dx_1 + m_{eff} L \psi_i(L) + J_{eff} \frac{\partial \psi_i(L)}{\partial x_1} \tag{46}$$

and the terms coupling the modes of vibration to each other are

$$H_{i,j} = \rho AL \sum_{i=1}^{N} \int_0^L \psi_i(x_1)\psi_j(x_1) \, dx_1 + M_{eff}\psi_i(L)\psi_i(L) + J_{eff} \sum_{i=1}^{N} \frac{\partial \psi_i(L)}{\partial x_1} \frac{\partial \psi_j(L)}{\partial x_1} \tag{47}$$

Therefore in terms of the complete generalized coordinate vector $q = \begin{bmatrix} \theta & q_{f_1} & \cdots & q_{f_1} \end{bmatrix}^T$ the complete mass matrix for the system is found to be (48).

$$M = \left[\begin{array}{c|ccccc} J_{total} & W_1 & W_2 & W_3 & \cdots & W_n \\ \hline W_1 & H_{1,1} & H_{1,2} & H_{1,3} & \cdots & H_{1,n} \\ W_2 & H_{2,1} & H_{2,2} & H_{2,3} & \cdots & H_{2,n} \\ W_3 & H_{3,1} & H_{3,2} & H_{3,3} & \cdots & H_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ W_n & H_{n,1} & H_{n,2} & H_{n,3} & \cdots & H_{n,n} \end{array}\right] \tag{48}$$

The stiffness matrix K is found in a similar manner by expressing the potential energy of the complete system, which originates solely from the strain energy of the beam deflection.

$$\begin{aligned} V &= \frac{1}{2} \int_{0^+}^{L} EI \left[\frac{\partial^2 w(x_1, t)}{\partial x_1^2} \right]^2 dx_1 \\ &= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} q_{f_i} q_{f_j} \, EI \int_{0^+}^{L} \frac{d^2 \psi_i(x_1)}{dx_1^2} \frac{d^2 \psi_j(x)}{dx_1^2} dx_1 \end{aligned} \tag{49}$$

Evaluating the fourth term of (37) with respect to the generalized coordinates yields

$$K = \left[\begin{array}{c|ccccc} 0 & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & k_{1,1} & k_{1,2} & k_{1,3} & \cdots & k_{1,n} \\ 0 & k_{2,1} & k_{2,2} & k_{2,3} & \cdots & k_{2,n} \\ 0 & k_{3,1} & k_{3,2} & k_{3,3} & \cdots & k_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & k_{n,1} & k_{n,2} & k_{n,3} & \cdots & k_{n,n} \end{array}\right] \tag{50}$$

where

$$k_{i,j} = EI \int_{0^+}^{L} \frac{d^2\psi_i(x_1)}{dx_1^2} \frac{d^2\psi_j(x_1)}{dx_1^2} dx_1 \tag{51}$$

Note that the rank deficiency of K indicates that the eigenvalue problem $(K - \omega^2 M)v = 0$ would result in a 0 eigenvalue, indicating a rigid body mode is present in the system dynamics.

For non-conservative systems a dissipation term must also be considered. Rayleigh's dissipation function $\mathscr{D}$ is used in this instance and is equal to $\frac{1}{2}\mathscr{P}_{dis}$

$$\mathscr{D} = \frac{1}{2} c_j \dot{\theta}^2 + \frac{1}{2}\mathscr{P}_{dis} = \frac{1}{2}\gamma EI \sum_{i=1}^{n}\sum_{j=1}^{n} \dot{q}_i \dot{q}_j \int_{0^+}^{L} \frac{d^2\psi_i(x_1)}{dx_1^2} \frac{d^2\psi_j(x_1)}{dx_1^2} dx_1 \tag{52}$$

Evaluating the remaining term of (37) on the left hand side results in the damping matrix C.

$$C = \left[\begin{array}{c|ccccc} c_j & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & c_{1,1} & c_{1,2} & c_{1,3} & \cdots & c_{1,n} \\ 0 & c_{2,1} & c_{2,2} & c_{2,3} & \cdots & c_{2,n} \\ 0 & c_{3,1} & c_{3,2} & c_{3,3} & \cdots & c_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & c_{n,1} & c_{n,2} & c_{n,3} & \cdots & c_{n,n} \end{array}\right] \tag{53}$$

where

$$c_{i,j} = \gamma EI \int_{0^+}^{L} \frac{d^2\psi_k(x_1)}{dx_1^2} \frac{d^2\psi_i(x_1)}{dx_1^2} dx_1 \tag{54}$$

The generalized forcing term indicates how control and disturbance energy enters the system. In this case it is assumed that the input is a torque Γ at the joint resulting in the generalized force vector

$$Q = \begin{bmatrix} \Gamma \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{55}$$

Extendable to multiple link manipulators, this approach is a valid method for determining the complete nonlinear equations of motion. For the remainder of this work, it is assumed that these equations are linearized about each operating point of interest and linear analysis methods applied or manipulated with nonlinear tools such as the extended Kalman filter in order to extend the developed estimation approach beyond single link manipulators.

3.3 Selecting Basis Functions for Accurate AMM Models

The accuracy of the assumed modes modeling approach is dependent on the choice of basis functions used to represent the spacial component of flexure. The only requirements for selecting appropriate basis functions is that they match the boundary conditions of the flexible element to which they are applied and are continuously differentiable. For example, a clamped-free beam has geometric boundary conditions on the clamped end

$$w(x_1 = 0) = 0 \tag{56}$$

$$\psi(x_1 = 0) = 0 \tag{57}$$

$$\tag{58}$$

which mean that, as suggested in [39]

$$\psi_i(x_1) = \left(\frac{x_1}{L}\right)^{j+1} \tag{59}$$

is a valid basis function. If, however, this choice is made, the result from [39] describing the convergence of the approximate natural frequencies to the true frequencies becomes important, (i.e. the approximate natural frequencies predicted by the AMM modeling process approach the true natural frequencies from above as the series length (number of modes considered in the Ritz approximation) $\rightarrow \infty$).

This is referred to as the upper bound theorem which predicts that model accuracy is higher with higher numbers of modes. Higher order models are thus required to obtain accurate approximations of important low frequency system dynamics. Nevertheless, the choice of basis function ultimately plays a large role in the overall error. In using better approximations of the mode shape for the basis functions in the initial estimates of the systems natural frequencies, the approximations are better for low order models

The best approximation of the mode shape is intuitively the true mode shape which can only be evaluated experimentally, but is best approximated through the transfer matrix method analysis of section 3.1. After determining the natural frequencies of the system, the base state vector υ_0 for each mode can be determined by solving for the non zero elements $\hat{\upsilon}_{ep}$ of the end point boundary conditions υ_{ep} using

$$0 = subU_G(s = j\omega_i)\hat{\upsilon}_{ep} \tag{60}$$

where $subU_G$ is the sub-matrix described in (36) and then evaluating

$$\upsilon_0 = U_G(s = j\omega_i)\upsilon_{ep} \tag{61}$$

.

The mode shapes are simply $w(x_1)$, or the first element of υ evaluated over the length of the structure by computing intermediary transfer matrices from the base vector to a position x along the arm. For example given the single link manipulator in Figure 9, the state vector at a position x_1 along the neutral axis of the beam is simply

$$\upsilon(x_1) = \left(C_1 R_1 A_1 B_1(x_1)\right)^{-1} \upsilon_0 \tag{62}$$

where $B_1(x_1)$ is the beam transfer matrix where x_1 is substituted for L. Evaluating this in an iterative fashion for successive values of x_1 results in the basis functions $\psi_i(x_1) = w_i(x_1)$ for $i = 1 \ldots n$ where n is the desired number of modes.

While the hybrid TMM/AMM approach is used exclusively in this work, alternative methods for establishing accurate mode shapes like finite element analysis are equally valid.

3.4 FLASHMan Introduction

The primary experimental testbed for this work is known as FLASHMan: Flexible, Lightweight, And Stable Manipulator. Pictured in Figure 10 FLASHMan is a re-purposed CAMotion Inc. gantry style packaging robot. Actuated by belt drives, the end effector is located at the end of a flexible beam and is capable of traversing the $x - y$ plane. The stiffness in the x-direction (left to right in the diagram) is significantly lower than in the y-direction. As such, the primary focus of the analysis presented will be with regard to motion in the x-direction.

Controlled motion of the x-axis is achieved through a timing belt pulley which moves the cart and the affixed aluminum flexible link and payload. Feedback signals consist of an encoder measurement of the pulley position and acceleration measurements from re-positionable MEMs accelerometers along the length of the flexible link. A vision system is used throughout to assess the effectiveness of the developed approach by providing a ground truth measurement of payload position.

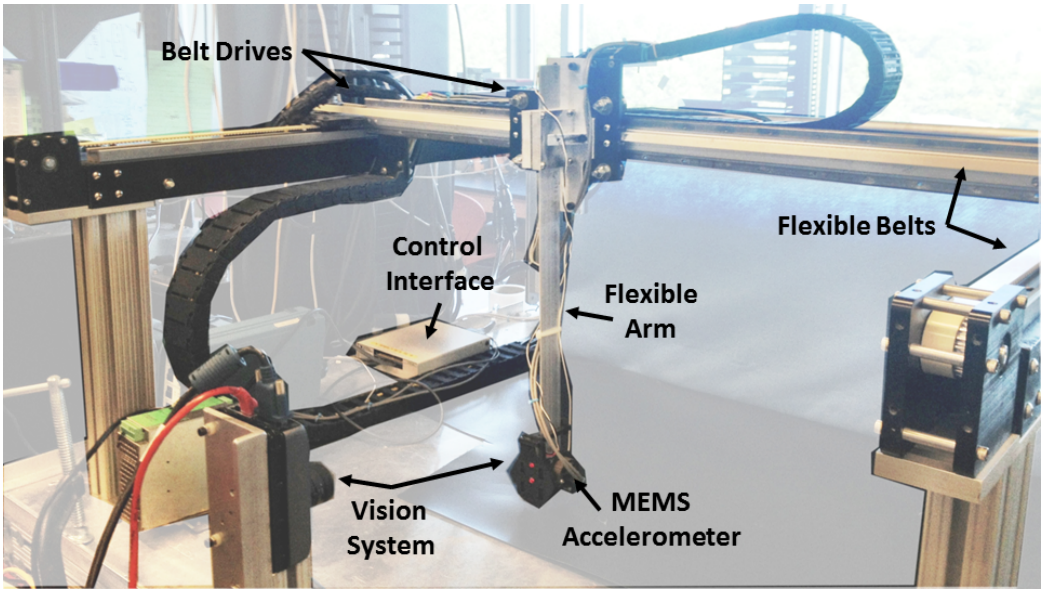


Figure 10: FLASHMan (Flexible, Lightweight, And Stable Manipulator) Testbed

3.5 Hybrid TMM-AMM Model Derivation - FLASHMan

Following the hybrid TMM-AMM modeling approach detailed in the preceeding sections, a state space model has been developed as follows.

A simplified schematic of the system is presented in Figure 11, isolating motion in the x direction. Constructing the system model requires first calculating the kinetic energy of the complete system. The generalized position vector of a mass element of the system is

$$R_{dm} = (x_c + w(y,t))\hat{i} + y\hat{j} \tag{63}$$

where x_c is the displacement of the cart, $w(y,t)$ is the displacement of a mass element at position y along the flexible beam. Separating the spatial and time dependent components of $w(y,t)$ per (41).

$$R_{dm} = (x_c + \sum_{i=1}^{n} q_{f_i}\psi_i(y))\hat{i} + y\hat{j} \tag{64}$$

Therefore, the dot product of the velocity of the beam element is

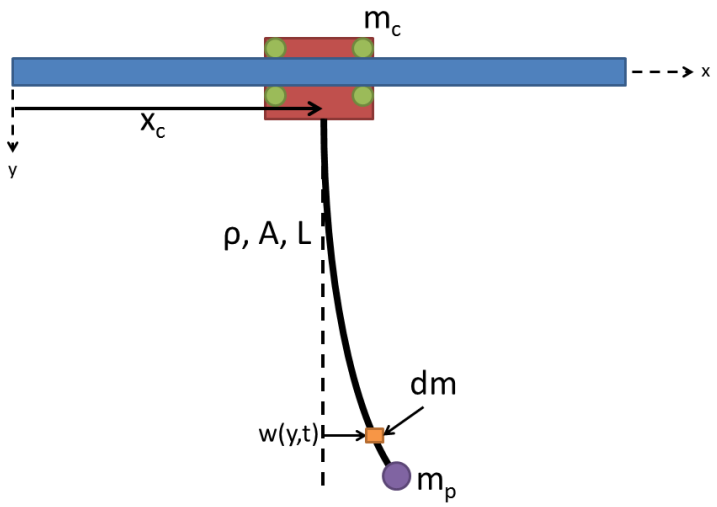


Figure 11: System Diagram

$$\dot{R}_{dm} \cdot \dot{R}_{dm} = \dot{x_c}^2 + 2\dot{x}_c \sum_{i=1}^{n} \dot{q}_{f_i} \psi_i(y) + \sum_{i=1}^{n} \sum_{j=1}^{n} \dot{q}_{f_i} \dot{q}_{f_j} \psi_i(y) \psi_j(y) \tag{65}$$

The kinetic energy of the system can be divided over the three individual mass elements: the cart, the beam, and the payload.

$$T = \frac{1}{2} m_c \int_0^{0^+} (\dot{R}_{dm} \cdot \dot{R}_{dm}) \delta(y)\, dy + \frac{1}{2} \rho A \int_{0^+}^{L} (\dot{R}_{dm} \cdot \dot{R}_{dm})\, dy + \frac{1}{2} m_p \int_L^{L^+} (\dot{R}_{dm} \cdot \dot{R}_{dm}) \delta(y)\, dy \tag{66}$$

Therefore, for clarity, the kinetic energy of each element will be expressed separately.

$$T = T_{cart} + T_{beam} + T_{payload} \tag{67}$$

$$T_{cart} = \frac{1}{2} m_c \int_0^{0^+} \left[\dot{x_c}^2 + 2\dot{x}_c \sum_{i=1}^{n} \dot{q}_{f_i} \psi_i(y) + \sum_{i=1}^{n} \sum_{j=1}^{n} \dot{q}_{f_i} \dot{q}_{f_j} \psi_{f_i}(y) \psi_{f_j}(y) \right] \delta(y)\, dy \tag{68}$$

$$= \frac{1}{2} m_c \dot{x}_c^2 \tag{69}$$

$$T_{beam} = \frac{1}{2}\rho A \int_{0^+}^{L} \left[\dot{x_c}^2 + 2\dot{x}_c \sum_{i=1}^{n} \dot{q}_i \psi_i(y) + \sum_{i=1}^{n}\sum_{j=1}^{n} \dot{q}_{f_i}\dot{q}_{f_j}\psi_i(y)\psi_j(y) \right] dy \tag{70}$$

$$= \frac{1}{2}\rho A L \dot{x}_c^2 + \rho A \dot{x}_c \sum_{i=1}^{n} \dot{q}_{f_i} \int_{0^+}^{L} \psi_i(y)\ dy + \frac{1}{2}\rho A \sum_{i=1}^{n}\sum_{j=1}^{n} \dot{q}_{f_i}\dot{q}_{f_j} \int_{0^+}^{L} \psi_i(y)\psi_j(y)\ dy \tag{71}$$

$$T_{payload} = \frac{1}{2} m_p \int_{L}^{L^+} \left[\dot{x_c}^2 + 2\dot{x}_c \sum_{i=1}^{n} \dot{q}_{f_i}\psi_i(y) + \sum_{i=1}^{n}\sum_{j=1}^{n} \dot{q}_{f_i}\dot{q}_{f_j}\psi_i(y)\psi_j(y) \right] \delta(y)\, dy \tag{72}$$

$$= \frac{1}{2} m_p \dot{x}_c^2 + m_p \dot{x}_c \sum_{i=1}^{n} \dot{q}_{f_i}\psi_i(L) + \frac{1}{2} m_p \sum_{i=1}^{n}\sum_{j=1}^{n} \dot{q}_{f_i}\dot{q}_{f_j}\psi_i(L)\psi_j(L) \tag{73}$$

To determine the mass matrix the first two elements of Lagrange's equations are employed with respect to the joint and flexible generalized coordinates.

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{x}_c}\right) = \frac{d}{dt}\left(\frac{\partial T_{cart}}{\partial \dot{x}_c}\right) + \frac{d}{dt}\left(\frac{\partial T_{beam}}{\partial \dot{x}_c}\right) + \frac{d}{dt}\left(\frac{\partial T_{payload}}{\partial \dot{x}_c}\right) \tag{74}$$

(Joint Variables)

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_{f_k}}\right) = \frac{d}{dt}\left(\frac{\partial T_{cart}}{\partial \dot{q}_{f_k}}\right) + \frac{d}{dt}\left(\frac{\partial T_{beam}}{\partial \dot{q}_{f_k}}\right) + \frac{d}{dt}\left(\frac{\partial T_{payload}}{\partial \dot{x}_c}\right) \tag{75}$$

(Flexible Variables)

Carried out one element at a time (w.r.t. Joint Variables):

$$\frac{\partial T_{cart}}{\partial \dot{x}_c} = m_c \dot{x}_c \tag{76}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial T_{cart}}{\partial \dot{x}_c}\right) = m_c \ddot{x}_c \tag{77}$$

$$\frac{\partial T_{beam}}{\partial \dot{x}_c} = \rho A L \dot{x}_c + \rho A \sum_{i=1}^{n} \dot{q}_{f_i} \int_{0^+}^{L} \psi_i(y)\ dy \tag{78}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial T_{beam}}{\partial \dot{x}_c}\right) = \rho A L \ddot{x}_c + \rho A \sum_{i=1}^{n} \ddot{q}_{f_i} \int_{0^+}^{L} \psi_i(y)\ dy \tag{79}$$

$$\frac{\partial T_{payload}}{\partial \dot{x}_c} = m_p \dot{x}_c + m_p \sum_{i=1}^{n} \dot{q}_{f_i} \psi_i (L) \tag{80}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial T_{payload}}{\partial \dot{x}_c}\right) = m_p \ddot{x}_c + m_p \sum_{i=1}^{n} \ddot{q}_{f_i} \psi_i (L) \tag{81}$$

Therefore:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{x}_c}\right) = (m_c + \rho AL + m_p)\, \ddot{x}_c + \sum_{i=1}^{n} \ddot{q}_{f_i} \left[\rho A \int_{0^+}^{L} \psi_i (y)\ dy + m_p \psi_i (L)\right] \tag{82}$$

Again one element at a time (w.r.t. Flexible Variables):

$$\frac{\partial T_{cart}}{\partial \dot{q}_{f_k}} = 0 \tag{83}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial T_{cart}}{\partial \dot{q}_{f_k}}\right) = 0 \tag{84}$$

$$\frac{\partial T_{beam}}{\partial \dot{q}_{f_k}} = \rho A \dot{x}_c \int_{0^+}^{L} \psi_k (y)\ dy + \rho A \sum_{i=1}^{n} \dot{q}_{f_i} \int_{0^+}^{L} \psi_k (y)\, \psi_i (y)\ dy \tag{85}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial T_{beam}}{\partial \dot{q}_{f_k}}\right) = \rho A \ddot{x}_c \int_{0^+}^{L} \psi_k (y)\ dy + \rho A \sum_{i=1}^{n} \ddot{q}_{f_i} \int_{0^+}^{L} \psi_k (y)\, \psi_i (y)\ dy \tag{86}$$

$$\frac{\partial T_{payload}}{\partial \dot{q}_{f_k}} = m_p \dot{x}_c \psi_k (L) + m_p \sum_{i=1}^{n} \dot{q}_{f_i} \psi_k (L)\, \psi_i (L) \tag{87}$$

$$\Rightarrow \frac{d}{dt}\left(\frac{\partial T_{payload}}{\partial \dot{q}_{f_k}}\right) = m_p \ddot{x}_c \psi_k (L) + m_p \sum_{i=1}^{n} \ddot{q}_{f_i} \psi_k (L)\, \psi_i (L) \tag{88}$$

Therefore,

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_{f_k}}\right) = \left(m_p \psi_k(L) + \rho A \int_{0^+}^{L} \psi_k(y)\ dy\right)\ddot{x}_c + \sum_{i=1}^{n} \ddot{q}_{f_i}\left[\rho A \int_{0^+}^{L} \psi_k(y)\,\psi_i(y)\ dy + m_p \psi_k(L)\,\psi_i(L)\right] \tag{89}$$

and the resulting generalized mass matrix is (90).

$$M = \left[\begin{array}{c|ccccc} m_c + \rho AL + m_p & W_1 & W_2 & W_3 & \cdots & W_n \\ \hline W_1 & H_{1,1} & H_{1,2} & H_{1,3} & \cdots & H_{1,n} \\ W_2 & H_{2,1} & H_{2,2} & H_{2,3} & \cdots & H_{2,n} \\ W_3 & H_{3,1} & H_{3,2} & H_{3,3} & \cdots & H_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ W_n & H_{n,1} & H_{n,2} & H_{n,3} & \cdots & H_{n,n} \end{array}\right] \tag{90}$$

where,

$$W_i = \rho A \int_{0^+}^{L} \psi_i(y)\, dy + m_p \psi_i(L) \tag{91}$$

and

$$H_{i,j} = \rho A \int_{0^+}^{L} \psi_i(y)\,\psi_j(y)\, dy + m_p \psi_i(L)\,\psi_j(L) \tag{92}$$

Similarly, to derive the stiffness matrix K the potential energy is expressed using the strain energy of the link in flexure.

$$V = \frac{1}{2}\int_{0^+}^{L} EI\left[\frac{\partial w(y,t)}{\partial y}\right]^2 dy \tag{93}$$

$$= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} q_{f_i} q_{f_j}\ EI \int_{0^+}^{L} \frac{d^2\psi_i(y)}{dy^2}\frac{d^2\psi_j(y)}{dy^2} dy \tag{94}$$

Therefore:

$$\frac{\partial V}{\partial x_c} = 0 \tag{95}$$

$$-\frac{\partial V}{\partial q_k} = \sum_{i=1}^{n} q_{f_i}\, EI \int_{0^+}^{L} \frac{d^2\psi_k(y)}{dy^2}\frac{d^2\psi_i(y)}{dy^2}dy \tag{96}$$

Let,

$$k_{i,j} = EI \int_{0^+}^{L} \frac{d^2\psi_i(y)}{dy^2}\frac{d^2\psi_j(y)}{dy^2}dy \tag{97}$$

then the generalized stiffness matrix is:

$$K = \left[\begin{array}{c|ccccc} 0 & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & k_{1,1} & k_{1,2} & k_{1,3} & \cdots & k_{1,n} \\ 0 & k_{2,1} & k_{2,2} & k_{2,3} & \cdots & k_{2,n} \\ 0 & k_{3,1} & k_{3,2} & k_{3,3} & \cdots & k_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & k_{n,1} & k_{n,2} & k_{n,3} & \cdots & k_{n,n} \end{array}\right] \tag{98}$$

Dissipation in the system comes in the form of structural damping, which is measured via a loss factor γ, and viscous damping between the cart and track (c_{belt}).

$$\mathscr{D} = \frac{1}{2}\mathscr{P}_{dis} = \frac{1}{2}\gamma EI \sum_{i=1}^{n}\sum_{j=1}^{n} \dot{q}_{f_i}\dot{q}_{f_j} \int_{0^+}^{L} \frac{d^2\psi_i(y)}{dy^2}\frac{d^2\psi_j(y)}{dy^2}dy \tag{99}$$

Evaluating the dissipation term from Lagrange's equations,

$$\frac{\partial \mathscr{D}}{\partial \dot{x}_c} = c_{belt}\dot{x}_c \tag{100}$$

$$\frac{\partial \mathscr{D}}{\partial \dot{q}_k} = \gamma EI \sum_{i=1}^{n} \dot{q}_{f_i} \int_{0^+}^{L} \frac{d^2\psi_k(y)}{dy^2}\frac{d^2\psi_i(y)}{dy^2}dy \tag{101}$$

Let

$$c_{i,j} = \gamma EI \int_{0^+}^{L} \frac{d^2\psi_k(y)}{dy^2}\frac{d^2\psi_i(y)}{dy^2}dy \tag{102}$$

then the damping matrix C is:

$$C = \left[\begin{array}{c|cccccc} c_{belt} & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & c_{1,1} & c_{1,2} & c_{1,3} & \cdots & c_{1,n} \\ 0 & c_{2,1} & c_{2,2} & c_{2,3} & \cdots & c_{2,n} \\ 0 & c_{3,1} & c_{3,2} & c_{3,3} & \cdots & c_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & c_{n,1} & c_{n,2} & c_{n,3} & \cdots & c_{n,n} \end{array}\right] \tag{103}$$

Forcing functions enter through a direct excitation of the rigid system. These functions influence the flexible variables by the corresponding basis function, which is evaluated at the point of application, i.e.

$$Q = \begin{bmatrix} 1 \\ \psi_1(0) \\ \psi_2(0) \\ \psi_3(0) \\ \vdots \\ \psi_n(0) \end{bmatrix} F \tag{104}$$

3.5.1 Closing the Loop Around the Pulley Position

Without a very accurate model of the AC servo drive, motor, gear reduction, and power electronics that actuate the x-axis, the input force on the cart is known with relatively little accuracy. Conversely, the pulley position is known with very high accuracy, because of the attached encoder. It is therefore desirable to use the motion of the pulley as the input to the system rather than the force on the cart.

A tension is applied to the belt as the pulley rotates. This pulls the cart via a set of steel cables embedded in the rubber cover. This effect can be approximated as a base excitation where a displacement of the pulley results in a motion of the cart via an approximate spring constant. The true spring constant is nonlinear as it is dependent on the free length of the

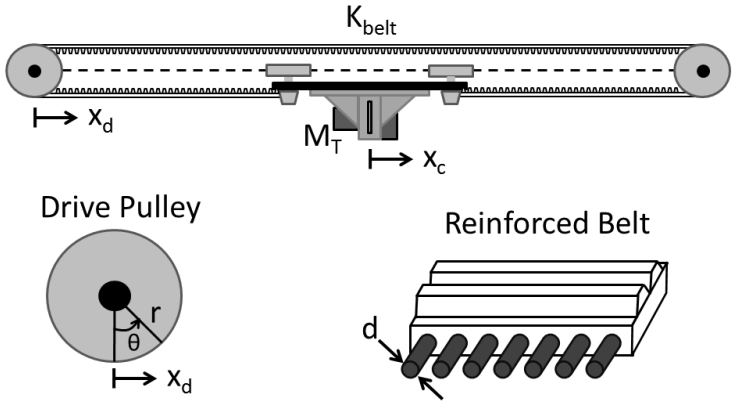


Figure 12: Belt Drive Diagram

belt, which is in-turn dependent on both the cart position and direction of travel.

Let the diameter of a cable be d, the equilibrium length be L and Young's Modulus be E, then the spring constant of a single cable is

$$K = \frac{EA}{L} = \frac{E\pi d^2}{4L} \tag{105}$$

.

All of the cables are stretched in parallel. Thus the effective spring constant is

$$K_{belt} = K_1 + K_2 + \ldots + K_n = n\frac{E\pi d^2}{4L} \tag{106}$$

where n is the number of cables. The kinetic energy stored in this elastic element is described by

$$V_{belt} = \frac{1}{2}K_{belt}\left(x_c - x_d\right)^2 = \frac{1}{2}K_{belt}x_c^2 - K_{belt}x_c x_d + \frac{1}{2}K_{belt}x_d^2 \tag{107}$$

.

Therefore Lagrange's equations have an additional term from the belt stiffness which is decoupled from the distributed stiffness of the elastic beam. Differentiating the potential energy of the belt system with respect to the previously defined generalized coordinate x_c yields:

$$\frac{\partial V_{belt}}{\partial x_c} = K_{belt}x_c - K_{belt}x_d \tag{108}$$

Augmenting the AMM model to include the belt stiffness and belt displacement (x_d) input results in the following mass, stiffness, damping, and forcing matrices:

$$M = \left[\begin{array}{c|ccccc} m_c + \rho AL + m_p & W_1 & W_2 & W_3 & \cdots & W_n \\ \hline W_1 & H_{1,1} & H_{1,2} & H_{1,3} & \cdots & H_{1,n} \\ W_2 & H_{2,1} & H_{2,2} & H_{2,3} & \cdots & H_{2,n} \\ W_3 & H_{3,1} & H_{3,2} & H_{3,3} & \cdots & H_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ W_n & H_{n,1} & H_{n,2} & H_{n,3} & \cdots & H_{n,n} \end{array}\right] \tag{109}$$

$$K = \left[\begin{array}{c|ccccc} K_{belt} & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & k_{1,1} & k_{1,2} & k_{1,3} & \cdots & k_{1,n} \\ 0 & k_{2,1} & k_{2,2} & k_{2,3} & \cdots & k_{2,n} \\ 0 & k_{3,1} & k_{3,2} & k_{3,3} & \cdots & k_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & k_{n,1} & k_{n,2} & k_{n,3} & \cdots & k_{n,n} \end{array}\right] \tag{110}$$

$$C = \left[\begin{array}{c|ccccc} C_{belt} & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & c_{1,1} & c_{1,2} & c_{1,3} & \cdots & c_{1,n} \\ 0 & c_{2,1} & c_{2,2} & c_{2,3} & \cdots & c_{2,n} \\ 0 & c_{3,1} & c_{3,2} & c_{3,3} & \cdots & c_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & c_{n,1} & c_{n,2} & c_{n,3} & \cdots & c_{n,n} \end{array}\right] \tag{111}$$

$$Q = \left[\begin{array}{c} K_{belt} \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{array}\right] x_d \tag{112}$$

where W, H, k, and c are as previously defined. The complete equations of motion are the end product.

$$M\begin{bmatrix} \ddot{x}_c \\ \ddot{q}_{f_1} \\ \vdots \\ \ddot{q}_{f_n} \end{bmatrix} + C\begin{bmatrix} \dot{x}_c \\ \dot{q}_{f_1} \\ \vdots \\ \dot{q}_{f_n} \end{bmatrix} + K\begin{bmatrix} x_c \\ q_{f_1} \\ \vdots \\ q_{f_n} \end{bmatrix} = \begin{bmatrix} K_{belt} \\ 0 \\ \vdots \\ 0 \end{bmatrix} x_d \tag{113}$$

Note that if the belt is very stiff, the cart displacement x_c will be essentially equivalent to the desired cart displacement x_d.

3.6 AMM Model Verification

A nominal model of the FLASHMan testbed was constructed using the identified system parameters listed in Table 1. Most parameters were measured experimentally with a few, including damping, loss factor, cart mass, and belt stiffness, approximated by inspection and analysis.

Table 1: Nominal FLASHMan Parameters

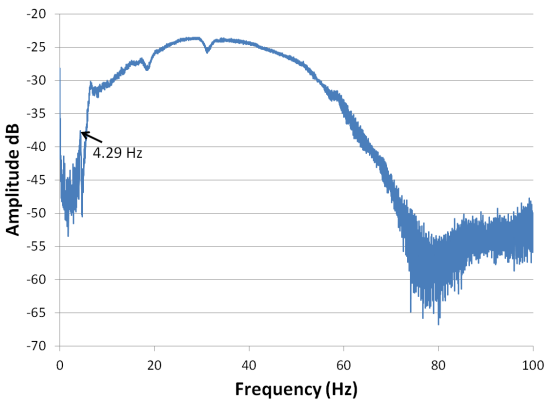
Parameter	Value	Unit
Payload (m_t)	0.281	kg
Cart Mass (m_c)	10	kg
Arm Length (L)	0.42	m
Elastic Modulus (E)	$7E^{10}$	N/m
Density (ρ)	2700	kg/m^3
Area Moment (I)	$1.0114E^{-10}$	m^4
Belt Stiffness (K_b)	$2.1814E^5$	N/m
Cart Damper (C_d)	100	Ns/m
Structural Damping Coefficient (γ)	0.0025	ND

Comparison of natural frequency predictions of the AMM modeling approach using the aforementioned techniques for specifying basis functions, demonstrate the improvement realized through utilization of TMM produced mode shapes over polynomial approximations.

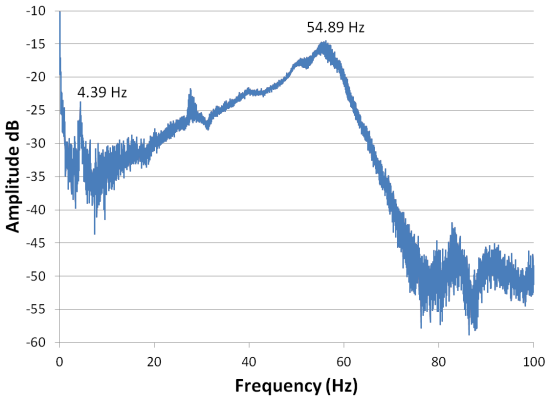
Table 2: Natural Frequency Predictions

Basis Function Choice	ω_{n1}	ω_{n2}
True Values	4.81Hz	66.63Hz
Poly - Series Length 2	4.81Hz	89.32Hz
Poly - Series Length 3	4.81Hz	66.9Hz
TMM MShape - Series Length 2	4.81Hz	66.64Hz

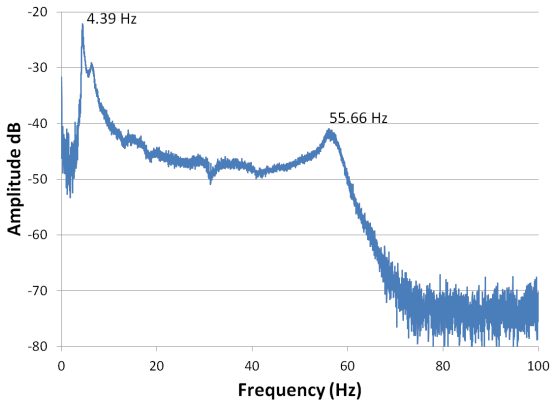
As indicated in Table 2, using the basis functions determined through the TMM modeling approach provides an accurate approximation of the true natural frequencies without having to expand the series length. The use of the TMM mode shapes as basis functions eliminates the necessity of using higher order models to achieve low order mode accuracy. Given the bandwidth of the control system for this application, models beyond the second mode of vibration are of little use. This is because the controller cannot realistically alter the response of the higher order modes.



(a) FFT Using Sensor on the Cart



(b) FFT Using Sensor Near the Middle of the Beam



(c) FFT Using Sensor at End of Arm

Figure 13: FFTs of Acceleration Measurements

From FFTs of the system shown in Figure 13, captured via accelerometers placed on the flexible link, the damped natural frequencies of the system are approximately 4.4 and 55Hz respectively. The model predictions place the damped natural frequencies at 4.8 and 56.8Hz. These experimental results are in good agreement ($< 10\%$ error) with the predicted frequencies but indicate some small persistent level of modeling error.

3.7 State Space Models From Equations of Motion

In order to cast the generated equations of motion into state space form, i.e.

$$\dot{z} = Az + Bu \tag{114}$$

$$y = Cz + Du \tag{115}$$

,

the equations of motion are converted to their modal form by solving the eigenvalue problem $(K - \omega^2 M)v = 0$, resulting in the natural frequencies and eigenvalues of the system. The eigenvectors are then normalized by their modal masses

$$\phi_j = \frac{v_j}{\sqrt{v_j^T M v_j}} \tag{116}$$

and organized into the normal mode matrix Φ where

$$\Phi = \begin{bmatrix} \phi_1 & \dots & \phi_n \end{bmatrix} \tag{117}$$

Substitution of the generalized coordinates in (38) by the state transformation $q = [\Phi]\eta$, the modal equation of motion can be obtained

$$\ddot{\eta} + \left[\hat{C}\right]\dot{\eta} + \left[\omega^2\right]\eta = [\Phi]^T Q \tag{118}$$

where η are the modal coordinates of the system, $\hat{C}$ is the modal damping matrix, and ω^2 is a diagonal matrix of the systems natural frequencies in the following order: the first beam mode, the belt-cart mode, and the second beam mode. Defining the state vector

$$z = \begin{Bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \dot{\eta}_1 \\ \dot{\eta}_2 \\ \dot{\eta}_3 \end{Bmatrix} \tag{119}$$

yields the state space realization

$$\dot{z} = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ -\omega^2 & -\hat{C} \end{bmatrix} z + \begin{bmatrix} 0_{3\times1} \\ \Phi^T Q \end{bmatrix} u \tag{120}$$

Determining the output and feed-through relationship requires the selection of a sensor (choices of sensors for flexible state recovery will be discussed in Chapter 4). For example, assume the measured output is the position of a point p along the link, then given $q = [\Phi]\,\eta$,

$$x_p = \begin{bmatrix} 1 & q_{f_1} & \dots & q_{f_n} \end{bmatrix} [\Phi]\,\eta \tag{121}$$

For the system model with parameters listed in Table 1 with belt displacement input and an accelerometer placed at the tip for output measurement, the pole zero map is displayed in Figure 14.

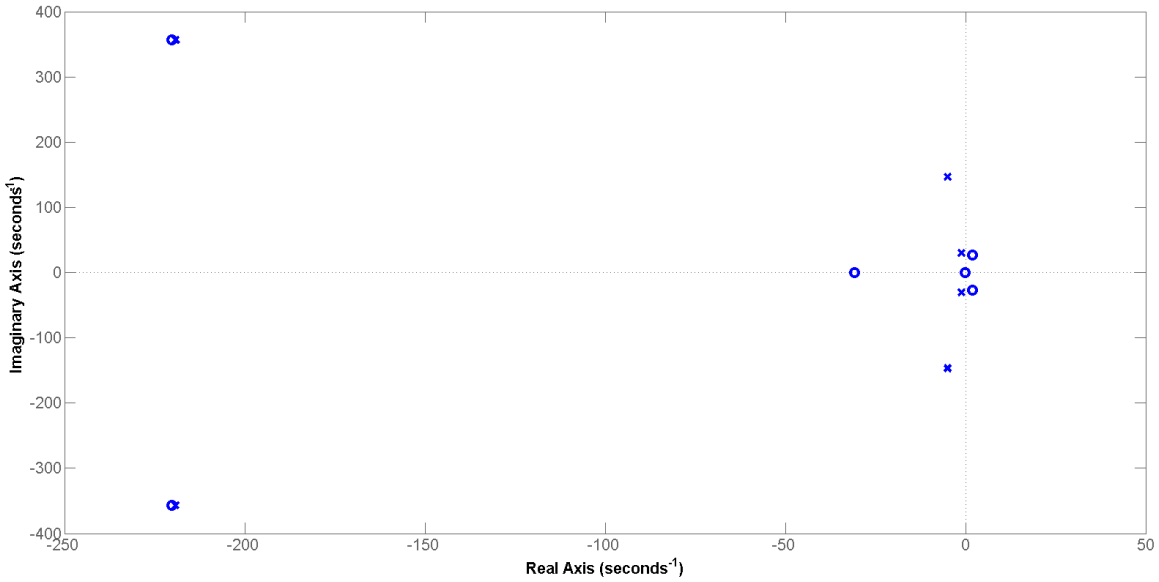


Figure 14: Pole/Zero Map Tip Accelerometer Feedback

Note that the system exhibits both non-minimum phase behavior and near pole zero cancellation of the 2nd beam mode of vibration. Satisfied that a simple, accurate model of the system can be determined, the question becomes "Where should sensors be located to provide the most robust state estimates?"

Chapter IV

SENSING STRATEGIES FOR STATE FEEDBACK ESTIMATION

4.1 Measuring Flexible State Variables

Reiterating the assertion that the deflection of a link in flexure can be separated into a spatial dependent term and a time dependent term, and assuming the mode shape of the link is known exactly, the problem of measuring and ultimately controlling link deflection becomes a matter of determining the time dependent flexible coordinates q_i where $i = 1, 2, \ldots, \infty$. From q_i and $\psi_i(x)$ any desired position output quantity can be calculated, (i.e. tip position, with velocities and accelerations requiring $\dot{q}_i$ (to be discussed)).

Assume that the link deflection at a point x_s is directly measurable as in:

$$w(x_s, t) = \sum_{i=1}^{\infty} \psi_i(x_s) q_{f_i}(t) \tag{122}$$

However, the contributions of higher order modes is significantly diminished. And for most practical purposes, this series is truncated to a finite number of modes n. Therefore,

$$w(x_s, t) \approx \psi_1(x_s) q_{f_1}(t) + \psi_2(x_s) q_{f_2}(t) + \ldots + \psi_n(x_s) q_{f_n}(t) \tag{123}$$

Or more simply described by the vector equation,

$$w(x_s, t) \approx \begin{bmatrix} \psi_1(x_s) & \psi_2(x_s) & \ldots & \psi_n(x_s) \end{bmatrix} \begin{bmatrix} q_{f_1}(t) \\ q_{f_2}(t) \\ \vdots \\ q_{f_n}(t) \end{bmatrix} \tag{124}$$

If sufficient distinct, measurements p are available where $p \geq n$ then the flexible coordinates can be calculated in a least squares sense as in

$$\left[\Psi^T\Psi\right]^{-1}\Psi^T\begin{bmatrix} w(x_1,t) \\ w(x_2,t) \\ \vdots \\ w(x_n,t) \\ \vdots \\ w(x_p,t) \end{bmatrix} \approx \begin{bmatrix} q_{f_1}(t) \\ q_{f_2}(t) \\ \vdots \\ q_{f_n}(t) \end{bmatrix} \tag{125}$$

where,

$$\Psi = \begin{bmatrix} \psi_1(x_1) & \psi_2(x_1) & \dots & \psi_n(x_1) \\ \psi_1(x_2) & \psi_2(x_2) & \dots & \psi_n(x_2) \\ \vdots & \vdots & & \vdots \\ \psi_1(x_n) & \psi_2(x_n) & \dots & \psi_n(x_n) \\ \vdots & \vdots & & \vdots \\ \psi_1(x_p) & \psi_2(x_p) & \dots & \psi_n(x_p) \end{bmatrix} \tag{126}$$

For example, imagine the pinned-pinned beam in bending shown in Figure 15 with sensors S_1 and S_2 located at $\frac{L}{2}$ and $\frac{L}{4}$ respectively.

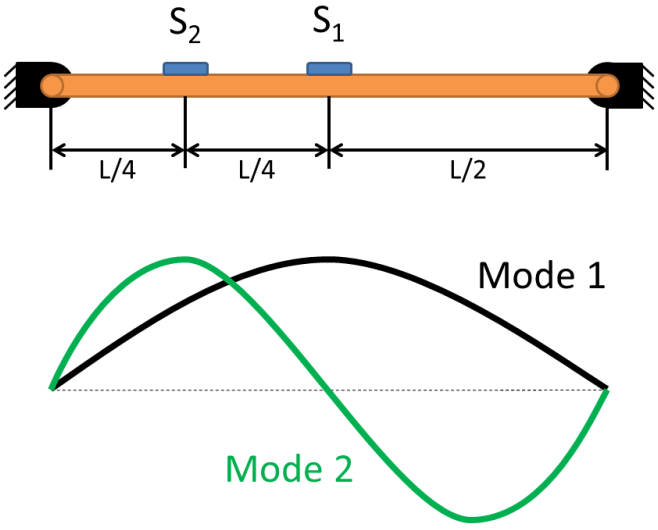


Figure 15: Pinned - Pinned Beam and Modes of Vibration

The corresponding basis functions are

$$\psi_i(x) = \sin(\frac{i\pi x}{L}) \tag{127}$$

and therefore,

$$\Psi = \begin{bmatrix} 1 & 0 \\ .707 & 1 \end{bmatrix} \tag{128}$$

The generalized flexible coordinates can be determined via,

$$\begin{bmatrix} 1 & 0 \\ -.707 & 1 \end{bmatrix} \begin{bmatrix} w_{S_1}(t) \\ w_{S_2}(t) \end{bmatrix} = \begin{bmatrix} q_{f_1}(t) \\ q_{f_2}(t) \end{bmatrix} \tag{129}$$

Note that because the first sensor lies in the node of the 2^{nd} mode of vibration, it is not correlated with the 2^{nd} generalized coordinate. If instead, two sensors were chosen very close to the node, then large gains would be associated with the 2^{nd} generalized coordinate. This would result in high noise sensitivity and low accuracy. In addition, the full system state consists not only of rigid coordinates and flexible coordinates, but also their derivatives. Thus, the number of necessary sensors is in fact $2n$, or twice the number of considered modes. Conversely a numeric derivative of the determined generalized coordinates could be taken at the expense of extra measurement noise.

To estimate the full state of the system from a limited set of sensors, a state observer can be utilized which utilizes a model of the system to extrapolate the missing state information.

As illustrated in Figure 16, an observer is a dynamic system which acts as a closed loop control system with a loop gain L. This gain drives the model predicted state estimates $\hat{z}$ to the true state z by acting on the error between estimated and measured output quantities. As a dynamic system, the performance characteristics can be modified via the loop gain to alter convergence properties and filter measurement noise. The addition of a state observer to the complete controlled system adds observer poles, effectively doubling the number of system poles. The additional poles are associated with the convergence of the estimation error and, as a result of the eigenvalue separation principle, can be placed irrespective of the

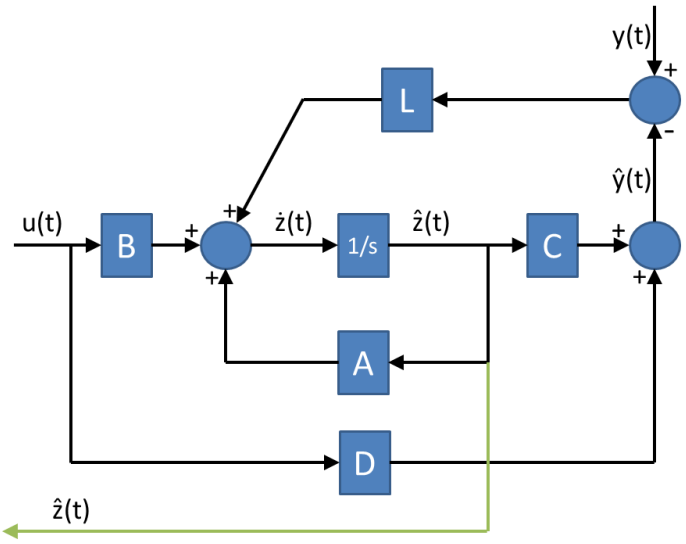


Figure 16: Observer Diagram

locations of the controller poles. However, their placements have important repercussions on the system dynamics; too fast and measurement noise is prevalent, too slow and the closed loop system performance will suffer.

When referring to the task of measuring flexible state variables, it is assumed going forward that the primary vehicle will be variations on a common linear state estimator.

4.2 Sensor Types and State Relationships

4.2.1 Strain Measurements

While direct measurement of link deflection at multiple locations might be possible through machine vision or other optical sensing techniques, in an industrial environment where structured lighting is difficult to maintain, other sensor types are preferable. The most common sensors used for measuring the state of flexible structures are strain gages (Figure 17).

A strain gage operates on a simple principle, small conductors are oriented parallel to one another and connected in series. Affixed to the structure with an adhesive, the geometry of the conductors necessarily changes as the material is stretched, increasing resistance for elongation and decreasing resistance for compressive strain. Figure 18 illustrates the strain on a beam in bending.

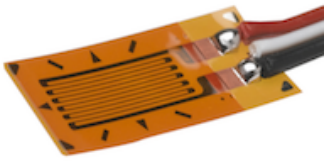


Figure 17: Strain gage

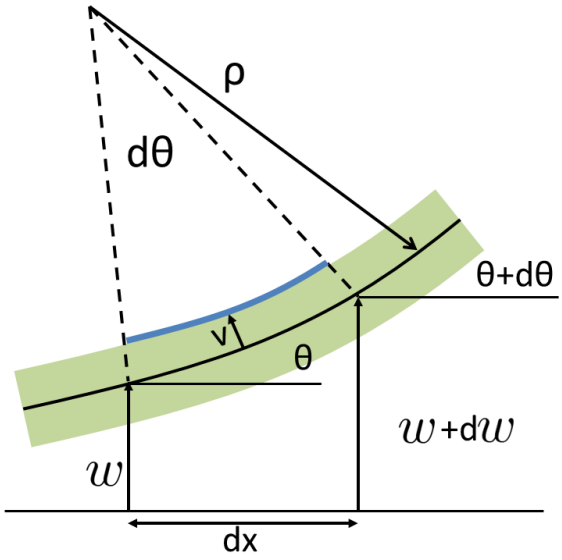


Figure 18: Beam Strain Diagram

Initially the strain gage on the surface of a beam element has an undeformed length of dx. In bending, this length is changed to dx' where,

$$dx' = (\rho - v)d\theta = dx - vd\theta \tag{130}$$

The strain, therefore, can be represented as

$$\epsilon = \frac{dx' - dx}{dx} = -v\frac{d\theta}{dx} \tag{131}$$

where $\frac{d\theta}{dx}$ is the curvature of the beam element. From (49), it has already been determined that the curvature of a beam element can be related to the second derivative of the

mode shapes and the corresponding flexible generalized coordinates.

$$\Rightarrow \epsilon(t) = -v\frac{d^2w(x_s,t)}{dx^2} = -v\sum_{i=1}^{\infty}\frac{d^2\psi_i(x_s)}{dx^2}q_{f_i} \tag{132}$$

or in modal form,

$$\epsilon(t) = -v\begin{bmatrix} \frac{d^2\psi_1(x_s)}{dx^2} & \frac{d^2\psi_2(x_s)}{dx^2} & \dots & \frac{d^2\psi_n(x_s)}{dx^2} \end{bmatrix}\Phi\eta \tag{133}$$

Thus, the output-state relationship for a general flexible manipulator with a modal state space representation is:

$$\epsilon(t) = -v\left[\begin{bmatrix} 0 & \frac{d^2\psi_1(x_s)}{dx^2} & \frac{d^2\psi_2(x_s)}{dx^2} & \dots & \frac{d^2\psi_n(x_s)}{dx^2} \end{bmatrix}\Phi \quad \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \end{bmatrix}\right]z + 0u \tag{134}$$

given a strain measurement at position x_s at a distance v from the beam's neutral axis.

4.2.2 Pros and Cons of Strain Gage Measurements

Strain gage measurements have been shown to be relatively successful in reconstructing the flexible system states by many researchers. To facilitate direct comparison with acceleration measurements for this task the pros and cons of strain sensors are listed below.

- Pros
  - Strain gages operate on a simple principle.
  - Gages themselves are relatively inexpensive $\approx$ \$10+ [1].
  - They exhibit no direct feed-through in output-state relationship.
- Cons
  - The uutput range is measured in mV or μV and, therefore, requires extra signal conditioning before ADC.

[1] As of May, 2013

- For accurate measurements, signal conditioning units can be relatively expensive $\approx \$200$ [1].
- The reliability and accuracy of gage measurements is very sensitive to environmental conditions, including
  - Temperature fluctuations (compensation with "dummy gage")
  - Self-heating (applied voltage heats gage changing resistance value) → low driving/measurement voltages
  - Bonding effects (poor adhesion results in inaccurate readings, and local stiffness variation caused by bonding agent)
  - Sensitivity to electromagnetic interference
  - Overloading - strain beyond elastic limit will introduce a permanent bias and degrade performance
- There is no link to rigid state variables (e.g. joint motion).

In summary, strain measurements are an effective and accurate method for determining the state variables of a flexible robotic manipulator. However, in order to function reliably, environmental factors must mitigated using expensive and complex signal conditioning equipment. (it is interesting to note that the origin of the term "Murphy's law" was coined after a failed implementation of strain gage conditioning equipment.)

4.2.3 Acceleration Measurements

Accelerometer technology has, in recent years, advanced significantly. Driven in large part by the ubiquity of mobile devices (e.g. smart phones, tablets, gaming systems) and improvements in micro and nano-scale manufacturing, micro-electromechanical systems (MEMs) accelerometers are readily available and cost effective.

The acceleration sensor pictured in Figure 19 is of the type used in this analysis. The Analog Devices ADXL325 3 axis $\pm 5g$ MEMs accelerometers, is shown with conditioning filter capacitors on the custom circuit board. Each accelerometer, fully assembled, is priced[2] at

[2] As of August 2012

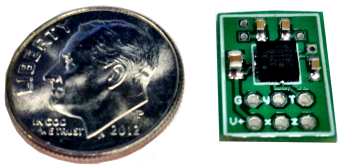


Figure 19: Analog Devices 3 Axis MEMs Accelerometer

$6.91.

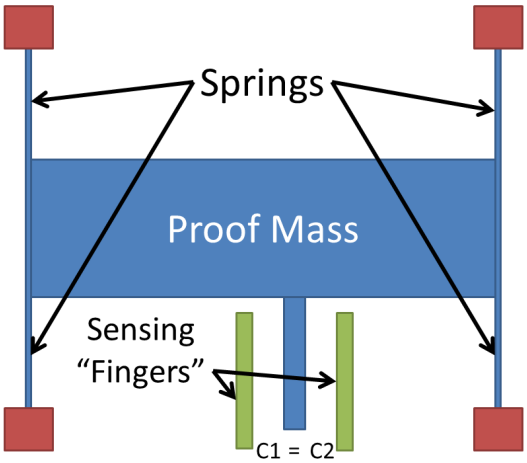


Figure 20: MEMs Accelerometer - Zero State

The operating principle behind MEMs accelerometers is very simple. A proof mass is affixed to spring elements as shown in Figure 20 (Note that while only one set is pictured, in practice many are utilized in parallel). Voltage is applied to the mass/spring system and measured at the "sensing fingers." In the nominal, undeformed state, the capacitance between the left and right fingers and the proof mass is equal. Therefore, the voltage measured at the left and right is identical.

In the presence of acceleration pictured in Figure 21, the springs deform and shift the position of the proof mass in proportion to the subjected acceleration. The mass shift results in a change of capacitance between the left and right fingers and therefore a voltage measurement proportional to the acceleration.

The relationship between measured acceleration, state, and input is easily determined.

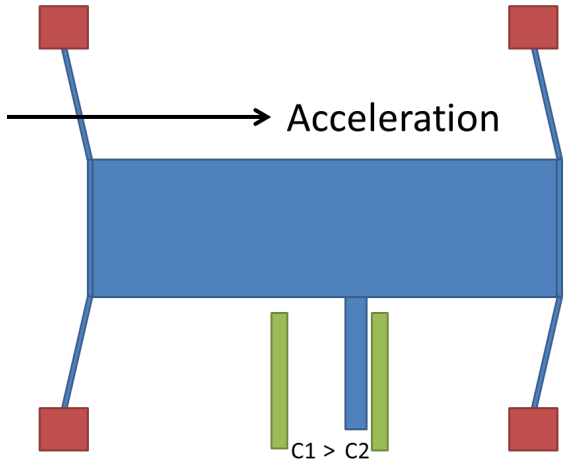


Figure 21: MEMs Accelerometer - Deformed State

For the FLASHMan flexible manipulator the acceleration of a point p on the flexible link is

$$\ddot{x}_p = \ddot{x}_c + \ddot{w}(y_p, t) \tag{135}$$

where $\ddot{x}_c$ is the acceleration of the cart and y_p is the location of the sensor along the length of the beam. Therefore, as a function of the generalized flexible coordinates,

$$\ddot{x_p} = \ddot{x_c} + \sum_{i=1}^{n} \psi_i (y_p) \ddot{q}_{f_i}(t) \tag{136}$$

Since $q = [\Phi]\eta$ then it follows that $\ddot{q} = [\Phi]\ddot{\eta}$ must also be true. Therefore, from (118):

$$\ddot{q} = [\Phi]\ddot{\eta} = -[\Phi]\left[\omega^2\right]\eta - [\Phi]\left[\hat{C}\right]\dot{\eta} + [\Phi][\Phi]^T Q \tag{137}$$

The tip acceleration is a function of the cart acceleration and the acceleration of a beam element at the tip, i.e.

$$\ddot{x_p} = \ddot{x_c} + \sum_{i=1}^{n} \ddot{q}_{f_i} \psi (y_p) \tag{138}$$

where

$$\ddot{x}_c = [1, 0, \ldots, 0]\ddot{q} \tag{139}$$

and

$$\sum_{i=1}^{n} \ddot{q}_{f_i} \psi\left(y_p\right) = \left[0, \psi_1\left(y_p\right), \ldots, \psi_n\left(y_p\right)\right] \ddot{q} \tag{140}$$

which means that,

$$\ddot{x_p} = \left[1, \psi_1\left(y_p\right), \ldots, \psi_n\left(y_p\right)\right] \ddot{q} \tag{141}$$

it follows that $\beta(y_p) = \left[1, \psi_1\left(y_p\right), \ldots, \psi_n\left(y_p\right)\right]$. Then

$$\ddot{x_p} = -\beta(y_p)\left[\Phi\right]\left[\omega^2\right]\eta - \beta(y_p)\left[\Phi\right]\left[\hat{C}\right]\dot{\eta} + \beta(y_p)\left[\Phi\right]\left[\Phi\right]^T Q u \tag{142}$$

The equation (142) determines the output and feed-through matrices C and D, where

$$C = \begin{bmatrix} -\beta(y_p)\left[\Phi\right]\left[\omega^2\right] & -\beta(y_p)\left[\Phi\right]\left[\hat{C}\right] \end{bmatrix} \tag{143}$$

and

$$D = \begin{bmatrix} \beta(y_p)\left[\Phi\right]\left[\Phi\right]^T Q \end{bmatrix} \tag{144}$$

Therefore the resulting output-state relationship is

$$\ddot{x}_p = \begin{bmatrix} -\beta(y_p)\left[\Phi\right]\left[\omega^2\right] & -\beta(y_p)\left[\Phi\right]\left[\hat{C}\right] \end{bmatrix} z + \begin{bmatrix} \beta(y_p)\left[\Phi\right]\left[\Phi\right]^T Q \end{bmatrix} u \tag{145}$$

4.2.4 Pros and Cons of Acceleration Measurements

Acceleration measurements offer some significant practical benefits over strain sensing.

- Pros
  - Low cost MEMs accelerometers are readily available.
  - Sensors are robust to environmental factors (e.g for ADXL325) in that
    - They can be subjected to 2000× the maximum rated acceleration value.
    - They have large operating temperature ranges ($-40°C$ to $+85°C$).
  - They have an output range $0 - 3.3V$ without additional amplification.

- There is a direct link to rigid state variables (no additional sensor necessary).
- Digital versions are available which reduce sensitivity to EMI in transmission lines.

- Cons
  - There is a direct feedthrough in the output-state relationship.
  - The output-state relationship is dependent on the model parameters.
  - Drift is a possibility at low frequencies.

4.3 Sensor Placement

The process of selecting placements for sensors of a flexible system can be closely correlated with the observability gramian. The observability gramian provides a method of determining if the initial state of a system ($x_0(t_0)$) can be characterized by a given output measurement ($y(t)$).

In order to derive the observability gramian, first consider the following state equations:

$$\begin{aligned} \dot{z}(t) &= Az(t) + Bu(t) \\ y(t) &= Cz(t) + Du(t) \end{aligned} \tag{146}$$

which can be discretized to form (147).

$$\begin{aligned} z_{k+1} &= A_d z_k + B_d u_k \\ y_k &= C_d z_k + D_d u_k \end{aligned} \tag{147}$$

Assuming for a moment that the system is autonomous, i.e. $u_k = 0$, without loss of generality, then the relationship between the future state (z_{k+1}) and the current state (z_k) is simply

$$z_{k+1} = A_d z_k \tag{148}$$

and therefore,

$$y_{k+1} = C_d z_{k+1} = C_d A_d z_k \tag{149}$$

If the goal is to recreate the initial condition z_0 from a series of output measurements $y_0, y_1, y_2, \ldots, y_p$, where $p > n$ this problem becomes a straight forward least squares problem. Continuing the pattern of (149)

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_p \end{bmatrix} = \begin{bmatrix} C_d A_d^0 \\ C_d A_d^1 \\ C_d A_d^2 \\ \vdots \\ C_d A_d^p \end{bmatrix} z_0 \tag{150}$$

Pre-multiplying both sides by the transpose of the RHS matrix,

$$\overbrace{\begin{bmatrix} A_d^{0^T} C^T & A_d^{1^T} C^T & A_d^{2^T} C^T & \cdots & A_d^{p^T} C^T \end{bmatrix}}^{(n\times p)} \overbrace{\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_p \end{bmatrix}}^{(p\times 1)} =$$

$$\overbrace{\begin{bmatrix} A_d^{0^T} C^T & A_d^{1^T} C^T & A_d^{2^T} C^T & \cdots & A_d^{p^T} C^T \end{bmatrix}}^{(n\times p)} \overbrace{\begin{bmatrix} CA_d^0 \\ CA_d^1 \\ CA_d^2 \\ \vdots \\ CA_d^p \end{bmatrix}}^{(p\times n)} z_0^{(n\times 1)} \tag{151}$$

Written more succinctly,

$$\sum_{i=0}^{p} A_d^{i^T} C_d^T y_i = \left[\sum_{i=0}^{p} A_d^{i^T} C_d^T C_d A_d^i\right] z_0 \tag{152}$$

And, therefore, the initial condition, z_0, for a given series of measurements, $y_0, \ldots, y_p$, is

$$\left[\sum_{i=0}^{p} A_d^{i^T} C_d^T C_d A_d^i\right]^{-1} \sum_{i=0}^{p} A_d^{i^T} C_d^T y_i = z_0 \tag{153}$$

where $\left[\sum_{i=0}^{p} A_d^{i^T} C_d^T C_d A_d^i\right]$ is the discrete observability gramian W_{o_d}. Therefore, for a solution to exist, W_{o_d} must be invertible (i.e., $rank(W_{o_d}) = n$). If the system is stable then z_0 will approach a steady-state value as $p \to \infty$. Therefore,

$$\lim_{p\to\infty} W_{o_d} = W_{o_d}^{\infty} \tag{154}$$

where

$$W_{o_d}^{\infty} = \sum_{i=0}^{\infty} A_d^{i^T} C_d^T C_d A_d^i \tag{155}$$

is the steady-state discrete observability gramian. Beyond a simple litmus test for observability, it provides a description of the state energy corresponding to a selected sensor [19, 54]. Take for example the sensor energy, $\|y\|$, where

$$\|y\| = y^T y = z_o^T W_{o_d} z_o \tag{156}$$

The singular value decomposition of the observability gramian, indicates the directions of high state influence for a given sensor measurement. In fact, one can construct an "ellipsoid of detectability", describing this correlation. Similar to the ellipsoid of dexterity for a manipulator which describes the relative ease of motion in any direction, the ellipsoid of detectability describes the relative contribution of energy from an output to the initial state vector:

$$W_{o_d} = U \Sigma V^T \tag{157}$$

where U and V represent the left and right singular vectors and describe how the state energy is divided. The representation in (157) can be replaced by a summation of rank one matrices:

$$W_{o_d} = \sum_{i=1}^{n} \sigma_i u_i v_i^T \tag{158}$$

For a modal system, the singular vectors are along a single axis of the state space, as demonstrated in Figure 22.a. Therefore, if the singular values are sorted according to their state associations rather than their magnitudes as in:

$$W_{o_d} \approx I \Sigma_z I^T = \Sigma_z \tag{159}$$

then

$$\|y\| = \sigma_{z_1} z_{0_1}^2 + \sigma_{z_2} z_{0_2}^2 + \ldots + \sigma_{z_n} z_{0_n}^2 \tag{160}$$

The condition number of the observability gramian, therefore, describes the symmetry of the ellipsoid of detectability

$$Q = \frac{1}{CN} = \frac{min(\sigma_z)}{max(\sigma_z)} \tag{161}$$

Q defines the "observation quality" of a sensor, which describes the relative contribution of the least observable state compared to the most observable state [88]. If $Q = 1$, then all states are equally observable from a given measurement, and the ellipsoid of detectability becomes a sphere (for $n = 3$). If $Q = 0$ then at least one state is unobservable and the ellipsoid of detectability loses a dimension. One of the singular values is zero in Figure 22(b) leading to the representation of the observability gramian as a summation of $n - 1$ rank 1 matrices. Under this condition, the solution of (150) is indeterminate from the given sensor information.

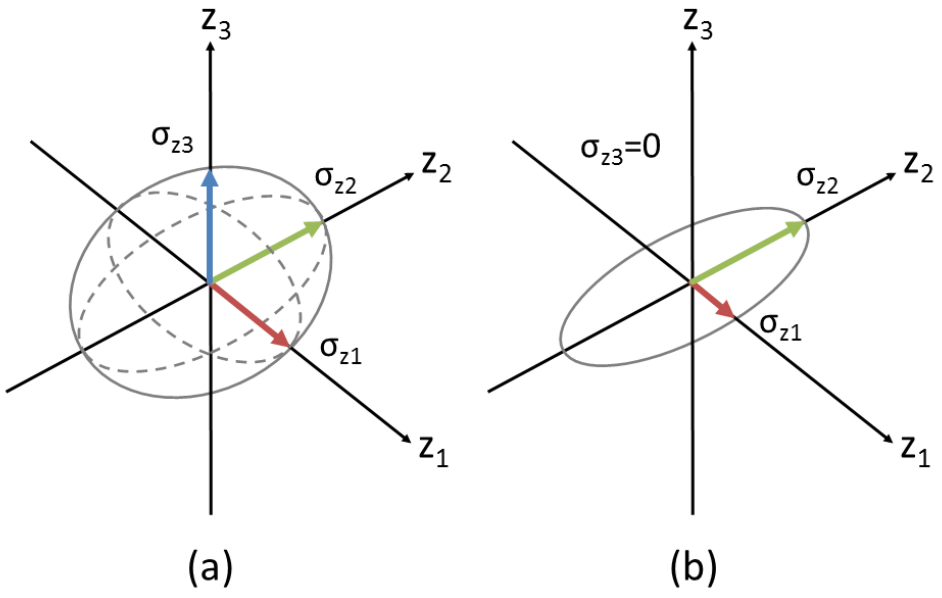


Figure 22: Ellipsoid of Detectability: (a) Complete Observability, (b) z_3 Unobservable

The magnitude of the singular values also plays a role in the ability of a specific sensor to accurately describe the behavior of a specific mode of vibration. For example, a sensor which is placed at the end of a flexible beam will have a large singular value associated with the state that corresponds to the first mode of vibration. However in all likelihood, the state corresponding to the 2^{nd} mode of vibration will have a small singular value, indicating a small portion of that sensor's energy is being used to determine that state. Thus while a large singular value indicates a good estimate of the current state value based on a significant measurement, a small singular value indicates a poor estimate based on limited information.

While the discussion above is limited to discussions of single sensors, the logic of the analysis can be applied to groups of sensors as well.

4.3.1 Simulations of FLASHMan Sensor Placements

Computing the singular values of the observability gramian for a sensor placed along the length of the arm at candidate points yields curves which describe the ability of an estimator to reconstruct an individual state from the chosen measurement. For a single accelerometer placed on the FLASHMan testbed, these curves are shown in Figure 23.

Note that because acceleration is the chosen measurement, the magnitude of the singular values attributed to higher order modes is much higher than the low order modes, (i.e. the expected accelerations for equal excitations of the modes would be larger for the higher order modes). In reality, these modes are damped more significantly than the lower order modes resulting in behavior primarily attributed to the low frequency dynamics.

Because of the way the singular values of a matrix are organized in the chosen computational tool (Matlab), i.e. by magnitude rather than by state correspondence, the singular vectors are used to associate the singular values with their corresponding state variable. When two or more singular values are relatively close in magnitude, the singular vectors associated with each singular value may be selected arbitrarily: the only constraint is that they span the proper subspace. Since the singular vectors are used to determine the state correspondence, when the singular values of two separate states are close in value, the incorrect state is often chosen. This results in the discrete jumps obvious in Figure 23. These

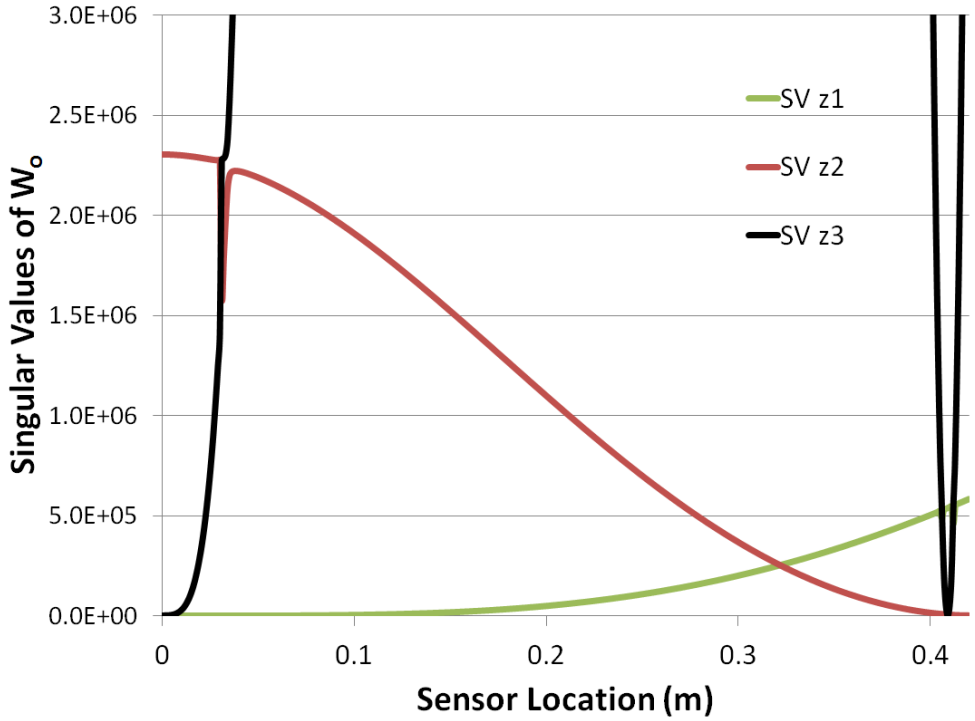


Figure 23: Singular Values of the Observability Gramian vs. Sensor Location

locations are, nevertheless, important, as they represent sensor positions where the states are equally observable, (i.e. the output energy is equally distributed between the system modes).

Figure 24 is a close up of Figure 23 near the location where the singular values for the first three states are almost equivalent in value. At this point the system is "most equally observable" and, thus, is in the desired sensor location offering the highest sensor quality. However, this location is precariously near to where the singular value related to state z_3 is zero (i.e. state z_3 is unobservable, and z_2 is nearly unobservable)! Given a slight shift in the system parameters, this sensor could fall into the node of mode 3 and be blind to the energy stored in that mode of vibration. If this were to occur, system performance may be reduced and/or the system may become unstable.

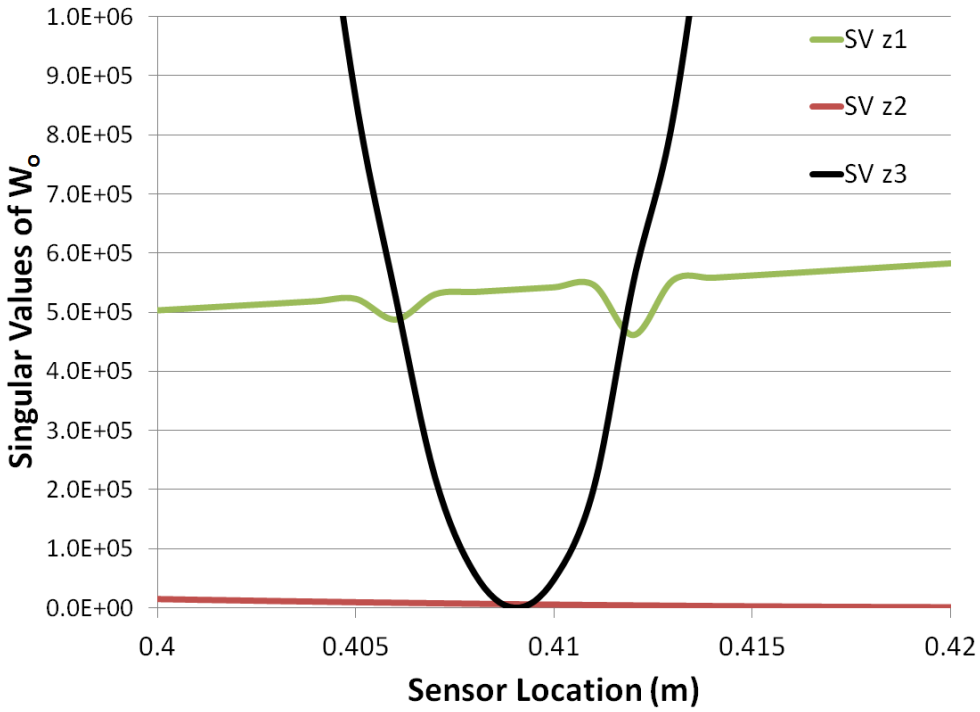
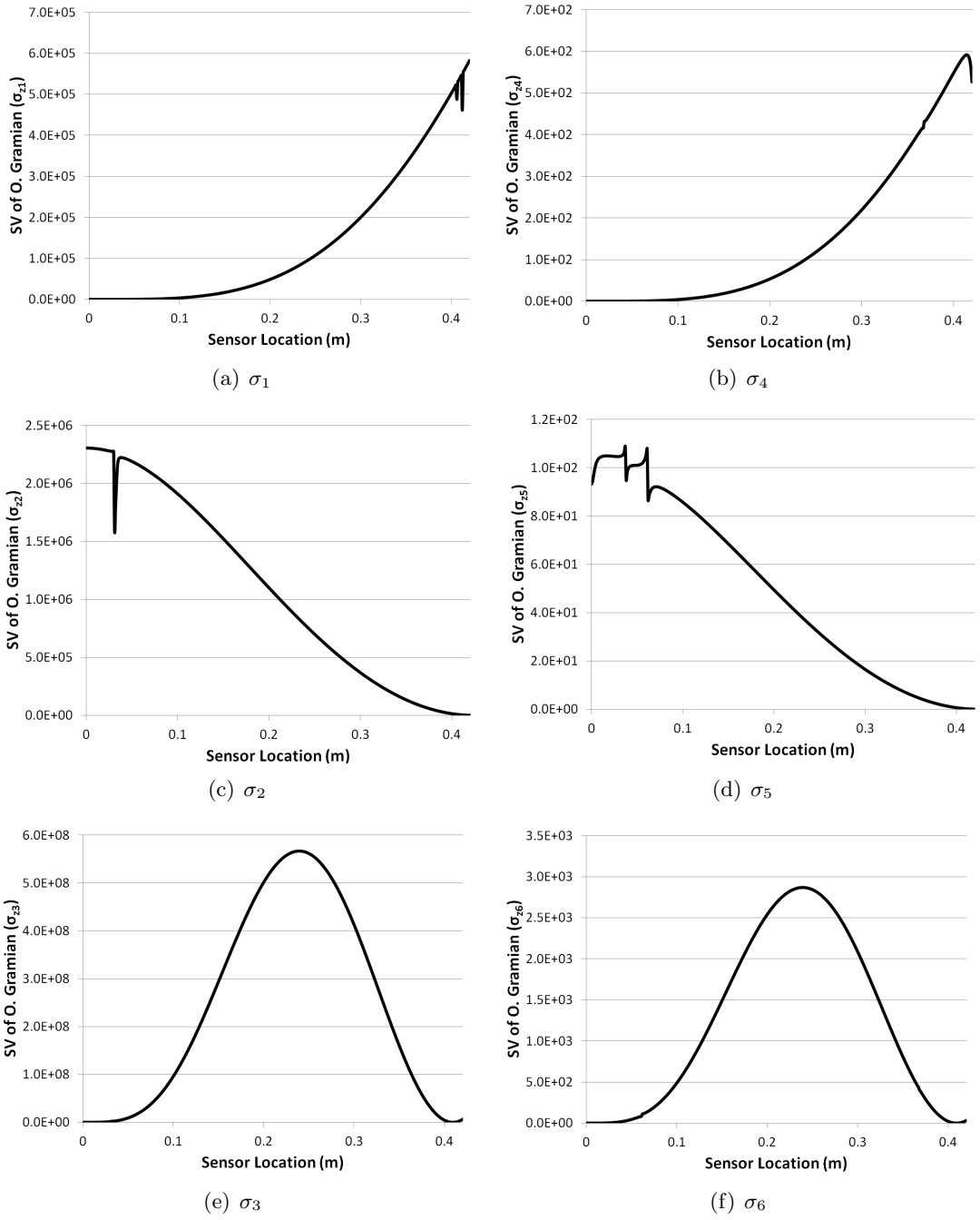


Figure 24: Singular Values of the Observability Gramian vs. Sensor Location (Zoomed)

Figures 25(a)-25(f) show the individual singular values, σ_i, associated with each state, z_i. The correspondence between a state and its derivative indicates that the derivatives of the modal coordinates $(z_4 \dots z_6)$ can be ignored for the purposes of establishing sensor placements, (e.g. a satisfactory sensor placement for the state z_1 will also suffice for z_4).



(a) σ_1 (b) σ_4

(c) σ_2 (d) σ_5

(e) σ_3 (f) σ_6

Figure 25: State Correspondence of SV with Sensor Location

With the reduced dimensions, the ellipsoids of detectability, originally presented in Figure 22, can be visualized as seen in Figure 26.

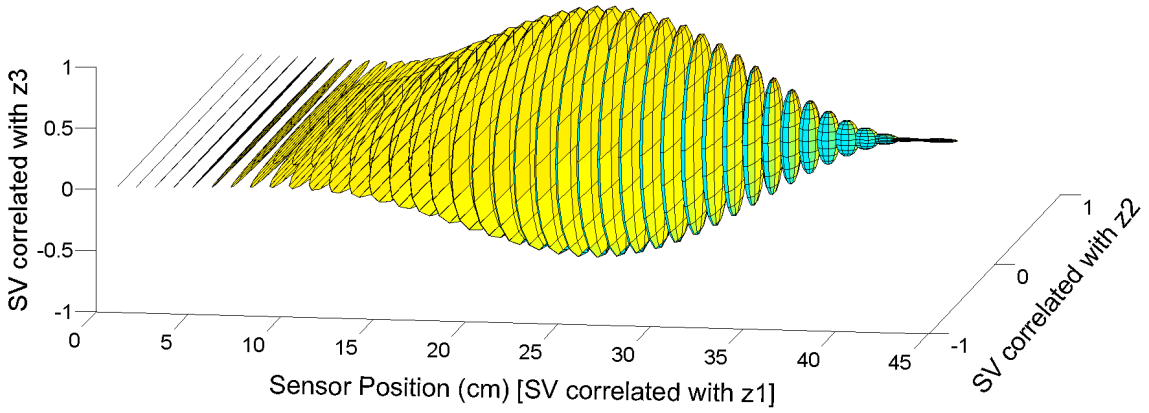


Figure 26: Elipsoids of Detectability

Each ellipsoid in Figure 26 is centered about its corresponding sensor location along the beam. The dimensions of the ellipsoid represent the state correlation of that particular sensor as described in Figure 22. On inspection this graphic procedure intuitively describes the observability of each state given a sensor position: the more spherical the ellipsoid, the more equal the balance of state information observed in the output. At the beginning of the flexible link, states z_2 and therefore z_5 are the most observable. Since these states correspond primarily to the vibration of the belt/cart system, it makes sense that a sensor placed on the cart would provide the best reconstruction of these states. The states corresponding to the vibration of the link are very weakly observable at this location. Traveling down the length of the arm, the z_3 and z_6 states become more observable peaking in magnitude near the middle of the link. The z_1 and z_4 states become the most observable near the tip of the link. Metrics can be designated for the placement of a single sensor, one choice is the location in which the ellipsoid of detectability most closely resembles a sphere (i.e. $min(\sigma_{max}/\sigma_{min})$). This condition is equivalent to the location where the condition number of the observability gramian is minimum. This criteria must be considered in conjunction with a bound on minimum elipsoid volume such that conditions of poor observability are avoided. From Figure 27. This optimal placement occurs near the tip of the beam where the quality, Q, of the sensor placement is maximum. Maximum ellipsoid volume also serves as

a viable metric for determining sensor placement, but will not be utilized in the remaining discussion.

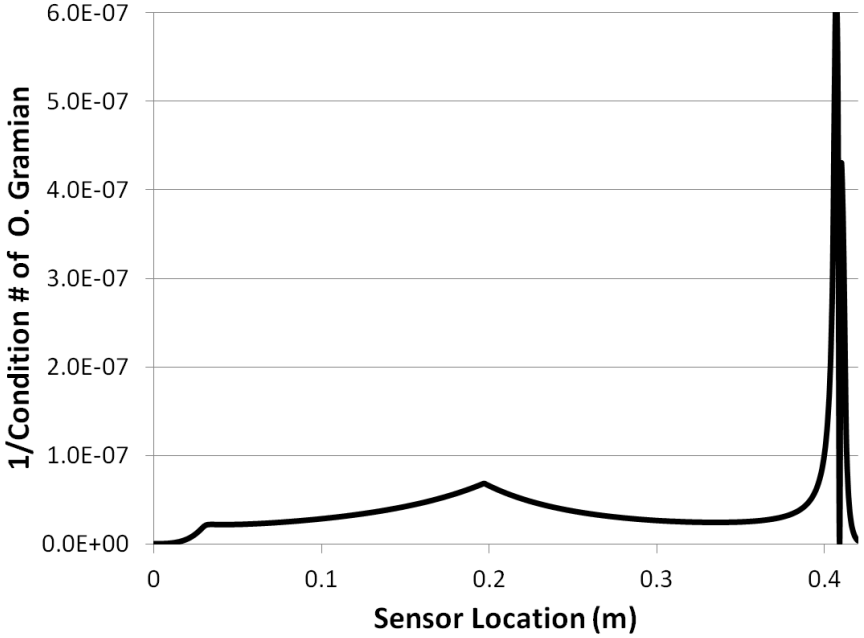


Figure 27: Sensor Observation Quality vs. Location

There are also points of modal unobservability, where no sensor energy is devoted to a particular system mode. These are points where the quality drops to zero (see Figure 27). No accurate state estimates corresponding to those modes can be determined. This problem has led to interest in the question of robustness of observability which will be discussed in section 4.7.

Applying the same analysis to strain sensing, the singular values of the observability gramian as a function of strain gage location are shown in Figure 28. From the figure it's obvious that the two beam modes are closely correlated with the link strain. The belt motion, however, exhibits significantly less correlation with the strain sensor.

In contrast with the accelerometer measurement, here, the optimal locations appear at the start of the link. These will necessarily be locations with highest curvature when deformed, rather than the locations of greatest displacement. The singular value corresponding to the second beam mode drops to zero at approximately 1.1m indicating that a gage placed at this point would result in unobservability of this mode.

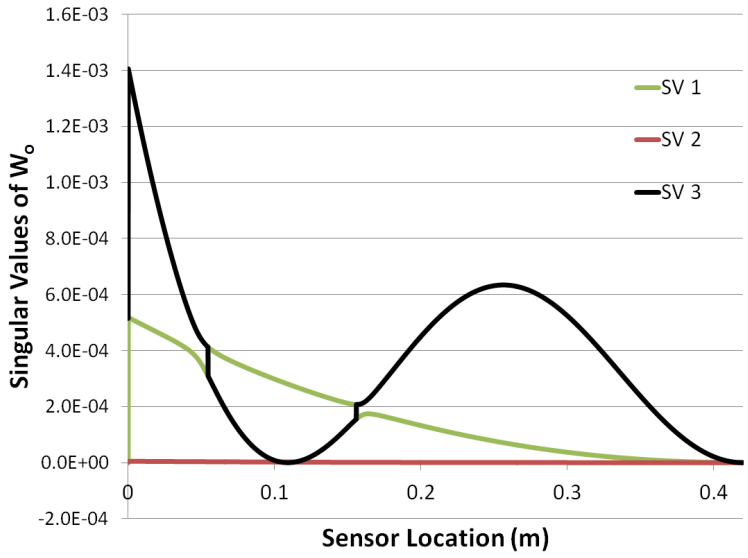


Figure 28: Singular Values of the Observability Gramian vs. Strain Gage Location

From the sensor quality shown in Figure 29, optimal placement at the first peak is determined to be around 0.075m, sacrificing observability of the beam modes in order to balance the belt/cart mode. If the limitation of placing a single sensor could be lifted, vast improvements could be made, both in terms of acceleration and strain measurements.

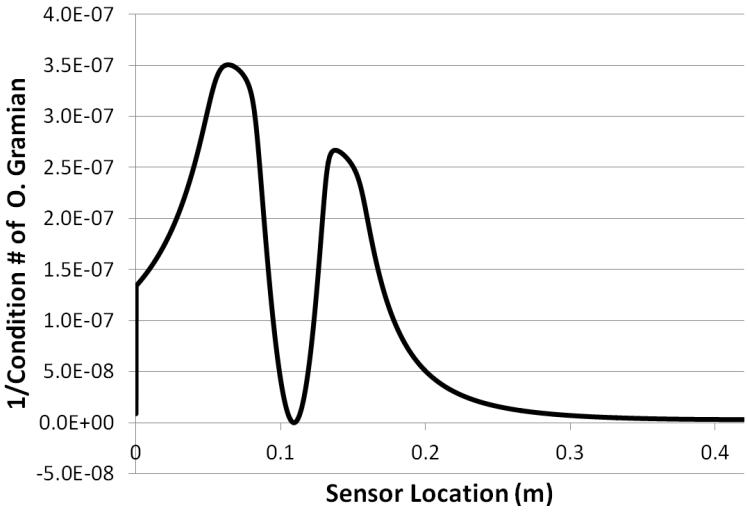


Figure 29: Strain Sensor Observation Quality vs. Location

4.4 Grouping Sensors

In order to consider the consequences of multiple sensors on state observability, the state-output description must be extended to include multiple output measurements:

$$\begin{bmatrix} y_{s_1} \\ y_{s_2} \\ \vdots \\ y_{s_j} \end{bmatrix}_k = \begin{bmatrix} C_{s_1} \\ C_{s_2} \\ \vdots \\ C_{s_j} \end{bmatrix} z_k + \begin{bmatrix} D_{s_1} \\ D_{s_2} \\ \vdots \\ D_{s_j} \end{bmatrix} u_k \tag{162}$$

More succinctly for autonomous systems ($u_k = 0$), these sensor groups can be represented as

$$y_{g_k} = C_g z_k \tag{163}$$

Therefore, taking multiple successive samples as in (150):

$$\begin{bmatrix} \begin{Bmatrix} y_{s_1} \\ y_{s_2} \\ \vdots \\ y_{s_j} \end{Bmatrix}_0 \\ \vdots \\ \begin{Bmatrix} y_{s_1} \\ y_{s_2} \\ \vdots \\ y_{s_j} \end{Bmatrix}_p \end{bmatrix} = \begin{bmatrix} \begin{Bmatrix} C_{s_1} A_d^0 \\ C_{s_2} A_d^0 \\ \vdots \\ C_{s_j} A_d^0 \end{Bmatrix} \\ \vdots \\ \begin{Bmatrix} C_{s_1} A_d^p \\ C_{s_2} A_d^p \\ \vdots \\ C_{s_j} A_d^p \end{Bmatrix} \end{bmatrix} x_0 \tag{164}$$

Therefore, there are $j \times p$ equations and n unknowns. By the Cayley Hamilton theorem, A^n can be expressed as a linear combination of lower powers of A. Necessarily, any rows of (164) with powers of A beyond $n-1$ are linearly dependent. Thus, there exist, at most, $j \times n$ linearly independent equations for a system of n unknowns.

It suffices, however, to find a set of sensors which result in n linearly independent

equations to ensure complete observability of the full state vector. This can be accomplished in numerous ways. For example, 1) given a single sensor and for which W_{o_d} is invertible and n output measurements, 2) given 3 sensors for which $W_{o_{d_i}} \quad i = 1, 2, 3$ are invertible and $\frac{n}{3}$ measurements from each sensor, and 3) any combination of sensors which contribute, as a group $\geq n$ linearly independent rows to W_{o_g} (a gramian for the sensor group). A trivial example of the 3) would be full state measurements, where a single sensor is used to measure each state independently. Thus $W_{o_g} = I_{(n \times n)}$ and the state vector is completely observable in one time step. Even though a sensor may lack the ability to observe a specific state or set of states, the addition of other sensors targeted at the remaining states can supply the missing information.

The problem of multiple sensor placement simplifies to the selection of groups and placements of sensors to maximize the observability of the full state vector, (i.e. to recover the most state information possible). By placing sensors at the locations resulting in the maximum singular values of each mode, observability is assured.

Beyond multiples of the same sensor type, integration of multiple, different sensors is often beneficial. For example, take the strain measurement used to determine Figure 28. The states corresponding to the belt/beam mode are not very observable from the strain measurement. They are, however, very highly correlated to a measurement of the cart position. Adding in the cart position measurement results in the singular values shown in Figure 30

The strain sensor placement can then be made by examining the observation quality of the augmented system as demonstrated in Figure 31.

In comparison with Figure 29 the addition of the cart position measurement frees up the strain measurement to be placed at the optimal location for measuring the flexible beam states, at the beginning of the arm.

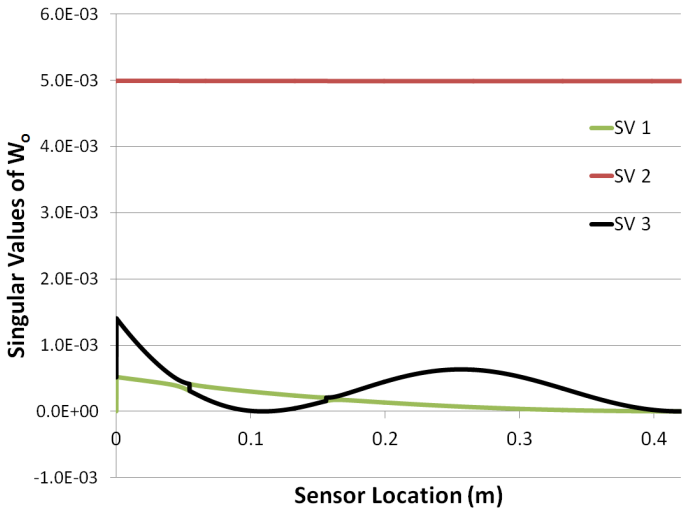


Figure 30: Singular Values of Observability Gramian of Strain Sensor vs. Location with Cart Position Measurement

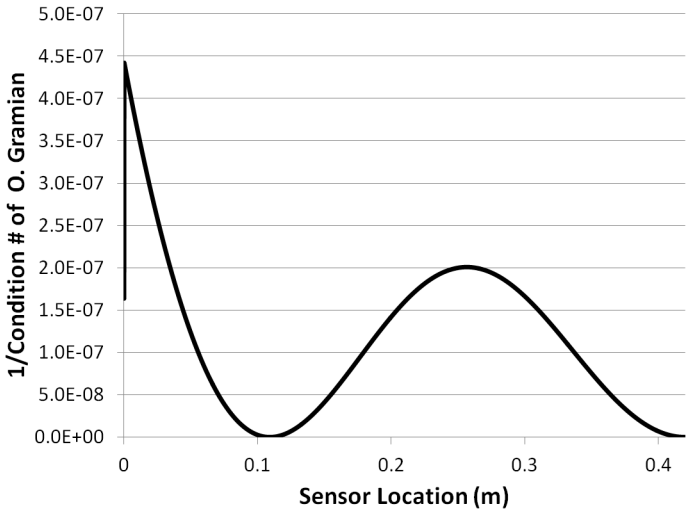


Figure 31: Cart Position + Strain Sensor Observation Quality vs. Location

4.5 Selecting Sensor Placements via Genetic Algorithm Optimization

Since the singular value decomposition of the observability gramian is unavailable in closed form and the parametrization with respect to changing sensor location is unknown, searching sets of singular values to evaluate candidate points can be computationally expensive for complex structures. In employing a genetic algorithm, the placement process can be simplified and evaluated more quickly than random search. The genetic algorithm also offers greater precision than evaluating candidate locations uniformly distributed over the entire arm length.

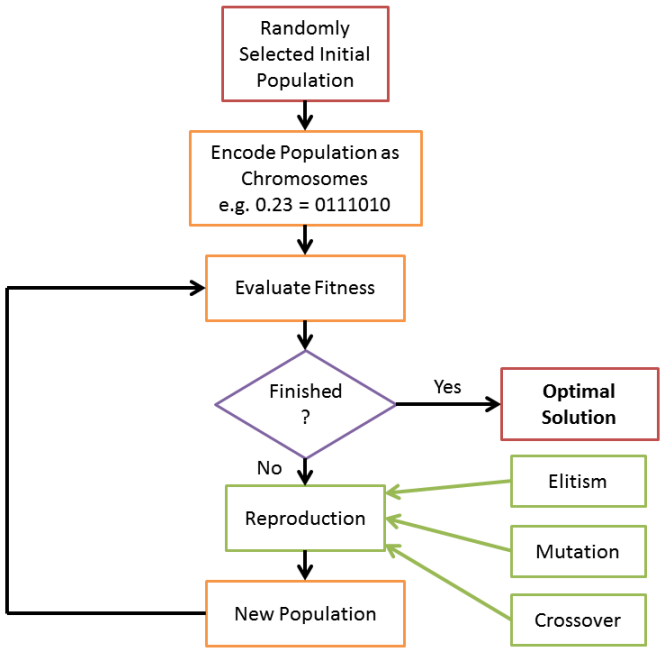


Figure 32: Genetic Algorithm Overview

A general overview of genetic algorithms is provided in Figure 32. Initially a population of potential sensor locations is randomly generated. Each sensor location in this set is evaluated through the computation of a fitness function. This function provides the selection criteria on which the sensor location is judged, for example to select a single sensor for reconstructing all the states, a simple fitness function would be the maximum of the sensor quality Q.

After evaluating the fitness of the individual sensors in the population, they are checked against a desired metric to determine if the optimal solution has been found. For example,

if the best solution hasn't changed for a number of generations, then it is likely the best solution. If deemed insufficient to stop the process, the individuals are ordered by their fitness and assigned reproductive probabilities based on their fitness. Therefore, an individual sensor with higher fitness is more likely to be selected for reproduction than one with a lower fitness. Often the first one or two "elite" individual's chromosomes are directly copied into the next population. Otherwise pairs are selected in a weighted random fashion based on their reproductive probability (fitness). Each pair selected for reproduction can then be subjected, according to the probability assigned, to several processes inspired by natural reproductive processes.

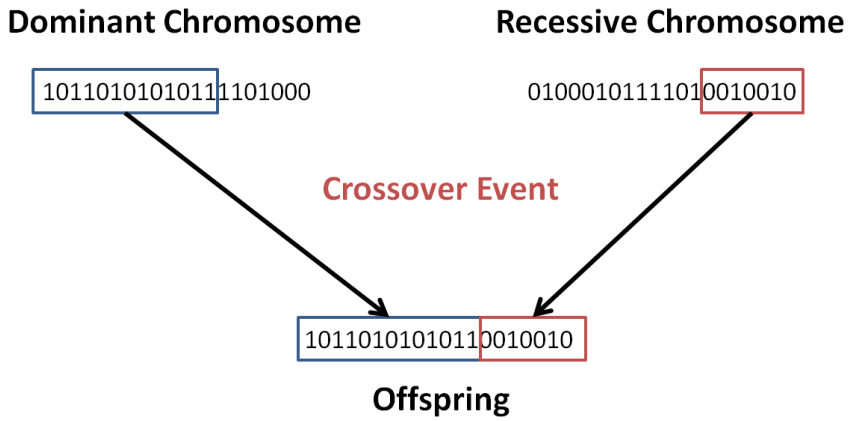


Figure 33: Crossover

Two copies of a chromosome may exchange genetic material as they may have different variants of a gene at the same location. This is referred to as crossover. Crossing over adds to the potential variability of the sensor positions considered. In the example shown in Figure 33, one copy of the chromosome carries alleles that are dominant (mask the effect of recessives) and the other only recessive copies of genes. Crossing over is a random event, though genes that are located further apart from one another are more likely to end up in different offspring. If the pair is not selected for crossover, the dominant partners chromosome is directly copied to form the offspring. The resulting offspring is then subjected to the possibility of genetic mutation according to a user specified probability. Genetic mutation occurs on the gene level flipping random genes in the chromosome, the likelihood of which

is weighted so that less significant genes (neutral to selection) are more likely candidates for mutation than very significant genes (subject to natural selection) as described by Figure 34.

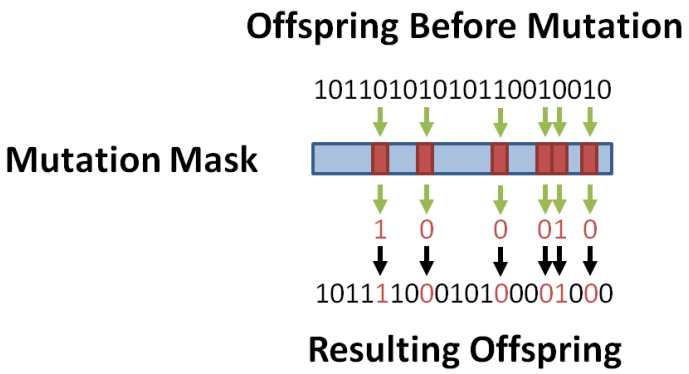


Figure 34: Mutation

After the reproduction events are complete, from which the new population is generated, the process begins again and continues to repeat until either a desired number of generations are produced or a stop condition has been reached. For example, over a significant number of generations the optimal solution variant remains unchanged.

Table 3: FLASHMan Simulation Parameters

Parameter	Value
Link Length	0.42m
Cart Mass	10kg
Payload Mass	0.291kg
Cart Damping	100Ns/m
Structural Damping Factor	0.0005
Young's Modulus	$7E10$Pa
Density	2700kg/m^3
Area Moment	$1.01E-10$m^4
Crossectional Area	$1.21E-4(m^2)$

The parameters of the FLASHMan testbed in Figure 10 are listed in Table 3. For this system a genetic algorithm as described in the preceding section was used to determine the optimal sensor placements for both a single sensor and for multiple sensors. Accelerometers were chosen to be the candidate sensors for this analysis and the parameters of the genetic

algorithm are listed in Table 4.

Table 4: Genetic Algorithm Settings

Parameter	Value
Population	20
# of Generations	100
# of Elites	2
Probability of Crossover	25%
Probability of Mutation	80%

The final population and chosen solution for the different placement metrics (observation quality, SV of W_o for $z_1 - z_5$) are shown in Figures 35 and 36(a)-36(f).

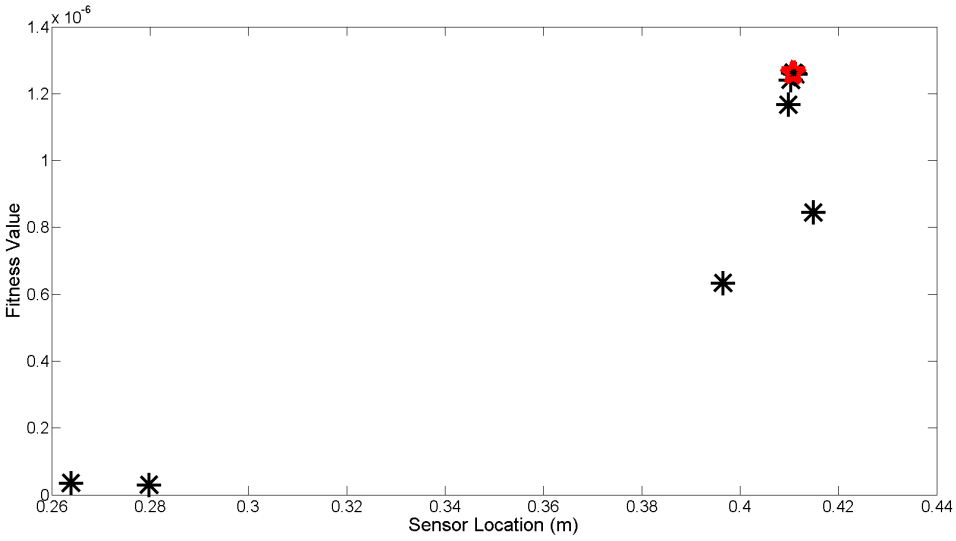
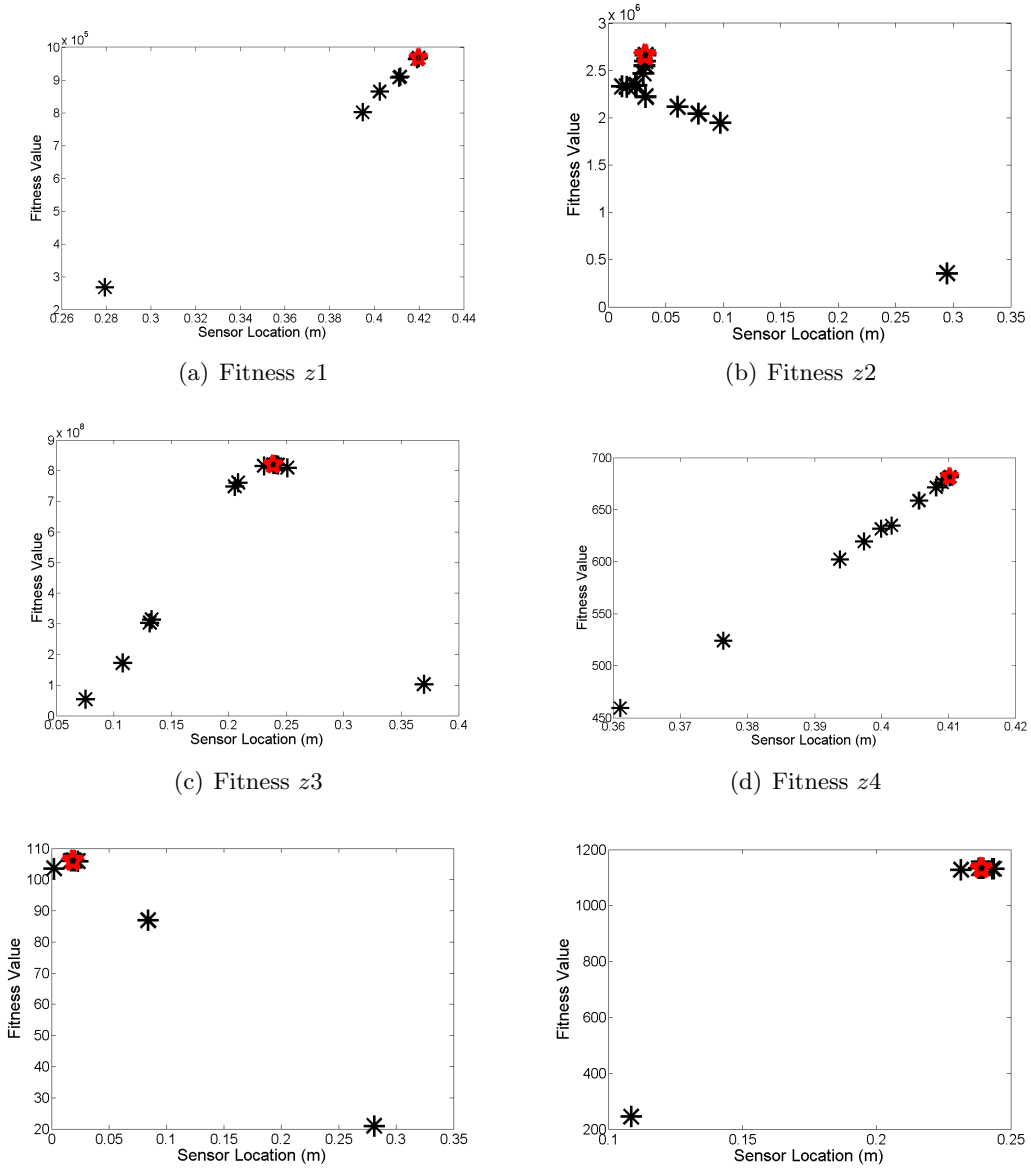


Figure 35: Genetic Algorithm Results Observation Quality Fitness Metric

Using the sensor quality metric, invariance to parametric error can be improved through the design of fitness function. For example, by selecting multiple models over the expected level of parametric variation, and taking the minimum sensor quality of the set for each potential sensor location, a search over the length of the arm for the maximum value of the fitness function results in a placement which will avoid potential nodal points.



(a) Fitness $z1$

(b) Fitness $z2$

(c) Fitness $z3$

(d) Fitness $z4$

(e) Fitness $z5$

(f) Fitness $z6$

Figure 36: Genetic Algorithm Results (Final Population and Optimal Placement)

The placements determined by the algorithm are listed in table 5.

Table 5: Chosen Sensor Locations

Fitness Function	Sensor Location
Q	0.4122m
σ_1	0.4198m
σ_2	0.0000m
σ_3	0.2387m
σ_4	0.4198m
σ_5	0.0098m
σ_6	0.2388m

Placing a single sensor using Q as the fitness function results in an optimal placement near the tip of the beam at 0.4122m. Using multiple sensors, and selecting placements based on the maximum singular values of each modal state, results in sensor locations of 0.420, 0.239, and 0.005m from the root of the beam. While rather trivial for this simple structure, the computational benefits are amplified when considering more complex structures.

4.6 Observability Robustness Evaluation

While optimal for a perfect system model, these sensor placements may or may not be robust to parametric error. Therefore in order to quantify the robustness of a given sensor placement to changes in a specific system parameter, the singular values must be inspected over the range of the expected parametric error. Parametric error can result in two distinct errors in the resultant state estimates. First given a limited set of sensors, parametric error may result in unobservability of a subset of system states. And second, the gains calculated using the sensor placement can result in static state error.

Special care must be taken to ensure the chosen sensor locations are robust to unobservability caused by potential parametric errors. This can be accomplished by redefining the fitness function used in the genetic algorithm. For example if sensor quality is used as the metric for the placement of a single sensor as in Figure 27, the optimal placement lies near the end of the beam where the condition number of the observability gramian is its minimum. If however, the payload changes from its nominal value, the singular values of

the observability gramian will have a corresponding shift. This shift is illustrated in Figure 37 for a true payload mass of 0.425kg (orange line) and a modeled mass of 0.281kg (black line).

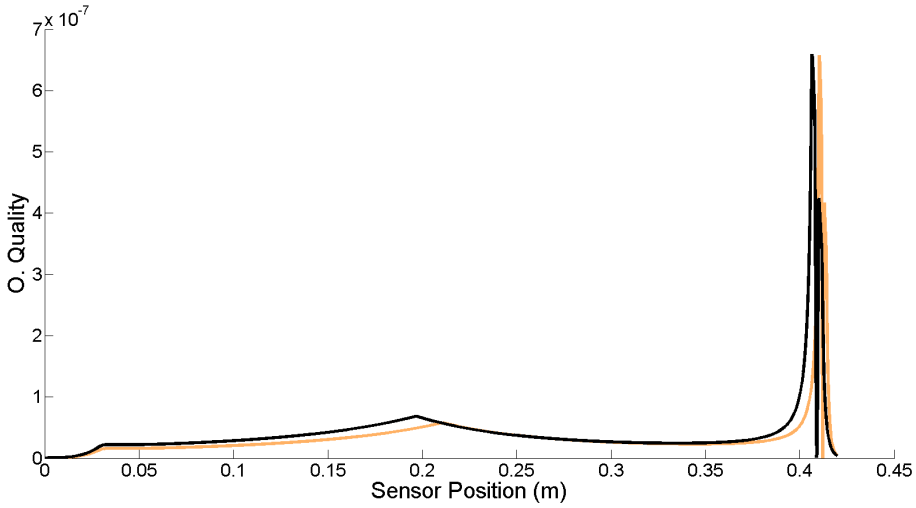


Figure 37: Observation Quality vs. Payload

From the analysis in Section 4.5, an accelerometer placement at x_s = 0.412m was determined to be optimal. From Figure 38 the increase in payload moves the nodal point associated with the second beam mode to the position of the sensor placement. At the nodal point, the system becomes unobservable.

However, by prescribing the placement of multiple sensors, one for each mode of vibration, robust observability is assured for a wide range of parametric variation. While a single sensor may experience a modal occlusion, the sensor corresponding to the occluded mode will not be obstructed and, thus, maintain system observability (a result which is discussed in detail in Section 5.1.1).

Modal occlusions are not always detrimental. They may be harnessed to combat residual mode spillover. If the sensors are located in the nodes of the residual mode, the state estimates will be an accurate description of the controllable modes and remain uncorrupted by spillover from the unmodeled residual modes. If residual modes are problematic, the described sensor placement analysis can be used to place these sensors by modifying the fitness function of the genetic algorithm to determine the nodal points of the residual modes

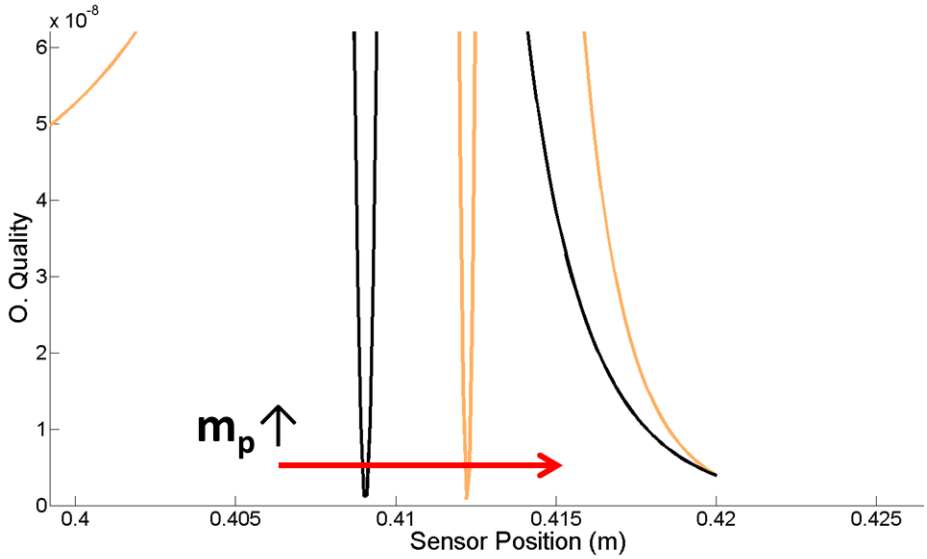


Figure 38: Observation Quality vs. Payload (Zoom)

instead of the modal maxima.

4.7 Connection Between Observability and Observer Design

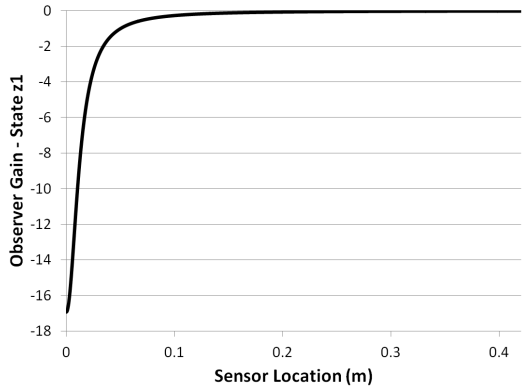
More than a simple test for determining if a state observer is feasible for a given sensor choice, the observability gramian and observability test matrix are integrated in the determination of observer gains as well. For example, Ackerman's formula (165) for pole placement relies on a direct inversion of the observability test matrix to determine the observer gain.

$$L = \alpha(A) \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \tag{165}$$

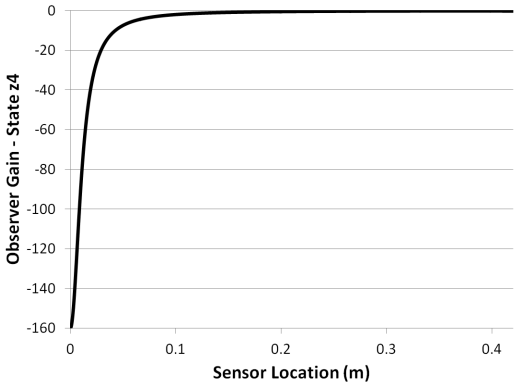
where $\alpha(A)$ is the desired characteristic equation as a function of the state matrix A, and

$$\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

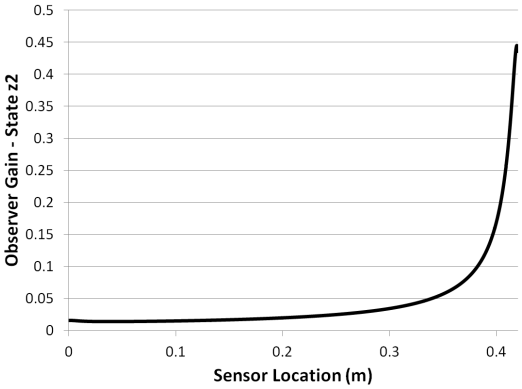
is the observability test matrix O. The observability test matrix is easily derived, as in (150) where $p = n-1$ and n is the total number of system states. If instead of only considering $n-1$ an infinite number of measurements are considered, O^TO becomes the observability gramian. Therefore, many of the results from previous sections apply to the observability test matrix as well as the the observability gramian. The inversion of O in (165) determines the sensitivity of the gain selection to the measurement. This allocates the sensor's energy among the individual states. Using Ackerman's formula, the gains associated with each state versus the sensor location are shown in Figures 39(a) through 39(f) (for desired pole locations of $-100, -110, -120, -130, -140, -150$ on the real axis).



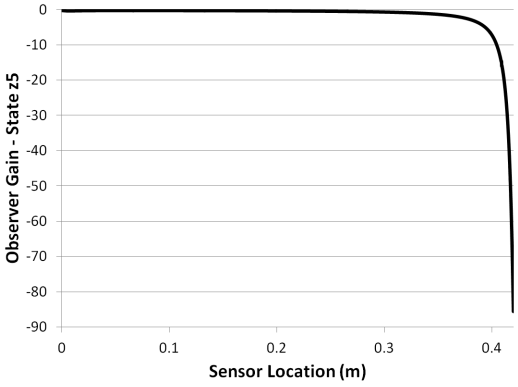
(a) Gain assigned to $z1$



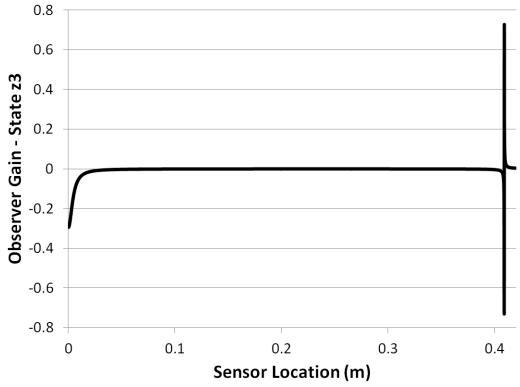
(b) Gain assigned to $z4$



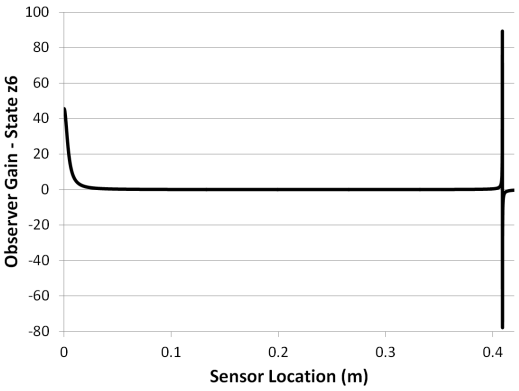
(c) Gain assigned to $z2$



(d) Gain assigned to $z5$



(e) Gain assigned to $z3$



(f) Gain assigned to $z6$

Figure 39: Observer Gain (L) Associated with Each State vs. Sensor Position

Comparing the results of Figure 39 with those of Figure 25, it is apparent that large singular values equate with smaller gain values. As expected, this simply means that less gain is necessary because the measurement energy is well correlated to the state. Very little energy is associated for small singular values, resulting in a large amplification. Amplification adds noise and diminishes robustness with respect to parametric error.

Discussed in detail in Chapter 5 the Kalman filter is an optimal state estimator which places the system poles by minimizing a quadratic cost function in order to balance convergence rate and measurement noise. Assuming the system is both discrete and autonomous for this discussion, the estimated state vector at the time step k is,

$$\hat{z}_k = A_d \hat{z}_{k-1} + L(y_{k-1} - C_d \hat{z}_{k-1}) \tag{166}$$

The Kalman gain, as described in the literature [92] is

$$L = P_k C_d^T R^{-1} \tag{167}$$

where the state covariance error after measurement is

$$P_k = \left(M_k^{-1} + C_d^T R^{-1} C_d\right)^{-1} \tag{168}$$

which is dependent on the covariance generated between measurements:

$$M_k = A_d P_{k-1} A_d^T + G Q G^T \tag{169}$$

Combining (168) and (169) and working backwards in time with the assumpitons of no process noise ($Q = 0$) and an infinite initial error covariance ($P_0^{-1} = 0$),

$$
\begin{aligned}
P_{k+1}^{-1} &= \left(A_d P_k A_d^T\right)^{-1} + C_d^T R^{-1} C_d \\
&= A_d^{-T} P_k^{-1} A_d^{-1} + C_d^T R^{-1} C_d \\
&= A_d^{-T}\left(A_d^{-T} P_{k-1}^{-1} A_d^{-1} + C_d^T R^{-1} C_d^{-1}\right) A_d^{-1} + C_d^T R^{-1} C_d \\
&= A_d^{-2T} P_{k-1} A_d^{-2} + A_d^{-T} C_d^T R^{-1} C_d A_d^{-1} + C_d^T R^{-1} C_d \\
&= A_d^{-2T}\left(A_d^{-T} P_{k-2}^{-1} A_d^{-1} + C_d^T R^{-1} C_d^{-1}\right) A_d^{-2} + A_d^{-T} C_d^T R^{-1} C_d A_d^{-1} \\
&= A_d^{-3T} P_{k-2} A_d^{-3} + A_d^{-2T} C_d^T R^{-1} C_d A_d^{-2} + A_d^{-T} C_d^T R^{-1} C_d A_d^{-1} + C_d^T R^{-1} C_d \\
&\vdots \\
&= A_d^{-kT} \cancelto{0}{P_0} A_d^{-k} + \sum_{i=0}^{k} A_d^{-i} C_d^T R^{-1} C_d A_d^{-i} \\
&= \sum_{i=0}^{k} A_d^{-i} C_d^T R^{-1} C_d A_d^{-i}
\end{aligned} \tag{170}
$$

Note the similarity to the discrete observability gramian (153). The result in (170) is called the stochastic observability gramian, and it includes the effects of measurement noise covariance. The Kalman gain is, therefore:

$$
L = \left(\sum_{i=0}^{k} A_d^{-i} C_d^T R^{-1} C_d A_d^{-i}\right)^{-1} C_d^T R^{-1} = W_{o_s}^{-1} C_d^T R^{-1} \tag{171}
$$

under the given assumptions and is therefore dependent on the inverse of the stochastic observability gramian for pole placement. For single sensor placements, the results of the singular value analysis remain unchanged, and for multiple measurements use of the stochastic observability gramian has the added benefit of weighting the singular values by the magnitudes of the sensor noise.

Chapter V

ROBUST STATE ESTIMATION FOR FLEXIBLE MANIPULATORS

Robust state estimation, as defined in this work, means that the estimator performance (i.e. the convergence rate and accuracy of the estimated state to the true state) is preserved under conditions of parametric uncertainty. In the following analysis, the fundamental properties of state estimators subjected to parametric error are derived and used to compare the robustness of the evaluated observers. While the analysis is aimed at flexible manipulators, simplified second order mass-spring-damper examples are used to demonstrate the derived results. These second order examples allow effective visualization of the dynamic behavior, whereas, the sixth order FLASHMan system would require more dimensions than are available in this format to achieve the same result. Simulation and experimental results are presented, that: 1) validate this analysis; and 2) compare traditional linear estimators with a purportedly robust non-linear sliding mode observer.

5.1 Linear Estimation for State Reconstruction

The goal of a state estimator is to reconstruct the internal state of a dynamic system from measurable external observations. From these measurements an initial condition is inferred which would produce, given a suitable model of the system, the set of system states which correspond to the measured output. Given that these conditions are met, the state estimate will match the true state.

The restrictions on measurements are that they are sufficiently coupled to the internal state, so that all states can be inferred from the measurement(s). By evaluating the rank of the matrix O (166), the observability of the internal system states from a given measurement can be determined. If (166) is full rank, then the system is said to be "completely observable," meaning all the states can be reconstructed. If O is rank deficient, (i.e. $rank(O) = p < n$), then there exist $n - p$ unobservable states, and additional measurements may be necessary to fully observe the state of the system. Thus observability provides a

litmus test for whether a sensor, or set of sensors, sufficiently characterizes the state of the system. While the preceding observability discussion was tailored for determining optimal sensor placements, the following brief analysis is aimed at justifying the choice of feedback control and estimation structure used in this work.

As a motivating example, consider the simple cart system in Figure 40.

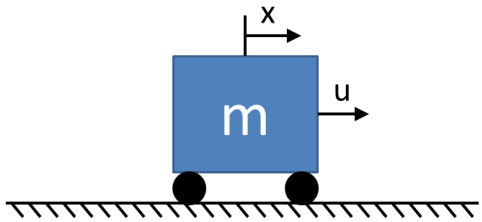


Figure 40: Unconstrained 1-DOF Mass

A state space representation of the system is

$$\begin{bmatrix} \dot{z_1} \\ \dot{z_2} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{A} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}}_{B} u \qquad where: \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \tag{172}$$

If the position of the mass is selected as the measurement, the output y can be specified as

$$y = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \end{bmatrix}}_{D} u \tag{173}$$

Therefore the observability test matrix is:

$$O = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{174}$$

Obviously O is full rank. Therefore, the position and velocity states are observable from a position measurement. Conversely, if velocity is chosen as the measurement, $C = \begin{bmatrix} 0 & 1 \end{bmatrix}$ and therefore:

$$O = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{175}$$

Which is rank 1. This means that the state corresponding to position is unobservable from a velocity measurement. The problem lies in the free integrator, which makes the initial condition on position indeterminate.

If the model can be expressed as a constrained system, by for example by making the input a displacement of a spring attached to the block as shown in Figure 41, then the problem is resolved and derivative measurements like acceleration are sufficient to reconstruct the full state of the system.

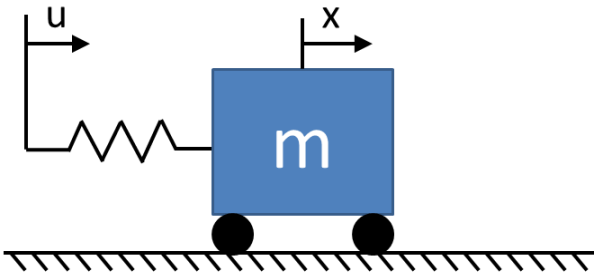


Figure 41: Constrained 1-DOF Mass

$$\begin{bmatrix} \dot{z_1} \\ \dot{z_2} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & 0 \end{bmatrix}}_{A} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ \frac{k}{m} \end{bmatrix}}_{B} u \tag{176}$$

and given the dynamic equation of motion $\ddot{x} = -\frac{k}{m}x + \frac{k}{m}u$, the output equation for accelerometer measurement becomes:

$$y = \underbrace{\begin{bmatrix} -\frac{k}{m} & 0 \end{bmatrix}}_{C} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{k}{m} \end{bmatrix}}_{D} u \tag{177}$$

Therefore:

$$O = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} -\frac{k}{m} & 0 \\ 0 & -\frac{k}{m} \end{bmatrix} \tag{178}$$

Because (178) is full rank, acceleration measurements for this system are sufficient to reconstruct the full state without an additional position measurement. For flexible manipulators this is beneficial as it is often difficult to prescribe, in an open loop manner, the amount of force or torque on a link from a joint with flexibility. Conversely, it is often relatively easy to close the loop on the displacement of the motor which commands the joint. This prescribes, as described in Figure 42, the displacement of a spring attached to the link.

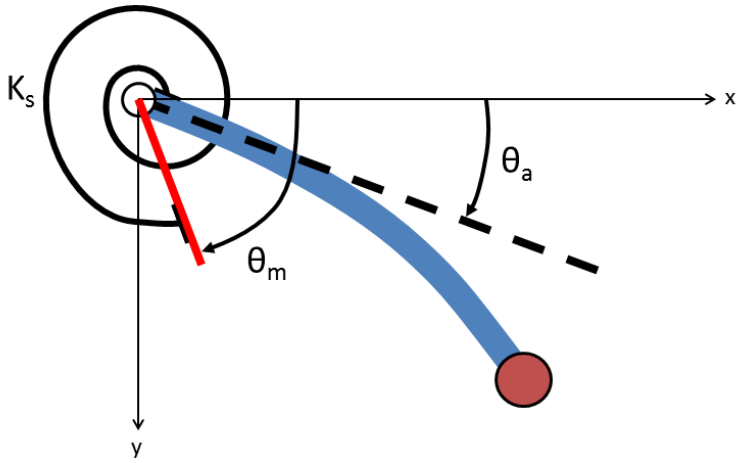


Figure 42: Flexible Arm and Joint Coupling

Because this input constraint removes the rigid body displacement information from the scope of the estimator and pushes it into the control domain, it allows acceleration measurements to be used as the sole source of state information and allows the input to the estimator to be a signal which is well known, i.e. the displacement of the drive motor.

Given the choice of model structure as illustrated in Figure 42, a complete state space control system can be constructed as described in Figure 43. A tight position control loop is placed around the actuator position which serves as the input to the flexible system. When using an encoder to measure this position, one applies a well known measured quantity rather than an approximate force as the input to the estimator. A state feedback controller

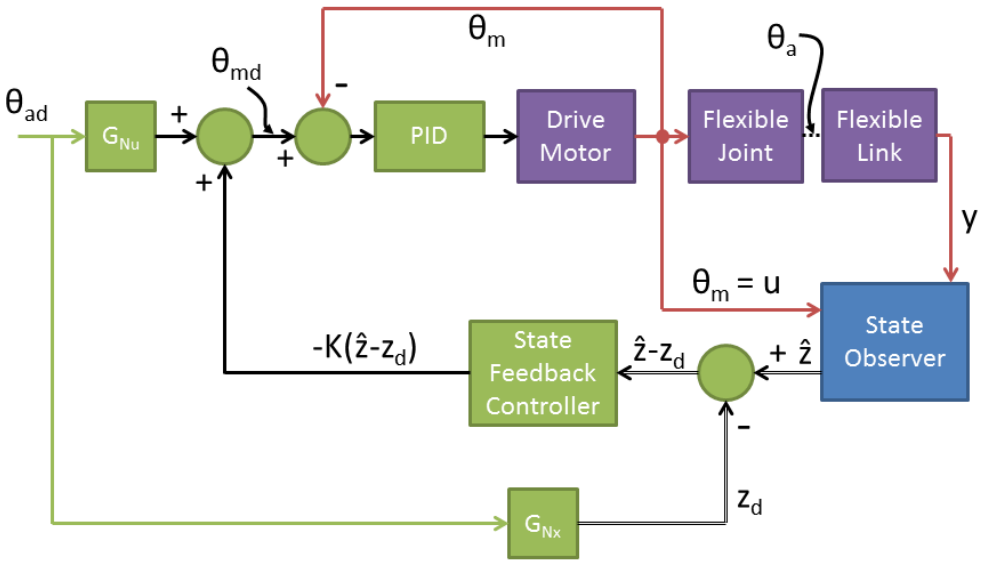


Figure 43: Control System Block Diagram

is used to drive the state to the desired state by perturbing the position command to the motor.

5.1.1 Observability Conditions for Accelerometer Based Flexible Manipulator Estimation Algorithms

For a single link manipulator with a belt drive the dynamic model can be derived, as in Chapter 3. The result of this analysis is reproduced below in the modal form considering three modes of vibration (two predominantly associated with the flexible beam modes, and one with the cart/belt mode) and no damping.

$$\underbrace{\begin{bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \\ \dot{\eta}_3 \\ \ddot{\eta}_1 \\ \ddot{\eta}_2 \\ \ddot{\eta}_3 \end{bmatrix}}_{\dot{z}} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -\omega_1^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\omega_2^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega_3^2 & 0 & 0 & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \dot{\eta}_1 \\ \dot{\eta}_2 \\ \dot{\eta}_3 \end{bmatrix}}_{z} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ \phi_{1,1} K_{belt} \\ \phi_{1,2} K_{belt} \\ \phi_{1,3} K_{belt} \end{bmatrix}}_{B} \underbrace{x_{belt}}_{u} \tag{179}$$

and the acceleration of a point along the length of the arm at position y is described by:

$$\ddot{x}_p(t) = \begin{bmatrix} -\phi_{1,1}\omega_1^2 - \psi_1(y)\phi_{2,1}\omega_1^2 - \psi_2(y)\phi_{3,1}\omega_1^2 \\ -\phi_{1,2}\omega_2^2 - \psi_1(y)\phi_{2,2}\omega_2^2 - \psi_2(y)\phi_{3,2}\omega_2^2 \\ -\phi_{1,3}\omega_3^2 - \psi_1(y)\phi_{2,3}\omega_3^2 - \psi_2(y)\phi_{3,3}\omega_3^2 \\ 0 \\ 0 \\ 0 \end{bmatrix}^T z + [d]\, u \tag{180}$$

where

$$\begin{aligned} d =& \phi_{1,1}^2 K_{belt} + \phi_{1,2}^2 K_{belt} + \phi_{1,3}^2 K_{belt} + \\ & \psi_1(y)(\phi_{2,1}\phi_{1,1}K_{belt} + \phi_{2,2}\phi_{1,2}K_{belt} + \phi_{2,3}\phi_{1,3}K_{belt}) + \\ & \psi_2(y)(\phi_{3,1}\phi_{1,1}K_{belt} + \phi_{3,2}\phi_{1,2}K_{belt} + \phi_{3,3}\phi_{1,3}K_{belt}) \end{aligned} \tag{181}$$

Or written more concisely:

$$\ddot{x}_p(t) = \underbrace{\begin{bmatrix} c_1\omega_1^2 & c_2\omega_2^2 & c_3\omega_3^2 & 0 & 0 & 0 \end{bmatrix}}_{C} z + \underbrace{[d]}_{D} u \tag{182}$$

Therefore, the observability can be checked using (166) to determine if all states are observable.

$$O = \begin{bmatrix} c_1\omega_1^2 & c_2\omega_2^2 & c_3\omega_3^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_1\omega_1^2 & c_2\omega_2^2 & c_3\omega_3^2 \\ c_1\omega_1^4 & c_2\omega_2^4 & c_3\omega_3^4 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_1\omega_1^4 & c_2\omega_2^4 & c_3\omega_3^4 \\ c_1\omega_1^6 & c_2\omega_2^6 & c_3\omega_3^6 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_1\omega_1^6 & c_2\omega_2^6 & c_3\omega_3^6 \end{bmatrix} \tag{183}$$

From (183) it is obvious that the system is unobservable if it exhibits a rigid body mode(s), i.e. ω_1, ω_2, or $\omega_3 = 0$. Less obviously, the accelerometer cannot be located at a

nodal point.

A mode shape exhibits nodes at locations where it crosses the neutral axis of the beam at rest. At these points, if one were to excite the flexible structure at the appropriate frequency ω_i, the displacement of the beam element would be zero for all time. Intuitively, this means that for a sensor located at the nodal point the corresponding mode has no contribution to the measured acceleration. Therefore, the full state of the system cannot be reconstructed from that specific sensor measurement (as previously described).

To demonstrate this fact consider the displacement of a beam element of the system described by (179) at location y_p:

$$x(t) = \begin{bmatrix} 1 & \psi_1(y_p) & \psi_2(y_p) \end{bmatrix} \begin{bmatrix} \phi_{1,1} & \phi_{1,2} & \phi_{1,3} \\ \phi_{2,1} & \phi_{2,2} & \phi_{2,3} \\ \phi_{3,1} & \phi_{3,2} & \phi_{3,3} \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \end{bmatrix} \tag{184}$$

If excited at a natural frequency, all energy is contained within that mode of vibration and therefore all other modal coordinates are zero. Thus the ith mode shape of the flexible system can be determined by setting all the other modal coordinates ($\eta_{j \neq i}$) to zero in (184).

$$x_{mode_1}(t) = (\phi_{1,1} + \psi_1(_p)\phi_{2,1} + \psi_2(_p)\phi_{3,1})\,\eta_1 \tag{185}$$

At a node of the mode shape, y_{node}:

$$0 = \underbrace{(\phi_{1,1} + \psi_1(y_{node})\phi_{2,1} + \psi_2(y_{node})\phi_{3,1})}_{-c_1}\,\eta_1 \tag{186}$$

Because $c_1 = 0$, (183) is singular when the sensor is placed at a nodal point, and the system is not completely observable. At a node, system poles corresponding to the mode of vibration attributed to the node, are obscured by zeros. This reduces the residues of that pole in the sensor signal preventing the observation of the states corresponding to the specific mode of vibration. This effect is illustrated in Figure 44, if the sensors are placed in the nodal points indicated, then the resulting zeros cover the poles corresponding to the obscured mode.

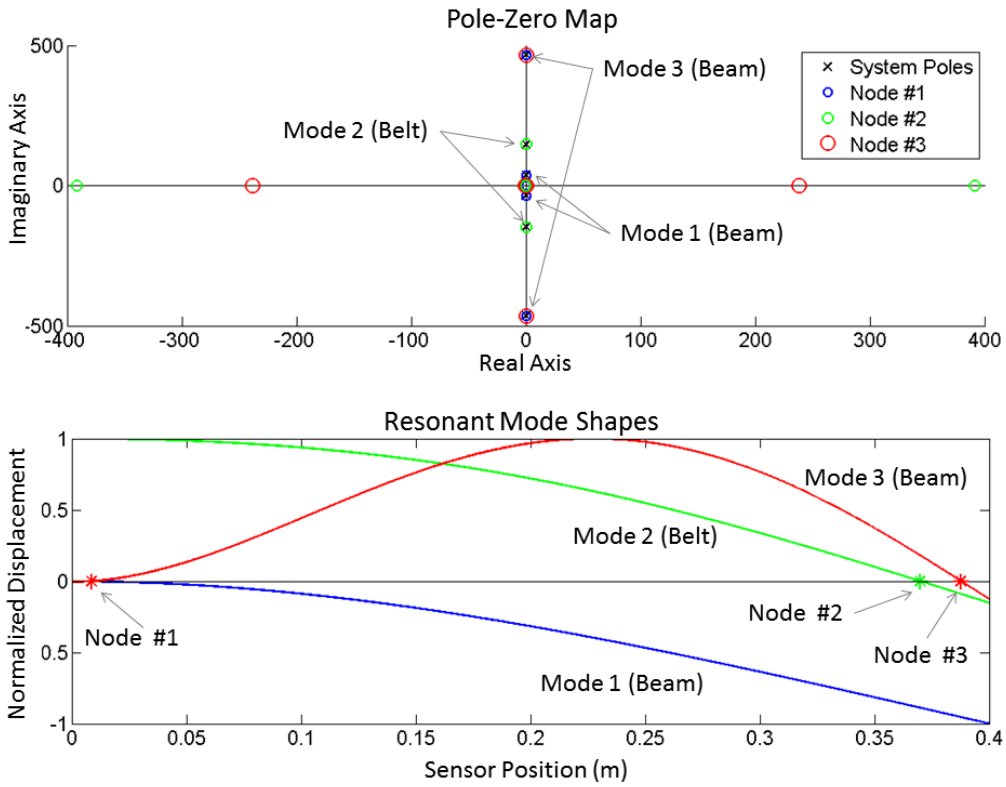


Figure 44: Pole-Zero Map of Sensors in Nodal Points

5.1.2 Linear Estimation of Systems with Parametric Error

Assuming the appropriate choice of sensor and sensor placement, linear estimation techniques can be applied to estimate the state of single link or linearized multi-link flexible manipulators. The goal of a state estimator is to force the error between the estimated states and the true system states to converge to zero by matching the predicted outputs and the measured outputs. This presents a problem which is complicated by modeling error and uncertainty. Modeling error can be both structured and unstructured, in that the form of the model is either of known or unknown order.

For flexible link manipulators it is assumed that the dynamic system can be truncated to a finite number of modes, and therefore any relevant uncertainty in the model can be described in terms of the system parameters of the reduced order model. This structured uncertainty corrupts both the state and measurement matrices in the system representation.

These effects are most easily described through a simple example. A mass spring system was illustrated in Figure 41. The system dynamics for the system, with damping, can be

expressed as:

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k}{m} \end{bmatrix} u \tag{187}$$

For the sake of argument assume that the position of the mass is directly measured for use in the estimator. Therefore the output y can be described by

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} z + [0]\, u \tag{188}$$

and is simply a state of the system. Conversely if acceleration of the mass ($\ddot{x}$) were chosen, the corresponding output formulation would be:

$$y = \begin{bmatrix} \frac{-k}{m} & \frac{-c}{m} \end{bmatrix} z + \begin{bmatrix} \frac{k}{m} \end{bmatrix} u \tag{189}$$

Immediately obvious from the above formulation is that for a given parametric error (e.g. spring stiffness, damping coefficient, or mass) the relationship that connects the measured output to the system states remains the same for the case when y is a direct state measurement (188). However, it will change if the measurement is linked to the specific parameters of the system as it is in (189).

To determine how parametric error influences the estimation of system states let a generic state space system be represented as:

$$\dot{z} = Az + Bu \tag{190}$$

$$y = Cz + Du \tag{191}$$

Allowing structured uncertainty in the model through parametric error, this model becomes:

$$\dot{z} = (A + \Delta A)\, z + (B + \Delta B)\, u \tag{192}$$

$$y = (C + \Delta C)\, z + (D + \Delta D)\, u \tag{193}$$

where ΔA, ΔB, ΔC, and ΔD are unknown but bounded matrices and A, B, C, and D are the approximate state space matrices on which the observer and controller are based. A current state observer (Figure 45) uses the system model to predict the current state of the system. By comparing the predicted output and measured output, a gain L is used to force convergence between the predicted and measured outputs, thus causing the estimated states to match the true system states.

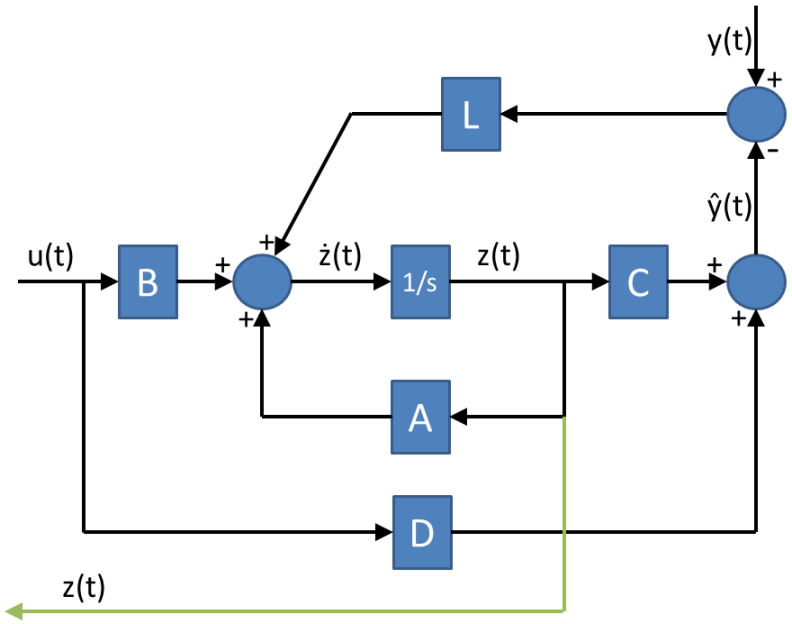


Figure 45: Current State Observer

However this logic is only valid with the assumption that there is no uncertainty in the system model. For the case of an ideal estimator the estimator dynamics are:

$$\dot{\hat{z}} = A\hat{z} + Bu + L(y - \hat{y}) \tag{194}$$

$$y = Cz + Du \tag{195}$$

$$\hat{y} = C\hat{z} + Du \tag{196}$$

which gives the state estimate dynamics:

$$\dot{\hat{z}} = A\hat{z} + Bu + L(Cz + \cancel{Du} - C\hat{z} - \cancel{Du}) \tag{197}$$

$$\Rightarrow \dot{\hat{z}} = A\hat{z} + Bu + LC(z - \hat{z}) \tag{198}$$

defining the estimate error as:

$$e = z - \hat{z} \tag{199}$$

$$\Rightarrow \dot{e} = \dot{z} - \dot{\hat{z}} \tag{200}$$

Thus the estimate error dynamics are:

$$\dot{e} = \cancel{Az - A\hat{z}}^{Ae} + \cancel{Bu} - \cancel{Bu} + \cancel{LC(z - \hat{z})}^{LCe} \tag{201}$$

$$\Rightarrow \dot{e} = (A - LC)e \tag{202}$$

Thus one can simply select L such that all the poles of $A - LC$ are in the left half plane (LHP) forcing the estimated state to converge to the true state as the state error converges. Consider, however, the case of modeling uncertainty under which the estimator model and physical systems differ by the quantities ΔA, ΔB, ΔC and ΔD. The new estimator dynamics are as follows:

$$\dot{\hat{z}} = A\hat{z} + Bu + L(y - \hat{y}) \tag{203}$$

$$y = (C + \Delta C)z + (D + \Delta D)u \tag{204}$$

$$\hat{y} = C\hat{z} + Du \tag{205}$$

Thus,

$$\dot{\hat{z}} = A\hat{z} + Bu + L((C + \Delta C)z + (D + \Delta D)u - C\hat{z} - Du) \tag{206}$$

$$\Rightarrow \dot{\hat{z}} = A\hat{z} + Bu + L(Cz - C\hat{z} + \cancel{Du} - \cancel{Du} + \Delta Cz + \Delta Du) \tag{207}$$

$$\Rightarrow \dot{\hat{z}} = A\hat{z} + Bu + LC(z - \hat{z}) + L\Delta Cz + L\Delta Du \tag{208}$$

$$\tag{209}$$

Again let $e = z - \hat{z}$ where z is as designated by (192), $\dot{z} = (A + \Delta A)\, z + (B + \Delta B)\, u$. Then

$$\dot{e} = \cancel{Az - A\hat{z}}^{Ae} + \Delta Az + \cancel{Bu - Bu} + \Delta Bu - \cancel{LC(z - \hat{z})}^{LCe} - L\Delta Cz - L\Delta Du \tag{210}$$

$$\Rightarrow \dot{e} = (A - LC)e + (\Delta A - L\Delta C)z + (\Delta B - L\Delta D)u \tag{211}$$

Let,

$$A_0(n \times n) = (A - LC) \tag{212}$$

$$W_{(n \times n)} = (\Delta A - L\Delta C) \tag{213}$$

$$J_{(n \times p)} = (\Delta B - L\Delta D) \tag{214}$$

where n is the number of model states, p is the number of inputs to the system, and W and J are unknown matrices. Then the state estimate error dynamics are:

$$\dot{e} = A_0 e + Wz + Ju \tag{215}$$

Obviously (215) contains the original error dynamics from (202) plus a term dependent

on the true system state and a term dependent on the controlled input. When a state feedback control of the form $u = -K\hat{z}$ is used then $Ju \to -JK\hat{z}$ and the error dynamics are dependent not only on the current system state but the estimated state as well. Both serve as exogenous inputs to the error system which will drive the equilibrium point of the error system away from the origin of the state space and produce state estimate error.

5.1.3 Important Repercussions of Parametric Modeling Error on Closed Loop Perfomance

While appropriate estimator behavior is important for accurate state estimates, it is imperative to note the repercussions of modeling error on closed loop system performance. Specifically, the destabilizing potential of controllers which act on model-corrupted estimated states should be carefully considered in selecting an appropriate feedback control system. While the convergence properties of the estimator and control system are unchanged when the estimated state remains unutilized in the control system, the changing equilibrium points of the estimator dynamics invalidate the stated "separation principle," effectively coupling the controller and estimator poles.

To demonstrate this effect consider the closed loop system dynamics

$$\dot{z} = (A + \Delta A)\, z - (B + \Delta B)\, K\hat{z} \tag{216}$$

And replacing $\hat{z}$ with $z - e$,

$$\dot{z} = (A + \Delta A - BK - \Delta BK)\, z + (BK + \Delta BK)\, e \tag{217}$$

Similarly the estimator dynamics are described by

$$\dot{\hat{z}} = A\hat{z} - BK\hat{z} + L\left[(C + \Delta C)\, z - (D + \Delta D)\, K\hat{z} - C\hat{z} + DK\hat{z}\right] \tag{218}$$

or more simply

$$\dot{\hat{z}} = A\hat{z} - BK\hat{z} + LCe + L\Delta Cz - L\Delta DK\hat{z} \tag{219}$$

The estimator dynamics are therefore,

$$\dot{e} = (A - LC - L\Delta DK + \Delta BK)\, e + (\Delta A - \Delta BK - L\Delta C + L\Delta DK)\, z \tag{220}$$

Combining the estimator (220) and controlled system dynamics (217) results in the full system dynamics

$$\begin{bmatrix} \dot{z} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A + \Delta A - BK - \Delta BK & BK + \Delta BK \\ \Delta A - \Delta BK - L\Delta C + L\Delta DK & A - LC - L\Delta DK + \Delta BK \end{bmatrix} \begin{bmatrix} z \\ e \end{bmatrix} \tag{221}$$

The eigenvalues of the full system block matrix A_{full} determines the location of the controller and estimator poles. If the modeling error is neglected, the system becomes an upper triangular block matrix

$$\begin{bmatrix} \dot{z} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} z \\ e \end{bmatrix} \tag{222}$$

for which the determinant is $det(A_{full}) = det(A - BK)det(A - LC)$. Therefore, the system poles are simply the combined estimator and controller poles.

However with parametric modeling error, the determinant is no longer separable into terms associated with only the control or estimation subsystems. For example, consider the aforementioned mass spring damper system with accelerometer feedback and force input and nominal parameters $m = 1$, $k = 1$ and $c = 0$.

$$\begin{bmatrix} \dot{z_1} \\ \dot{z_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{223}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} z + [1]\, u \tag{224}$$

The controller poles are placed at $p_c = -1 + i, -1 - i$ and the observer poles are placed at $p_0 = -5 + i, -5 - i$. Nominally, the full system matrix is

$$A_{full} = \begin{bmatrix} 0 & 1.00 & 0 & 0 \\ -2.00 & -2.00 & 1.00 & 2.00 \\ 0 & 0 & 7.50 & 8.50 \\ 0 & 0 & -18.50 & -17.50 \end{bmatrix} \tag{225}$$

, the poles of which are the controller and observer poles combined:

$$\begin{gathered} -1.00 + 1.00i \\ -1.00 - 1.00i \\ -5.00 + 1.00i \\ -5.00 - 1.00i \end{gathered} \tag{226}$$

If the mass was over estimated in the model by 20%, (i.e. the true mass is 0.8) the observer poles remain at:

$$\begin{gathered} -5.00 + 1.00i \\ -5.00 - 1.00i \end{gathered} \tag{227}$$

Assuming the system poles can be tuned to the desired locations p_c the true system matrix is:

$$A_{full} = \begin{bmatrix} 0 & 1.00 & 0 & 0 \\ -2.00 & -2.00 & 0.75 & 2.00 \\ 3.00 & 3.00 & 6.375 & 5.50 \\ -7.40 & -7.40 & -15.725 & -10.10 \end{bmatrix} \tag{228}$$

, which has poles at closed loop poles located at

$$\begin{gathered} -1.9259 + 5.6309i \\ -1.9259 - 5.6309i \\ -0.9366 + 0.7687i \\ -0.9366 - 0.7687i \end{gathered} \tag{229}$$

Thus, the resulting poles are not the combined controller and observer poles. Instead the closed loop performance is significantly affected by the inaccurate parameter estimate. Therefore, utilization of the separation principle allows only the approximation of the system pole locations for any real system. This approximation degrades as the level of error increases, meaning even if the observer and controller are designed correctly, a parameter shift or model inaccuracy can result in poor performance and potential destabilization.

The coupling between system state and estimate error can also result in steady state error when the desired final state is non zero. Because of the exogenous state error injection, the resulting state predictions from the estimator, in some systems, exhibit persistent prediction bias which results in set-point bias of the closed loop system. This prediction bias is further explored in the following illustrative example.

5.1.4 Illustrative Single Degree of Freedom Example

The following example is presented to further explore the effects of parametric error on estimation performance for traditional linear estimators. Again examine the mass spring damper system from Figure 41.

From a position measurement the the system is fully observable, and the output and feed through matrices become (230).

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad D = 0 \tag{230}$$

Assuming parametric error in the measurement of both the spring constant and damping coefficient of $k_m = k + \Delta k$ and $c_m = c + \Delta c$, then, the plant for which a model (187) was chosen is, in actuality,

$$\dot{z} = \left(\begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{-\Delta k}{m} & \frac{-\Delta c}{m} \end{bmatrix} \right) z + \left(\begin{bmatrix} 0 \\ \frac{k}{m} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{\Delta k}{m} \end{bmatrix} \right) u \tag{231}$$

where,

$$\Delta A = \begin{bmatrix} 0 & 0 \\ \frac{-\Delta k}{m} & \frac{-\Delta c}{m} \end{bmatrix} \quad \Delta B = \begin{bmatrix} 0 \\ \frac{\Delta k}{m} \end{bmatrix} \tag{232}$$

Let, $L = \begin{bmatrix} l_1 & l_2 \end{bmatrix}^T$ be the estimator gain determined from the inaccurate model of the system dynamics. Then the full dynamics of the error system become (233).

$$\dot{e} = \begin{bmatrix} -l_1 & 1 \\ -\frac{k}{m} - l_2 & -\frac{c}{m} \end{bmatrix} e + \begin{bmatrix} 0 & 0 \\ \frac{-\Delta k}{m} & \frac{-\Delta c}{m} \end{bmatrix} z + \begin{bmatrix} 0 \\ \frac{\Delta k}{m} \end{bmatrix} u \tag{233}$$

From (231) the equilibrium state $z_* = \begin{bmatrix} u_* & 0 \end{bmatrix}^T$ can be determined. This is the case because, the error in spring stiffness does not result in an error in resting length of the spring. Therefore, when the system reaches steady state the model and plant have identical equilibrium conditions. At equilibrium,

$$\begin{Bmatrix} 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} -l_1 & 1 \\ -\frac{k}{m} - l_2 & -\frac{c}{m} \end{bmatrix} \begin{Bmatrix} e_{1*} \\ e_{2*} \end{Bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{-\Delta k}{m} & \frac{-\Delta c}{m} \end{bmatrix} \begin{Bmatrix} u_* \\ 0 \end{Bmatrix} + \begin{bmatrix} 0 \\ \frac{\Delta k}{m} \end{bmatrix} u_* \tag{234}$$

indicating that even though transient state error will occur, as $t \to \infty$ the state error will converge to the origin of the error space for stationary inputs at the convergence rate of the controlled system.

Simulation results for the system using the parameters (exaggerated for demonstration) in table 7 are presented in Figure 46 and Figure 47. From Figure 46, note that initially the ideal estimate $\hat{z}_{1i}$ (based on a nominal plant model) and the true estimate $\hat{z}_1$ both converge to the true state z_1 and continue to track the true position of the mass as the step command u is given to the system, (indicated by the overlapping traces). The precise tracking is enabled by the fact that the position measurement is directly fed back as the output measurement.

Table 6: SMD Model Parameters

Parameter	True Value	Modeled Value	Unit
m	0.5	0.5	kg
k	100	150	N/m
c	1	50	Nm/s

Table 7: SMD Simulation Parameters

Parameter	Value	Unit
Command (u)	$\begin{cases} 0 & t < 1 \\ 10 & t \geq 1 \end{cases}$	m
Initial State (z_0)	$\begin{bmatrix} 0 & 0 \end{bmatrix}^T$	$\begin{bmatrix} \text{m} & \text{m/s} \end{bmatrix}^T$
Initial Estimate ($\hat{z}_0$)	$\begin{bmatrix} 2 & 10 \end{bmatrix}^T$	$\begin{bmatrix} \text{m} & \text{m/s} \end{bmatrix}^T$
Desired Est. Poles	$-100+0i \quad -101+0i$	
Est. Gain (L)	$\begin{bmatrix} 101 & -300 \end{bmatrix}^T$	

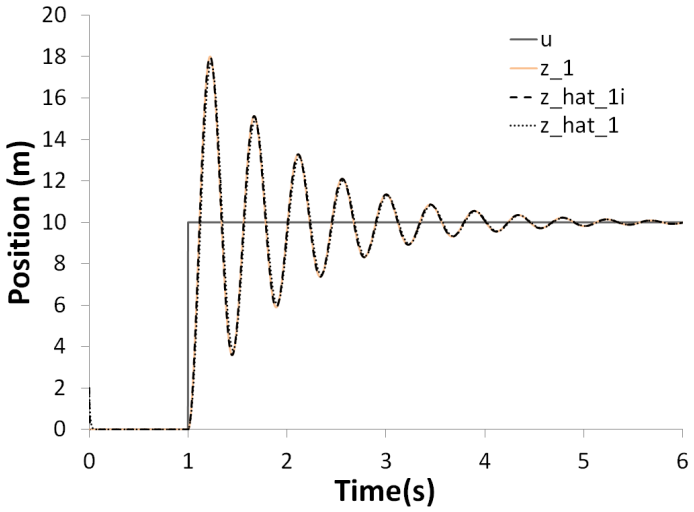


Figure 46: Simulated SMD Position Estimate (Position feedback)

The observer gain drives the predicted output $\hat{z}_1$ to match the measured output z_1. This is the equivalent of filtering a direct state measurement, and results in good coherence between the state prediction and true state.

However, from Figure 47 its obvious that after an initial convergence, the estimate of velocity $\hat{z}_2$ differs significantly from the true velocity state z_2 once the command is initiated forcing a transient condition where $z \neq z_*$. The estimate error is driven away from the origin and slowly converges again as $z \to z_*$ at the rate of damping in the plant as demonstrated in Figure 48. Conversely, the ideal estimator's velocity estimate (z_{2i}) is identical to the true velocity after the initial conditions are synchronized.

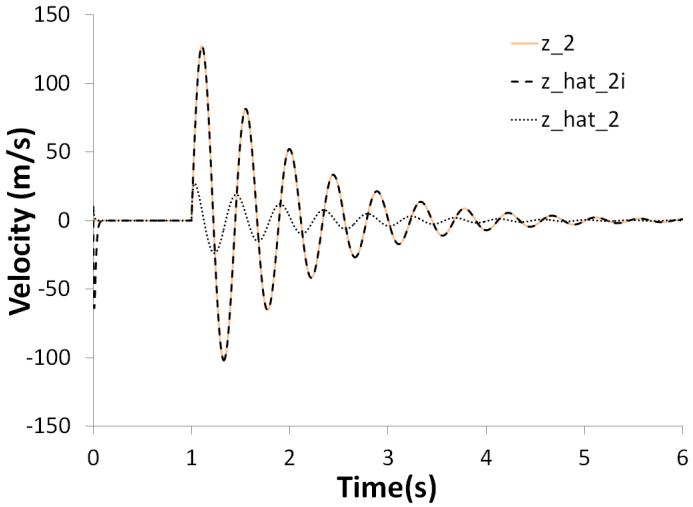


Figure 47: Simulated SMD Velocity Estimate (Position feedback)

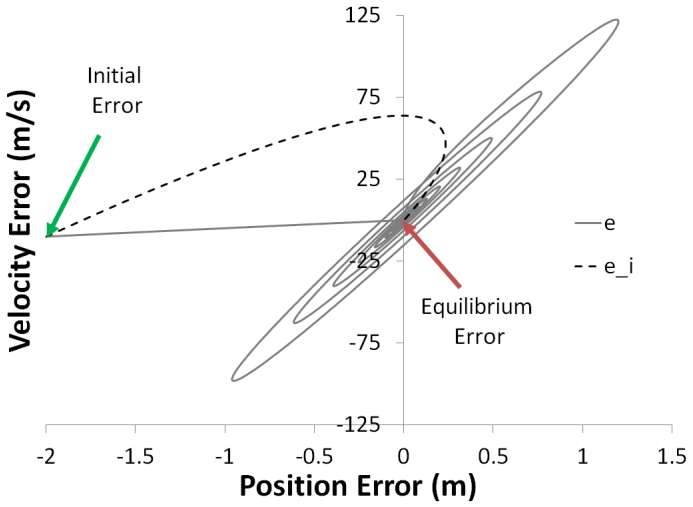


Figure 48: Error Space Trajectories of Ideal (e_i) and Real Estimator (e)

The observer gain constrains the position error by allowing the velocity error to grow to a state which produces initial conditions z_0 which, for the provided model, result in an estimate of the current output which closely resembles that of the physical system. And while the position is well known, the estimator must make an intelligent guess of what the velocity state is. This guess is based solely on the relationship between position and velocity described by the errant model. Given there is no feedback to the estimator of the error in velocity state, no corrective action can be performed. This problem is exacerbated when the measured outputs are not directly measured states. Take, for example, a scenario in which

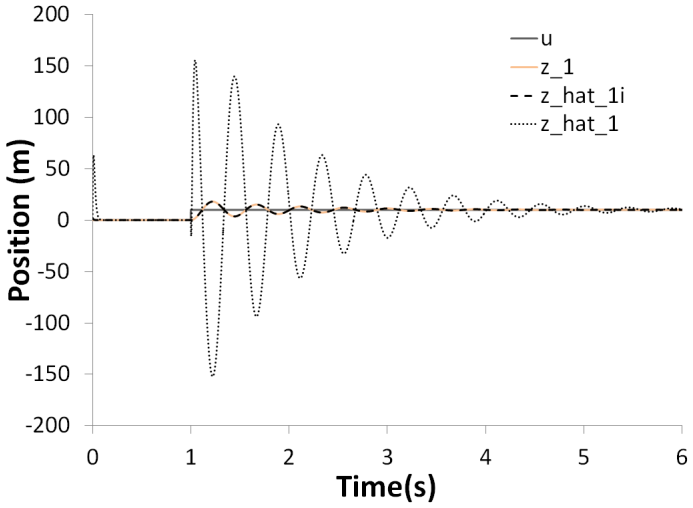


Figure 49: Simulated SMD Position Estimate (Acceleration feedback)

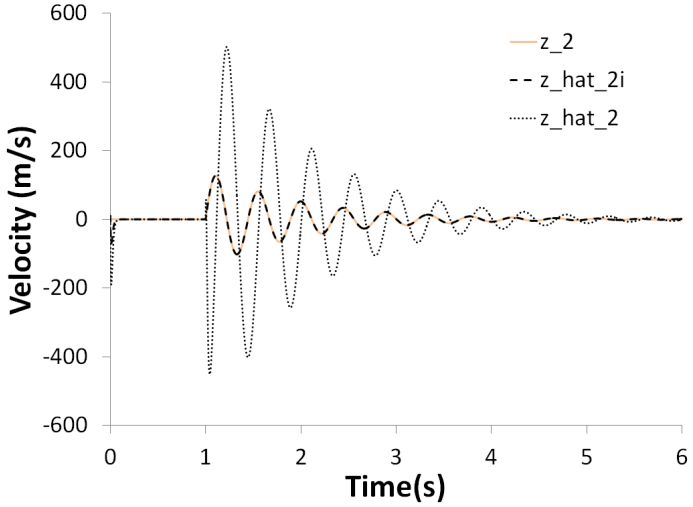


Figure 50: Simulated SMD Velocity Estimate (Acceleration feedback)

an acceleration measurement is used which is related to the states of the system through the modeled parameters via (235)

$$\ddot{x} = \begin{bmatrix} \frac{-(k+\Delta k)}{m} & \frac{-(c+\Delta c)}{m} \end{bmatrix} z + \frac{(k+\Delta k)}{m} u \tag{235}$$

Then, the connection to the system states is even more tenuous, and the error between plant and estimated states will be present in both estimates as shown in Figures 49 and 50.

Finally if the estimator model and the plant do not share equilibrium points, as is the case when the input to the system is a force on the cart and the spring is fixed at the end.

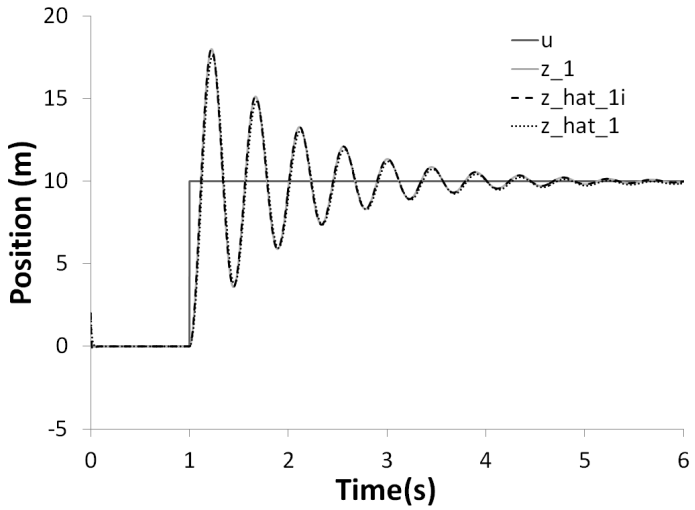


Figure 51: Simulated SMD Position Estimate (Position feedback, Force Input)

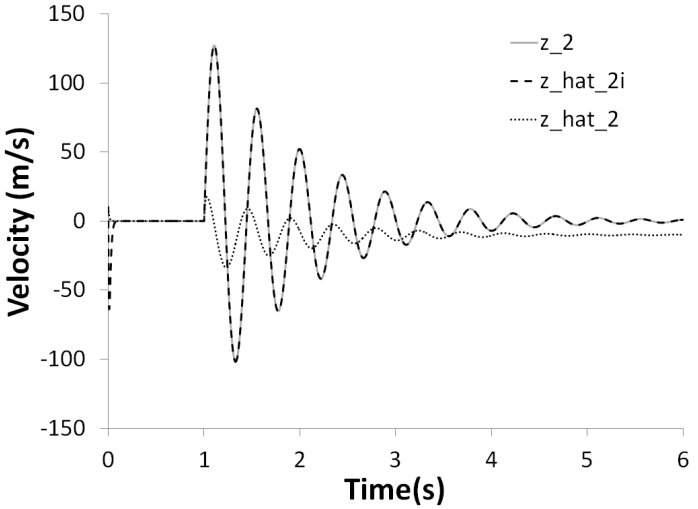


Figure 52: Simulated SMD Velocity Estimate (Position feedback, Force Input)

The error then, will not converge to the origin of the error space at steady state. Instead the control effort applied by the estimator gain seeks to drive the estimated position to the true position (Figure 51) by producing a constant offset in the velocity estimate (Figure 52). While physically impossible, it is the only mathematically possible solution that matches the output criteria at time t. The estimator model assumes it is traversing the state space according to the model dynamics to the predicted equilibrium state $\hat{z}_*$ given the input u, the gain L forces the trajectory to intersect the correct position state, resulting in the offset.

Initially the state error will converge to the origin, but given the dependency on system

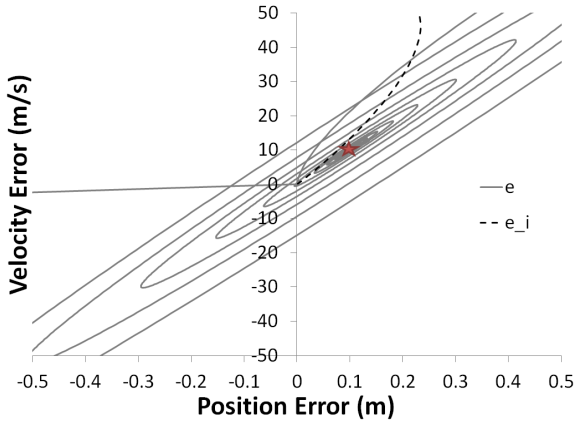


Figure 53: Error Space Trajectories of Ideal (e_i) and Real Estimator (e) (Force Input)

state, as the system moves in response to the command u it will diverge from the origin and as the system state converges, the state error will converge to an equilibrium point corresponding to that specific system state as demonstrated by Figure 53.

What this means is that parametric error is capable of producing bias in the state estimates. States which are used as measured outputs exhibit very little estimation error, but unmeasured states experience bias in proportion to the amount of modeling error and the current trajectory error, $(z - z_*)$. If the model and system plant share equilibrium states, (z_*), for a given set point, (u_*), the estimation error will eventually converge to zero, but at the rate of the system decay, and not the desired convergence rate of the estimator. Finally, if the model and plant have disparate equilibrium states the estimation error will persist if the desired set point is not the origin of the state space $z = 0$.

5.2 Extensions to other Estimation Structures

While this initial discussion has focused primarily on traditional Luenberger type linear estimators, most observers are based on these principles. They, therefore, suffer similar performance degradation in the presence of modeling error. Additional strategies have been tested both in simulation and experimentally applied to the FLASHMan testbed.

The following discussion will include results of both optimal and robust estimation strategies for observing the flexible system states of the FLASHMan testbed. Strategies for tuning are presented to improve the robustness of these approaches to modeling error.

However, the overall problem persists.

Without a quantitative measure of the accuracy of a given state estimate, from a single static estimator, errors will persist with the potential to significantly degrade the closed loop performance. In chapter 6 a strategy is proposed which successfully incorporates multiple sensors and multiple estimators to form a robust solution for the estimation of flexible system states.

5.3 Discrete Time Kalman Filtering for Flexible State Estimation

The Kalman filter (KF), since its introduction in the 1960's, has found widespread adoption for state estimation problems in dynamic systems subject to measurement noise. Studied unremittingly since its inception, the Kalman filter and its many variations have been discussed and implemented in a nearly limitless body of literature. To that end, this discussion will not attempt a repeat of the vast body of work. Instead it will focus on the specific application and the utility of the approach for robust state estimation under conditions of parametric uncertainty.

Kalman filtering was developed by R.E. Kalman [47] and serves as an statistically optimal recursive estimation routine for systems with Gaussian measurement and process noise. The Kalman filter is a conditional mean estimator, in that it attempts to estimate the conditional mean of the system state given a noisy state estimate, (i.e. determine $\hat{z}_k := E[z_k|y_k]$ where z_k and y_k are Gaussian distributed random variables).

Let the stochastic representation of the system be described by

$$\bar{z}_{k+1} = E[z_{k+1}|y_k] = E[A_d z_k|y_k] + E[Gw_k|y_k] \tag{236}$$

where w_k is a zero mean vector which represents the system process noise, (i.e. noise internal to the states). Therefore the propagated state estimate (estimate in between measurements) is

$$\bar{z}_{k+1} = A_d \hat{z}_k \tag{237}$$

Likewise the propagated estimate error covariance (M_{k+1}) is

$$M_{k+1} = E\left[\left(z_k - \bar{z}_{k+1}\right)\left(z_k - \bar{z}_{k+1}\right)^T | y_k\right] \tag{238}$$

$$= E\left[\left(A_d\left(z_k - \hat{z}_k\right) + Gw_k\right)\left(A_d\left(z_k - \hat{z}_k\right) + Gw_k\right)^T | y_k\right] \tag{239}$$

$$= A_d E\left[\hat{e}_k \hat{e}_k^T | y_k\right] A_d^T + GE\left[w_x w_k^T | y_k\right] G^T \tag{240}$$

$$= A_d P_k A_d^T + GQG^T \tag{241}$$

where P_k is the estimate error covariance immediately after measurement

$$P_k = \left(M_k^{-1} + C_d^T R^{-1} C_d\right)^{-1} \tag{242}$$

and Q is the process noise covariance which is assumed to be constant for all time (i.e. process noise Gaussian characteristics do not change with time).

The minimum variance estimate minimizes the mean square of the estimate error, and is the goal of the estimation routine.

$$\hat{z} = argmin\left(E\left[\left(z - \hat{z}\right)^2 | y\right], \hat{z}\right) \tag{243}$$

This problem simplifies to determining the state z at which the conditional probability distribution function

$$f_{x|z} = \frac{1}{(2\pi)^n |P|^{\frac{1}{2}}} e^{\phi} \tag{244}$$

where,

$$\phi = \left[\left(z - \bar{z}\right) - \left(M^{-1} + C^T R^{-1} C\right)^{-1} C^T R^{-1}\left(y - C\bar{z}\right)\right]^T \left(M^{-1} + C^T R^{-1} C\right) \tag{245}$$

$$\times \left[\left(z - \bar{z}\right) - \left(M^{-1} + C^T R^{-1} C\right)^{-1} C^T R^{-1}\left(y - C\bar{z}\right)\right] \tag{246}$$

reaches its mean value. The result is the optimal estimate

$$\hat{z} = E\left[z|y\right] = \bar{z} + \left(M^{-1} + C^T R^{-1} C\right)^{-1} C^T R^{-1}\left(y - C\bar{z}\right) \tag{247}$$

or written in a more familiar form processing sequential measurements instead of batch measurements:

$$\hat{z}_{k+1} = \bar{z}_k + L(y_k - \hat{y}_k) \tag{248}$$

where the observer gain is,

$$L = P_k C_d^T R^{-1} \tag{249}$$

Therefore, the Kalman filter consists of two distinct stages a prediction stage and an update stage.

1. Prediction Stage
   (a) Calculate apriori estimate $\bar{z}_{k|k-1}$
   (b) Calculate error covariance $P_{k|k-1}$
2. Update Stage
   (a) Sample measurement signal y_k
   (b) Calculate Kalman gain L_k
   (c) Update state measurement $\hat{z}_{k|k}$

Note that the recursive gain only varies in P_k, which in turn, contains no state or measurement dependent terms. Therefore, the estimator gain has no dependency on the current system state or measurement and can be precomputed. Furthermore as $k \to \infty$ $P_{k\to\infty}$ and $L_k \to$ the infinite time horizon Kalman gain L_∞, which can be determined a priori via the solution of the algebraic Ricatti equation (ARE):

$$A_d P_\infty + P_\infty A^T - P_\infty C^T R^{-1} C P_\infty + Q = 0 \tag{250}$$

The resulting infinite time horizon Kalman gain is

$$L = P_\infty C^T R^{-1} \tag{251}$$

Without the recursion, this means the Kalman filter is simply an optimally designed Luenberger state observer. As discussed in the following section, even with the recursion, parametric error results in poor system performance, degrading both the noise rejection properties of the Kalman filter and producing bias in the estimates.

5.3.1 Tuning Kalman Filters and Robustness Concerns

A Kalman filter is only "optimal" if a perfect model is available, the measurement covariance is known exactly, and the process noise covariance is known (an speculative assumption for any real system). Furthermore, perturbations to the measurements or state variables from external sources must be zero mean Gaussian valued random additions. These assumptions being infrequently validated, tuning a Kalman filter to achieve the desired performance characteristics becomes less rote mathematical evaluation and more of an art-form.

Determining the best Kalman gain for an application becomes a matter of tuning the process and measurement noise covariance matrices in order to reject the measurement noise. According to [92] (for a scalar system)

$$m_{k+1} = a_d^2 p_k + q \tag{252}$$

$$p_k = \frac{m_k r}{q + m_k c^2} \tag{253}$$

$$l_k = \frac{p_k c}{r} \tag{254}$$

and therefore the Kalman gain is determined by the balancing of the process noise covariance q and the measurement noise covariance r. Letting $r \to 0$

$$\lim_{r \to 0} l_k = \frac{p_k c}{q} = \frac{c}{\frac{r}{m_k} + c^2} = \frac{1}{c} \tag{255}$$

and the state estimate becomes

$$\hat{z}_k = \bar{z}_k + \frac{1}{c}(z_k - c\bar{z}_k) = \frac{z_k}{c} \tag{256}$$

Obviously this means that the state estimate is only a function of the measurement, and the model information is completely discarded. Conversely if the measurement noise covariance $r \to \infty$

$$\lim_{r\to\infty} l_k = \frac{m_k c}{r + m_k c^2} = 0 \tag{257}$$

i.e. the measurement is completely discarded and only the model is used to estimate the state:

$$\hat{z}_k = \bar{z}_k + 0\big(z_k - c\bar{z}_k\big) = a_d \hat{z}_{k-1} \tag{258}$$

Thus, the traditional procedure for tuning a Kalman filter is to directly measure the measurement noise covariance R through statistical analysis of the measurement signal, and then tune Q to achieve the desired level of filtering. By increasing Q more weight is placed on the measurements and by decreasing Q more emphasis is placed on the model in the state estimate.

If the model and internal system were perfect (i.e. no process noise, and no parametric error), there would be no need for measurement if the initial conditions were synchronized. However in practice this is never the case and increased process covariances Q are fictitiously attributed to modeling error to place extra weight on the measurements [62]. While this prevents the estimation error from diverging, it does not correct for the equilibrium bias error observed in the traditional Luenberger estimator [65](section 5.1.2).

"Robust" forms of the original formulation have been developed by many researchers, e.g. [62, 78, 107], which limit the filtering error covariance to a minimum value for a given level of parametric uncertainty. However this only enhances the consistency of the estimator over a range of values, relative to a Kalman filter designed without regard to the potential of parameter error, and not the accuracy of the estimates themselves, i.e. $e \to e^* + \Delta e^*$ as $t \to \infty$ at steady state where $\Delta e^* \leq \epsilon$ and not $e \to 0$ as $t \to \infty$.

5.3.2 Extended Kalman Filtering

If the parameter is not uncertain, but changes reliably and with known value as a function of time, the extended Kalman filter (EKF) can be applied. For the case of multiple

degree of freedom manipulators, the system state description is a function of joint variables which change continuously in time and are well known [60]. Coupled non-linearly, these joint variables affect the characteristic vibratory modes of the system (which are often modeled inaccurately). The result is a nonlinear description of the system dynamics. Application of the extended Kalman filter allows the extension the preceding analyses through linearization of the full nonlinear model of the system at each time step. From this linearized model a statistically optimal Kalman gain is generated. While capable of handling time dependent parameters, the problem of persistent estimate bias when error exists in the model parameters remains.

A summary of the extended Kalman filter algorithm (as described in [62]) is presented below.

- System and measurement model:

$$\dot{z} = a(z, u, t) + Gw \tag{259}$$

$$y_k = c(z(t_k), k) + v_k \tag{260}$$

- Jacobians:

$$A(z, t) = \frac{\partial a(z, u, t)}{\partial z} \tag{261}$$

$$C(z) = \frac{\partial c(z, k)}{\partial z} \tag{262}$$

- Time update:

$$\dot{\hat{z}} = a(\hat{z}, u, t) \tag{263}$$

$$\dot{P} = A(\hat{z}, t)P + PA^T(\hat{z}, t) + GQG^T \tag{264}$$

- Measurement Update:

$$L_k = \bar{P}(t_k)C^T(\bar{z}_k)\left[C(\bar{z}_k)\bar{P}(t_k)C^T(\bar{z}_k) + R\right]^{-1} \tag{265}$$

$$\dot{\hat{z}} = \bar{z}_k + L_k(y_k - \hat{y}_k^-) \tag{266}$$

Note that the primary difference between the extended Kalman filter and the Kalman filter (aside from the nonlinear model) is the time and state dependence on the error covariance P. This means that the Kalman gains must be computed each time interval rather than being precomputed or statically approximated as was the case in the KF procedure.

5.4 Robust Observers for Measuring Flexible System States

Many additional "robust" methods for determining flexible system states have been reported in the literature including, but not limited to, fuzzy logic [45], neural network [1], and sliding mode estimators [23, 69, 70, 105]. Sliding mode estimators were chosen for this work primarily because of their increasing popularity and reported performance guarantees. Implemented on the FLASHMan testbed, a benchmark comparison of the sliding mode estimator and traditional Kalman filter was performed to asses the validity of the claims reported in the literature. To that end, what follows is a general description and evaluation of the sliding mode observer (SMO) and its applicability to the task of reconstructing flexible system states for the control of robotic manipulators.

The function of the sliding mode observer is very similar to sliding mode control, a product of variable structures systems theory. The general analysis for these types of systems requires a segmentation of the state space, where the dynamics are governed by separate differential equations. The boundaries between segments are generally discontinuous. Therefore, the full dynamics of a system are represented as a piecewise continuous nonlinear function [111]. A subclass of hybrid systems referred to as variable structure systems have been studied since the early 1950's and resulted, most prominently, in the development of sliding mode control.

5.4.1 Sliding Mode Control

Sliding mode control is a nonlinear control method which influences the dynamics of a system through the application of a large discontinuous control effort. This large gain forces the state trajectory to a hypersurface in the state space called the "sliding surface" ($s = 0$). Once on the sliding surface (or hyper plane for $R^n, n > 3$), the gain keeps the state confined to the surface, where the trajectories have desirable characteristics and are invariant to parametric variation.

The process of deriving a sliding mode control is two-fold. First, one must choose a sliding surface s which exhibits desirable state trajectories. For example, the spring-mass-damper system (187) is fully described by the states corresponding to position (x_1) and velocity (x_2). By choosing a sliding surface $s = e_1 + \lambda e_2$ where $e = x - x_*$ the error trajectories are confined to a line which passes through the origin of the error space. Therefore, the state error will, after being driven to the sliding surface, converge exponentially to the origin of the error space (i.e. to the desired equilibrium state). The inclination of the sliding surface (λ) determines the performance of the closed loop system. As shown in Figure 54 the steeper the sliding surface (i.e. greater λ), the less position and more velocity error tolerated. Conversely, a shallow sliding surface corresponds to large position and smaller velocity error.

For position control, the steeper the sliding surface the more responsive the control system becomes (theoretically). There are, however, practical considerations, which can limit the achievable performance metrics of the controlled system. These limitations are discussed in section 5.4.2.

5.4.2 Limitations and Undesirable Behavior

One of the primary limitations of sliding mode control is the large gain applied to the actuators. Often resulting in actuator saturation, the large switching gain can excite unmodeled dynamics, wastes energy and results in undesirable chatter around the desired sliding surface [7].

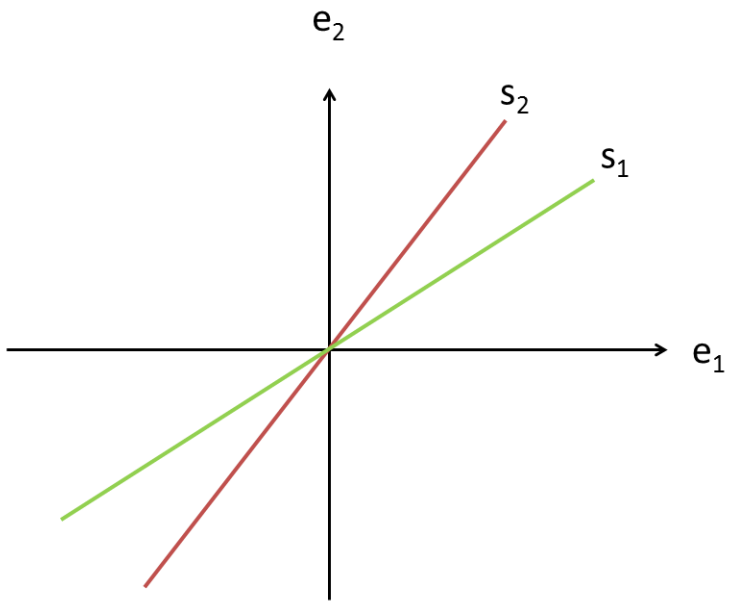


Figure 54: Inclination of the Sliding Surface $s_i = e_1 + \lambda_i e_2 \lambda_1 < \lambda_2$

Chattering occurs because of the practical impossibility of coincidence of the error trajectory with the sliding surface. In reality, the true error will always lie close but not coincident with the sliding surface. The signum function, therefore, applies the large sliding mode gain, pushing the trajectory across the surface where the gain direction is reversed again and over-shoots the surface. Thus, the error trajectory "chatters" across the surface as it "slides" to the origin of the error space as illustrated in Figure 55. Practical implementations of sliding mode control rely on high sampling rates and low control bandwidths to filter this high frequency chatter and avoid excitation of resonant system dynamics.

Other chatter mitigation strategies have been employed, including the implementation of a boundary layer and the insertion of integrators on the input side of the dynamics to raise the order of the system. These additions result in continuous inputs to the physical system [7]. By far the most popular of these techniques is the insertion of a boundary layer, below which the switching gain is replaced with a linear gain proportional to the distance from the sliding surface as demonstrated in Figure 56. Care must be taken however to avoid compromising the disturbance rejection properties of the sliding mode controller associated

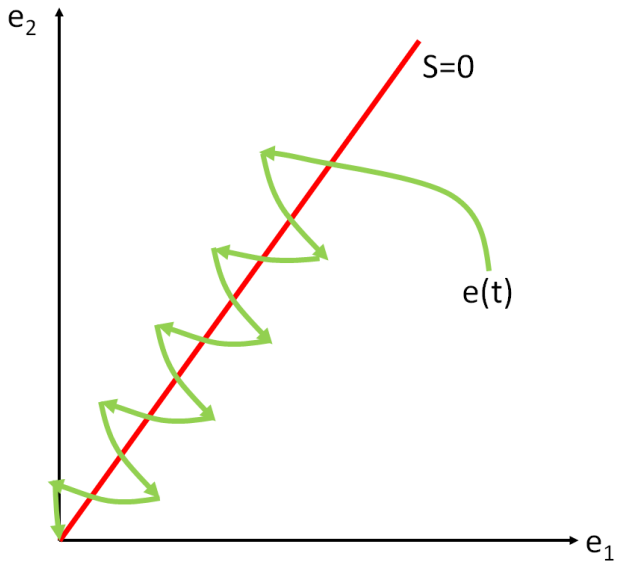


Figure 55: Chattering Behavior of Sliding Mode Control

with the high gain switching control.

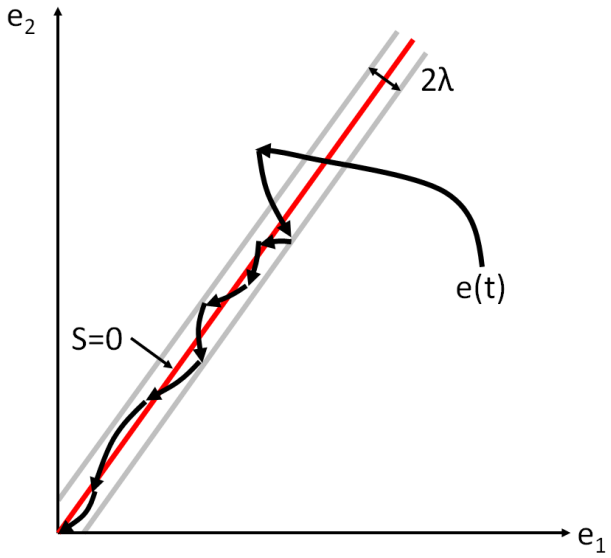


Figure 56: Boundary Layer Error Trajectory

5.5 Sliding-Mode Observation

Sliding-mode observers have widely been reported as a solution to the robustness issues present in typical state estimators. This result is generally attributed to a reported theoretical invariance to disturbances and parametric uncertainty during this sliding phase. While

many unique formulations have been employed, for linear and nonlinear systems, most sliding observers are predicated on the notion that a large gain is used to enforce a negativity constraint on the derivative of the Lyapunov energy function associated with the system. While the Walcott and Zak [103] observer form is used in this examination, the analysis of many sliding observer structures are similar, (e.g. the works in [24, 31, 32, 52, 111]).

Let a state estimator be defined with a structure similar to the linear observer, with an additional nonlinear sliding-mode term as defined in [103].

$$\dot{\hat{z}} = Az + Bu + L(y - \hat{y}) + K_s sgn(y - \hat{y}) \tag{267}$$

Where,

$$K_s = \rho_s P^{-1} C^T \tag{268}$$

Then the error dynamics of the estimator, given the plant (192) with parametric uncertainty confined to the state and input matrices, can be described by (269).

$$\dot{e} = A_0 e - K_s sgn(Ce) + \Delta Az + \Delta Bu \tag{269}$$

If the Lyapunov equation is chosen as

$$V = e^T Pe \tag{270}$$

then it can be shown that the derivative

$$\begin{aligned} \dot{V} = &- e^T Qe - 2\rho_s \|Ce\| + \left[z^T \Delta A^T Pe + \right. \\ &\left. e^T P \Delta Az + u^T \Delta B^T Pe + e^T P \Delta Bu \right] \end{aligned} \tag{271}$$

can be manipulated by the scalar gain ρ_s to dominate the terms which result from parametric uncertainty. This enforces a negative definite constraint and establishes sufficient

conditions for stability of the estimate error. If one were to restrict the types of parametric error under the ("matching") assumptions made in [103], (i.e. $\Delta Az = P^{-1}C^T h(t,z)$ and $\Delta Bu = P^{-1}C^T w(t)$), then if $\xi(t,z) = h(t,z) + w(t)$, then (271) reduces to (272).

$$\dot{V} = -e^T Qe - 2\rho_s \|Ce\| + 2e^T C^T \xi(x,t) \tag{272}$$

To ensure the negative definiteness of this expression, one must simply choose $\rho_s > \|\xi(x,t)\|$. Often in the literature this, or similar analyses are used to prove that $\hat{z} \to z$ and therefore $e \to 0$ [103, 104, 111]. Lyapunov stability asserts that if the derivative of a satisfactory Lyapunov candidate function is negative definite, then the system will converge to an equilibrium point of the system. Thus, the former argument is predicated on the assumption that the equilibrium of the error system is the origin. However, from (269) it is immediately obvious that while $\Delta Az \neq -\Delta Bu$, $e_* \neq 0$. Therefore, as $t \to \infty$, $e \to e_*(z,u)$. This relationship indicates that one would expect similar behavior to linear estimators under conditions of parametric uncertainty. In fact, the only assertion that can be made from (272), without extremely restrictive assumptions on the specific type of parametric error, is that the convergence rate of a sliding-mode estimator to the equilibrium error state will be equal to or faster than the underlying linear estimator itself [103]. This is a consequence of the large "bang-bang" style gain used to drive the errors to the sliding surface.

5.5.1 Return to the Single Degree-of-Freedom Example

To demonstrate the convergence properties of the sliding-mode observer, recall the single degree of freedom mass-spring-damper example from section 5.1.4. Assuming identical levels of parametric error and simulation parameters as described in Tables 6 and 7, and given measurements of the cart's position as the chosen output, the state estimates are shown in Figures 57 and 58 for $\rho_s = 10$ and $Q = I_{(2x2)}$. While the position estimates again track well and converge more quickly than the Luenberger estimates, the velocity error in the transient regime is significant and appears to offer no clear improvement over the linear estimator in Figure 47.

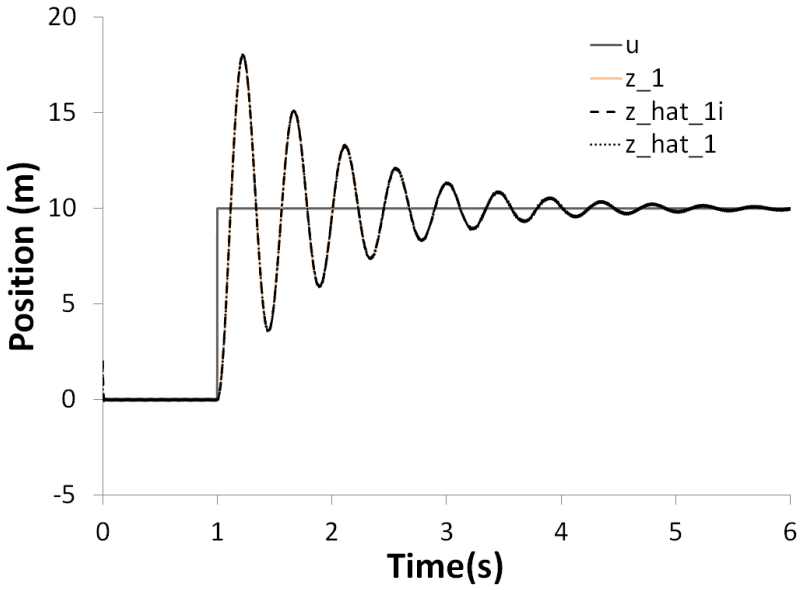


Figure 57: Simulated SMD Position Estimate (Position feedback)

From a plot of the error space (Figure 59) it is obvious that the switching gain forces the state error to the sliding surface, and initially the error slides down the surface to the origin. After initial convergence, motion is initiated causing a perturbation of the equilibrium state and forcing the error state away from the origin. The sliding mode gain keeps the errors confined to the surface but does not drive the error state to the origin. Instead, errors converge to the origin only once the oscillations of the mass-spring-damper system settle.

If parametric error is experienced in the output or feed-through matrices, then the orientation of the sliding surface itself will be state and input dependent. Thus the "instantaneous sliding surface" is a hyperplane which passes through the origin but its orientation varies with time dependent parameters. As the orientation changes, the error state is forced to the new surface and begins sliding towards the equilibrium point defined by the current state and input, coincident with the instantaneous sliding surface.

While significant design freedom is available to the control engineer in the form of scalar gain ρ, linear gain L, and the positive definite matrix Q, performance is directly linked to the command input and true state. While manipulation of these variables, along with boundary layer function λ in the case of the boundary layer sliding mode observer, can result in performance and robustness improvements relative to a poorly designed estimator,

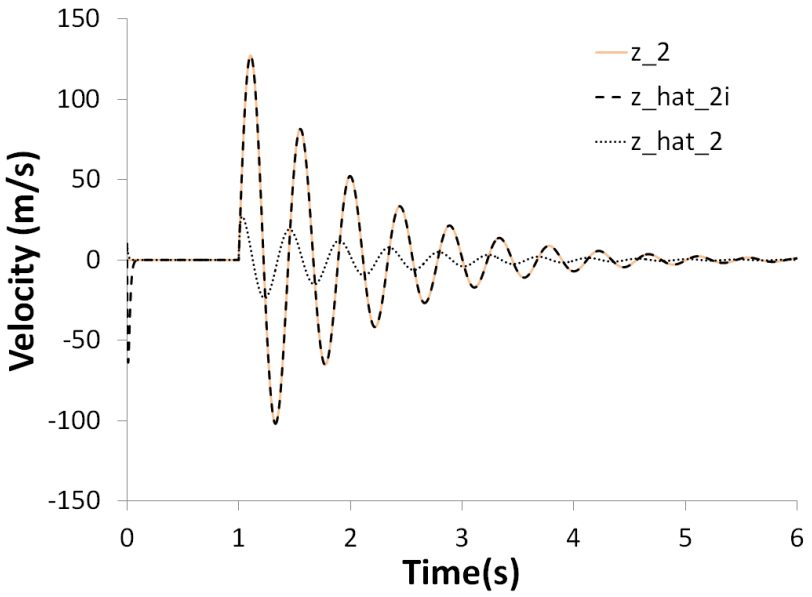


Figure 58: Simulated SMD Velocity Estimate (Position feedback)

the technique itself suffers many of the same limitations observed in linear estimators for parametrically uncertain systems.

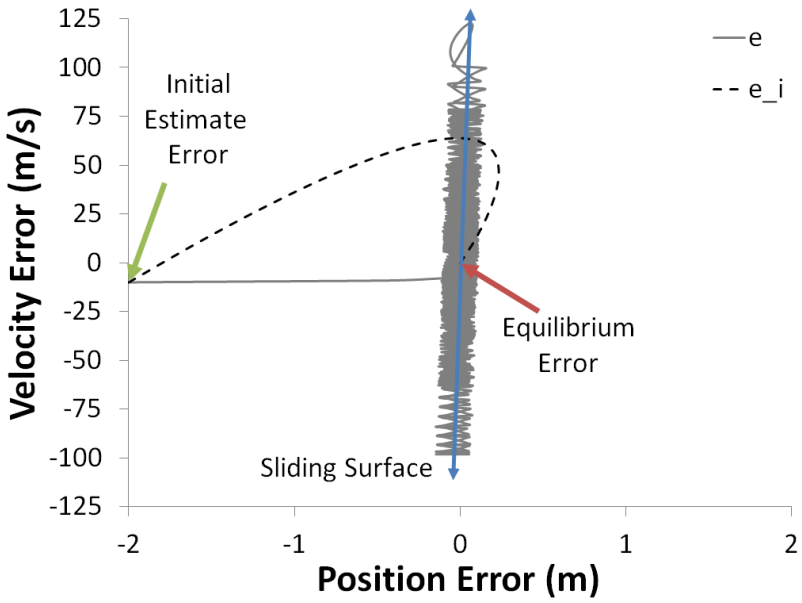


Figure 59: Error Space Trajectories of Ideal (e_i) and Real Estimator (e)

5.5.2 Simulation Results

Simulations of the FLASHMan testbed under different payload estimates were performed for the express purpose of determining the performance of the Wallcot and Zak sliding mode observer (SMO) in comparison to a traditional Luenberger observer for the reconstruction of flexible system states. An observer of the form (267) was designed with a scalar value $\rho_s = 0.005$ and $Q = 1E^5$, chosen through iteration to limit noise and ensure fast convergence of the estimated output to the true output (estimated tip acceleration to measured tip acceleration). Luenberger gains of

$$L = \begin{bmatrix} 0.0236 \\ 0.0109 \\ 0.0070 \\ 0.0951 \\ -10.3294 \\ -0.0781 \end{bmatrix}$$

Table 8: Motion Profile

Parameter	Value	Unit
Displacement	0.2	m
Maximum Velocity	2	m/s
Acceleration	30	m/s^2
Dwell	4	s

Table 9: Nominal Model Parameters

Parameter	Value	Unit
Payload (m_t)	0.281	kg
Cart Mass (m_c)	10	kg
Arm Length (L)	0.42	m
Elastic Modulus (E)	$7E^{10}$	N/m
Density (ρ)	2700	kg/m^3
Area Moment (I)	$1.0114E^{-10}$	m^4
Belt Stiffness (K_b)	$2.1814E^5$	N/m
Cart Damper (C_d)	100	Ns/m
Structural Damping Coefficient (γ)	0.0005	Ns/m

were used for both the SMO and traditional linear estimator. A version of the SMO with a boundary layer (BLSMO), below which the discontinuous gain $sgn(y-\hat{y})$ is replaced with a linear gain, was also implemented. Each observer system was used to track the manipulator given a cycle of open loop trapezoidal velocity profile moves $x_m(t)$ as described in Table 8. To excite the vibratory dynamics, the chosen trajectory is very aggressive, approaching the actuator limits of the robot.

Performance is assessed by examining the estimated tip position, which is calculated from the state estimates, and the true tip position. Under nominal conditions (table 9) each observer performs nearly identically, the only difference being the increased noise of the SMO and BLSMO from chatter as illustrated in Figure 60. To assess the performance degradation under varying levels of parametric error, the modeled tip mass (m_t) was perturbed from the nominal value. Note that while tip mass (i.e. payload) was chosen, for ease of experimental modification as demonstrated in chapter 6, any of the system parameters which effect the system's natural frequencies and damping ratios yield similar results.

When the tip mass is over estimated (m_t = 0.45kg), the modeled natural frequencies are lowered and for a given acceleration measurement the amount of tip displacement is exaggerated relative to the true value. As shown in Figure 61, the performance of the

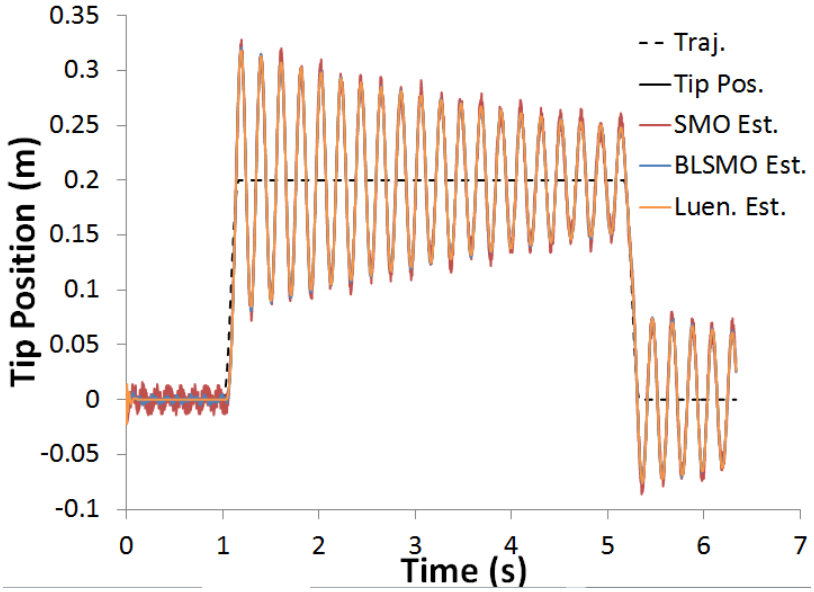


Figure 60: Nominal Estimates of Tip Position

estimators is comparable, with all significantly overestimating the actual tip displacement. Eventually the estimated tip position and true position would converge as the result of damping in the physical system. However, while oscillations persist, significant estimation error persists.

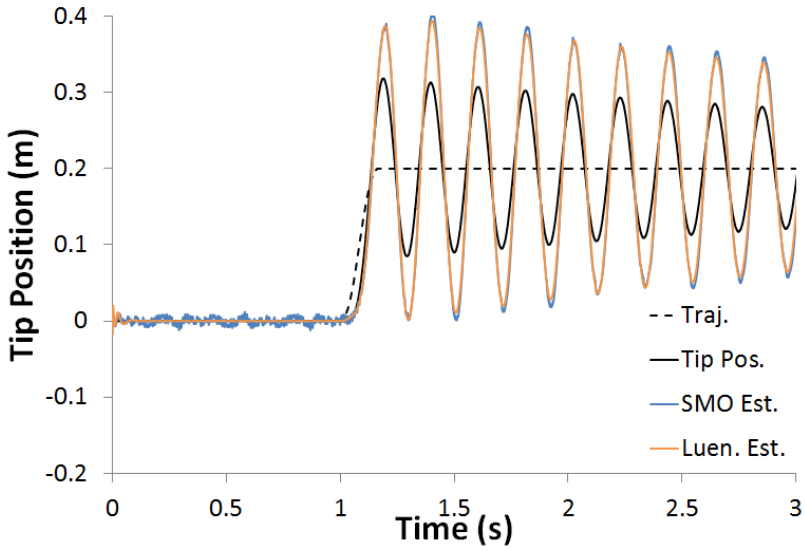


Figure 61: Estimates of Tip Position with Over Estimation of Tip Mass

If instead the value of the tip mass is underestimated (m_t = 0.1kg) the result is as shown in Figure 62. Because of the dependence on the output matrix C in the definition of the

sliding mode gain K_s (268), the parametric variation of C results in larger sliding mode gains, increasing the chatter.

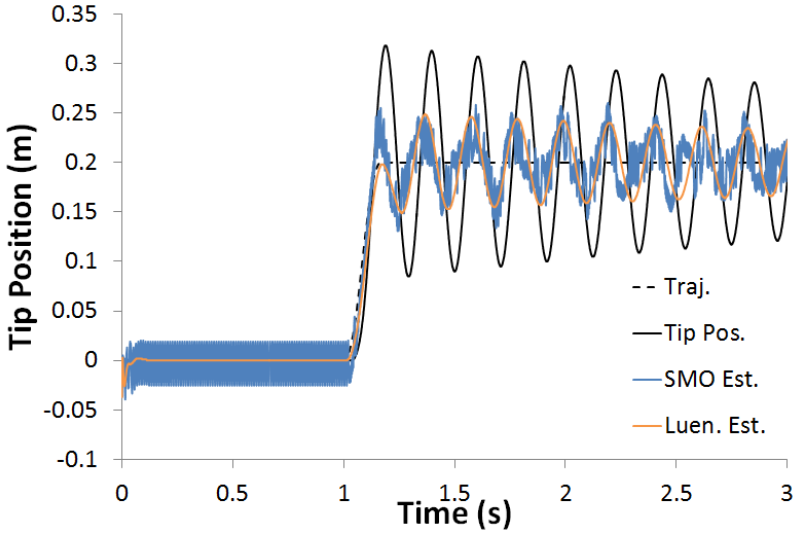


Figure 62: Estimates of Tip Position with Under Estimation of Tip Mass

As expected, as the parametric error grows, the estimation error increases. Figure 63 is a direct comparison of the mean squared estimation error for perturbations of the tip mass below and above the nominal value. For this application, the sliding mode observer provides no appreciable benefits over a traditional observer in terms of diminishing mean squared error. In fact, with the addition of chatter, noise is introduced into the system increasing the error.

Estimation error is indicative of control system performance, the larger the error, the worse the performance. Therefore, in order to ensure adequate performance of the complete control system, a maximum acceptable level of estimation error must not be exceeded.

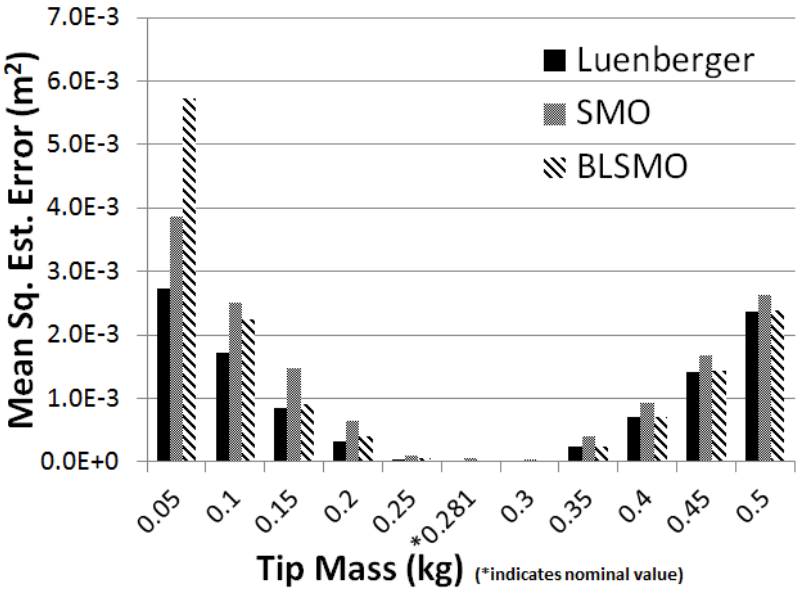


Figure 63: Mean Square Estimation Error

5.6 Experimental Results

Experiments were carried out to assess the ability of the sliding mode estimator to expand the stability bounds and improve performance of the FLASHMan testbed with uncertain parameter estimates. A linear output-weighted quadratic regulator, with penalties on tip acceleration and cart position, was used to control the position of the payload, while traversing motion cycles identical to those in Section 5.5.2, and a NI SmartCamera machine vision system was used as an unbiased ground truth measurement of tip position.

Figure 64 illustrates the structure of the complete control system. The state feedback controller produces a desired belt displacement which is implemented via a tight PD controller around the drive motor's position. The shaft mounted encoder measurement is used as the input to the state observer, and the sole output feedback measurement is obtained through a single MEMs accelerometer mounted near the tip of the arm. Nominal values of the system parameters were identified as those from Table 9 and the model of the flexible system is identical to the one used in the simulations of Section 5.5.2. However, the linear estimator gains were chosen via an infinite time horizon Kalman filter formulation to reduce the contribution of sensor noise to the comparison. All performance metrics of the system response were evaluated using the camera measurements and not the observer

estimates. Control loops were evaluated at 1000Hz and vision measurements were reported at an approximate rate of 30Hz and accuracy of approximately $0.0004m$.

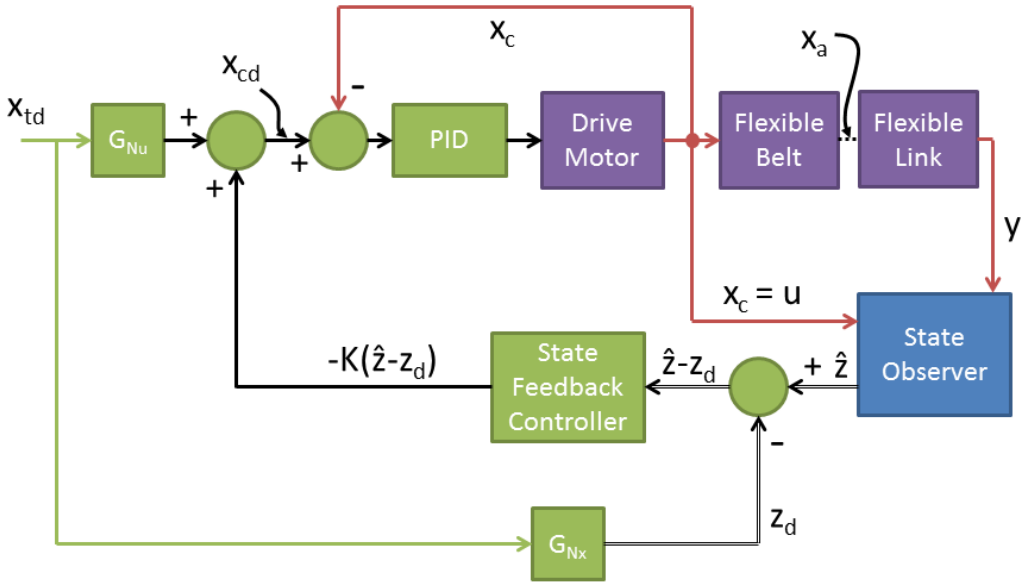


Figure 64: Complete Control System Diagram

Given a nominal system model, the controlled system response to the trajectory exhibits a significant reduction in oscillation and settling time when compared to the open loop response, as illustrated in Figure 65. A significant increase in damping ratio is also noted. At issue is whether the addition of a robust estimator improves the stability margin and extends the performance benefits of the nominal system to a perturbed system in which error exists in one or more parameters.

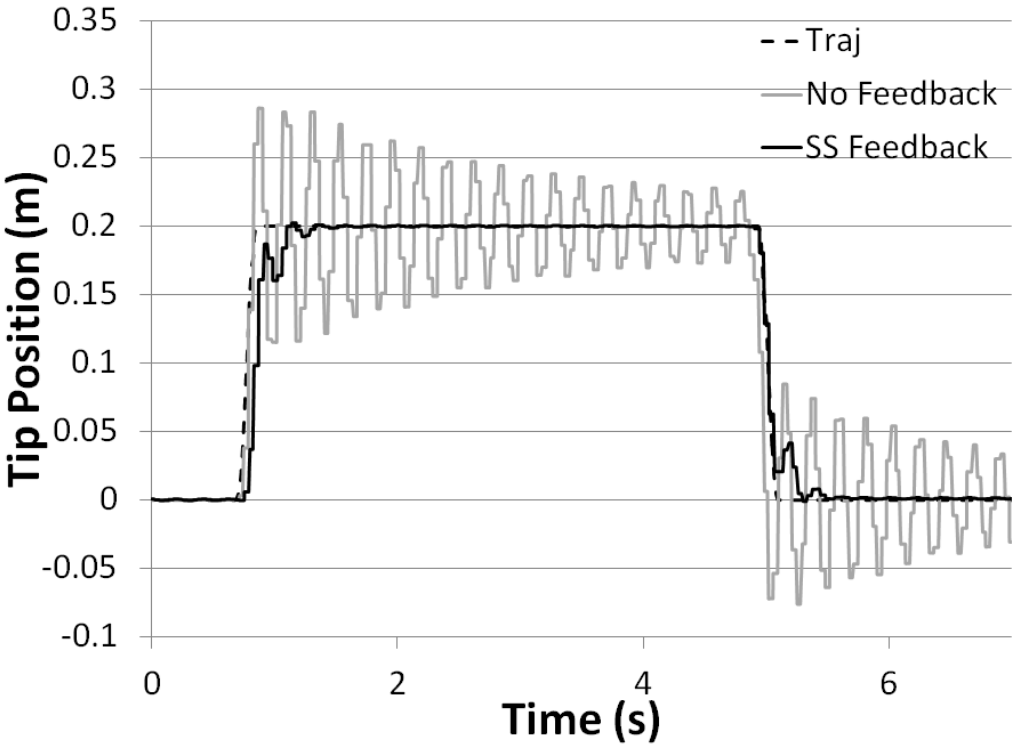


Figure 65: State Space vs. No Control

For the purpose of this study the modeled mass of the payload was perturbed from its nominal value to determine the deterioration of control performance. Figure 66 demonstrates the range of system responses for the sliding mode estimator given the minimum and maximum values of tip mass chosen experimentally to lie at the upper and lower bounds of stability. Above a modeled tip mass of 0.375kg, the first mode of vibration ($1st$ beam mode) becomes marginally unstable and quickly destabilizes, and below 0.225kg, the second mode of vibration (the cart/belt mode) destabilizes, regardless of the observation strategy. Thus, experimentally, the sliding mode observer does not usefully extend the stability bounds of the system beyond those offered by the Kalman filter.

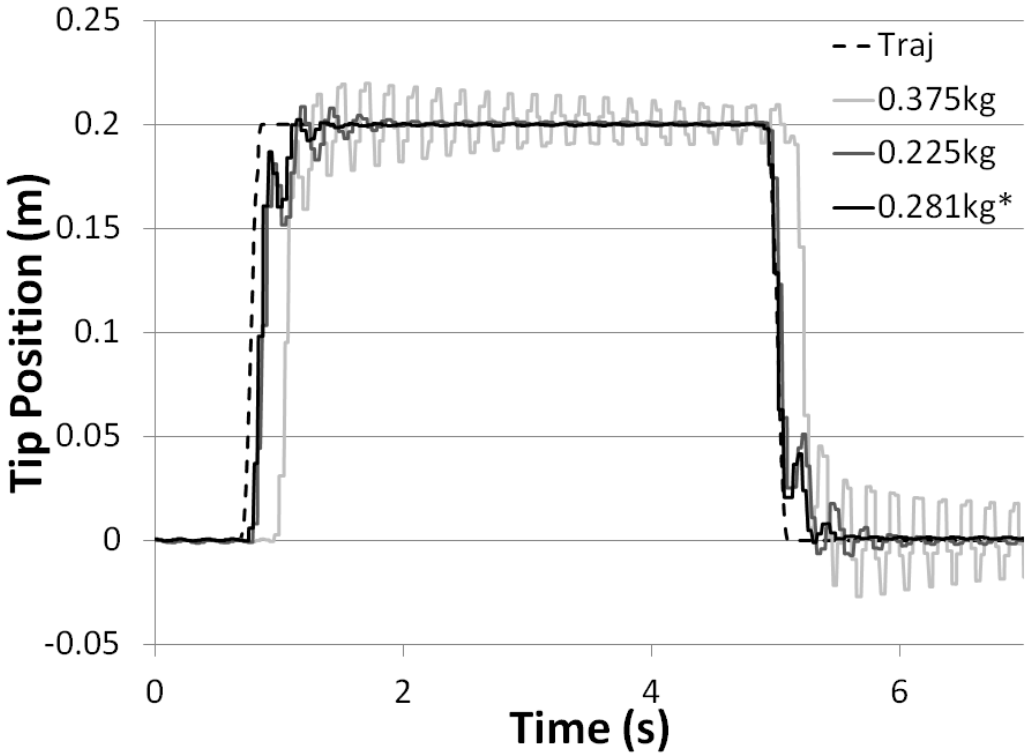


Figure 66: Measured Responses Using Sliding Mode Observer for Different Modeled Tip Masses (*indicates nominal value)

The comparison of damping ratios in figure 67 (equivalent damping ratio of a second order system evaluated via logarithmic decrement procedure assuming only the range of peaks between the maximum peak and the first peak within 5% of target position) over the range of parametric variation indicates a marginal performance increase through the use of a sliding mode estimator. The addition of a boundary layer eliminates this advantage and significantly deteriorates damping performance for tip mass values below the true value.

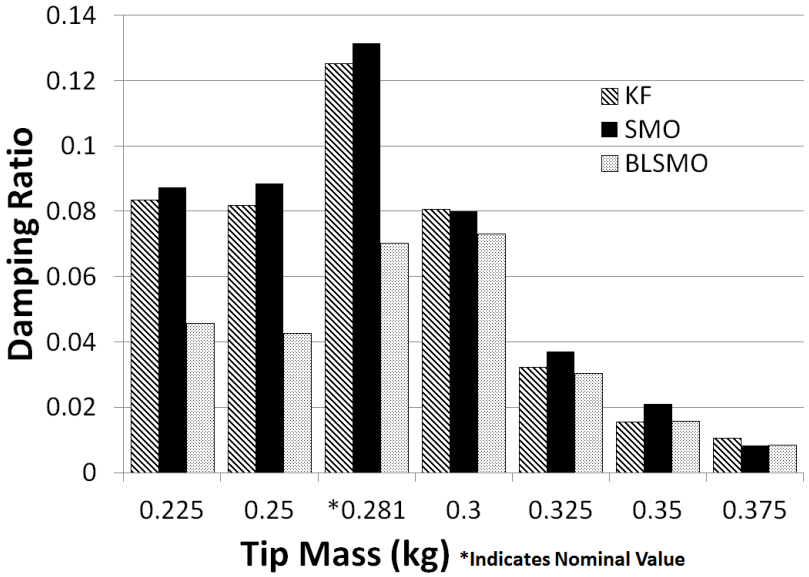


Figure 67: Damping Ratio vs Tip Mass

Results are similar for the 5% settling time metric as demonstrated in Figure 68. The sliding mode observer exhibits marginal performance improvements over the Kalman filter and BLSMO over the range of parametric error.

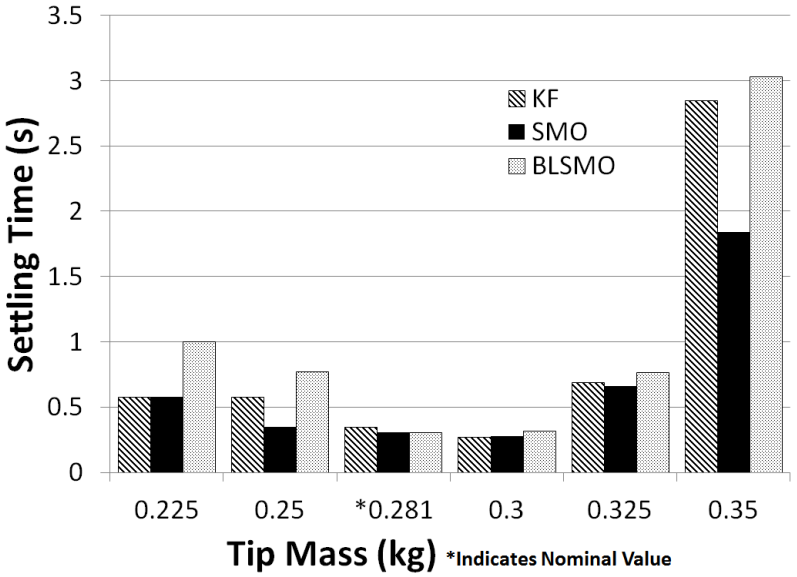


Figure 68: Settling Time vs Tip Mass

While slight improvements are observed in the performance metrics of the controlled system responses, the sliding mode observer also results in significant chatter, which adds noise to the system estimates, and therefore, to the control actions. This chatter results in

an audible hum of the servo drive at the sampling frequency.

These results bare practical significance as they expose a fundamental property of state estimators. With a limited set of unique measurements, k where $k < n$ a mapping $M_1 : \Re^k \rightarrow \Re^n$ maps the measurement to the state space and is defined by the provided dynamic model of the system. In the case of an errant model, these k measurements will produce an incorrect state vector which is mapped back into the the measurement space by $M_2 : \Re^n \rightarrow \Re^k$ (defined by output and feed-through matrices) to produce k error signals. The error signals are used to adjust the n states by manipulating the initial conditions of the dynamic system through the observer gain. If the k error signals have converged to zero, all n states are assumed to have converged as well. However, this is simply a sampling of the error in the states, and this assertion requires that both mappings M_1 and M_2 be accurately known. If for example, the measurements are the states, M_2 only maps the k states back to the measurement space, and no feedback of the $n-k$ unmeasured states is provided, no control action can be applied to drive these unmeasured states to the true state values, regardless of the approach taken for gain selection. Only by correcting the mappings M_1 and M_2 can the true state be extracted from a set of k unique measurements.

5.7 Remarks on the Use of Single Model/Single Sensor Estimators

Given that the sliding mode observer has been demonstrated to be of only marginal improvement over more traditional linear estimators, the problem of flexible state variable estimation in the presence of parametric uncertainty remains. However, the equilibrium state definitions derived in this work lead to an interesting and novel solution to this problem.

From the preceding analysis it has been demonstrated that an estimator will drive the output error to zero at the expense of estimate error. For a single sensor feedback signal, only one error quantity is available, and when that error is driven to zero, the estimator is assumed to have converged to the true state values. As demonstrated, this is only the case for a perfectly modeled system. Flexible link manipulators are unique in that their displacement fields are functions of their mode shapes. This means that a sensor

placed in one location and another placed in a separate location on the same link provide measurement signals which are disparate linear combinations of the modal responses. These modal responses are, in turn, affected by parametric error to differing relative degrees.

Since a single estimator using a single sensor is essentially blind to its own state error, the only way to correct the state estimates is to adapt the true model parameters [96]. Using multiple sensors placed in locations described in Chapter 4, multiple error signals are generated, permitting quantification of the estimator performance. In Chapter 6 multiple sensors and multiple estimators will be utilized in the creation of a novel multiple model adaptive estimator structure. This estimation approach uses the equilibrium analysis of Section 5.1.2, to determine the true parameters. In doing so, the true state vector is also recovered.

Chapter VI

MULTIPLE MODEL ADAPTIVE FLEXIBLE STATE ESTIMATION

In order to form a general solution for the problem of correctly identifying the state of a flexible manipulator under conditions of parametric uncertainty, it becomes necessary to correct any errant parameters and recover the true dynamic model. The degree to which this task can be accomplished is directly linked to the control system's ability to cancel vibrations and improve the utility of these devices. While the discussion in chapter 5 brought to light the persistent bias and performance degradation of flexible manipulators when the underlying estimation model contains parametric error, no form of gain manipulation can correct for this action. Furthermore, traditional estimators are, in a sense, unaware of any estimate errors resulting from parameter error as the true state is unavailable. However, this fact can be utilized to develop a robust solution in the form of a multiple model adaptive estimator.

A series of multiple model adaptive estimators is generated in the following analysis, which offers the ability to select the true state from a bank of state estimators executed simultaneously. This is accomplished by utilizing the knowledge that the level of parametric error is directly proportional to the persistent bias in the state estimates. Therefore, a selection metric can be defined which chooses the correct state estimate or combination of state estimates for the operating regime, while also identifying the best approximation of the true (unknown) parameter. Finally, a complete multiple model control system is developed and implemented on the FLASHMan testbed, consisting of an adaptive multiple model estimator and multiple model control.

6.1 Operating Principles

As previously discussed, parametric bias results in perturbations to the error dynamics of an estimator which seek to drive the system to equilibrium points other than the origin. Therefore, modeling error results in persistent errors between the estimated and true state

values. These results are reproduced in (273).

$$\begin{aligned}\dot{e}_{s_i}(t) = (A - L_{s_i}C_{s_i})\, e(t) + [\Delta A(t) - L_{s_i}\Delta C_{s_i}(t)]z(t) \\ + [\Delta B(t) - L_{s_i}\Delta D_{s_i}(t)]u(t)\end{aligned} \tag{273}$$

Meaning that at steady state, the estimation error converges to an equilibrium position,

$$\begin{aligned}e^{\star}_{s_i}(t) = (A - L_{s_i}C_{s_i})^{-1}\,[\Delta A(t) - L_{s_i}\Delta C_{s_i}(t)]z(t) \\ + (A - L_{s_i}C_{s_i})^{-1}\,[\Delta B(t) - L_{s_i}\Delta D_{s_i}(t)]u(t)\end{aligned} \tag{274}$$

as previously discussed. Note that "steady state" again implies that the state error has converged to its equilibrium value and not that the controlled system has reached steady state (much slower). For the latter, when the true plant and model have equivalent equilibrium states, as is often the case for reduced order models, at t_{ss}, $[\Delta A - L_{s_i}\Delta C_{s_i}]\, z(t_{ss}) = -[\Delta B - L_{s_i}\Delta D_{s_i}]\, u(t_{ss})$. Therefore, after the controlled system has converged to a static state $z(t_{ss})$ corresponding to a static input $u(t_{ss})$, the estimate error will vanish.

In the transient regime (after convergence of the estimate error, but before convergence of the controlled system) the true state $z(t)$ is unknown and therefore the estimate error is unquantifiable. However, if it can be determined that $e^{\star}_{s_i}(t)$ is nonzero, then it can be asserted that estimate error exists and is in proportion to the modeling error. Unfortunately, no quantitative measure of the estimate error can be determined from a single state estimate. If instead, multiple distinct measurements are used to produce independent estimates of the state vector, a direct measure of model inaccuracy can be determined under specific conditions.

6.2 State Estimate Difference Weighting

Modal models of systems can exhibit error in three potential aspects: natural frequency estimates ω, modal damping estimates $\hat{C}$, and the input state relationship $\hat{Q}$. These errors are directly correlated to errant state estimates from the closed loop state observers. Conceptually, in a closed loop state estimator, the observer gain (L) drives the estimated output ($\hat{y}$) toward the measured output (y) by manipulating the estimated states via the injection of extra control effort into the estimator dynamics. This serves as an attempt to recover from mismatched initial conditions. When the model is an accurate representation of the true system, the effort ($L(y-\hat{y})$) drops as $\hat{y}$ tracks y, (i.e. after the initial conditions have been matched). This is akin to using a small maintenance control effort to track a desired trajectory vs. the control effort in response to a step command. However, if the model is not a perfect match, the observer continues to inject control effort in an attempt to achieve equality of these signals. This indicates that the level of persistent effort applied by the estimator is in proportion to the level of parametric error in the underlying system model when noise is mitigated with a suitable estimator design.

Figure 69 illustrates this effect for the end of arm accelerometer based FLASHMan estimates described in the preceding chapter. As the model parameters deviate from nominal the error between the estimated and measured acceleration grows.

The effort applied is dependent on two factors, the output estimation error $y-\hat{y}$, and the control gain L. Regardless of the method used to design L, the end result for a linear system is a pole placement which determines the convergence properties of the estimator dynamics, ideally $\dot{e} = (A - LC)e$. Placing poles through traditional methods (e.g. Ackerman, Kalman filter, direct pole placement) is dependent on an inversion of the observability test matrix or, in the case of the discrete time Kalman filter, the stochastic observability gramian. As previously discussed, the gains produced through any design procedure will vary significantly based on sensor location.

Given the FLASHMan gantry robot previously described, and with three accelerometers positioned in their optimal placements from the preceding analysis (Figure 70). The sensor positioned on the cart (S_1) is an optimal placement for reconstructing the states associated

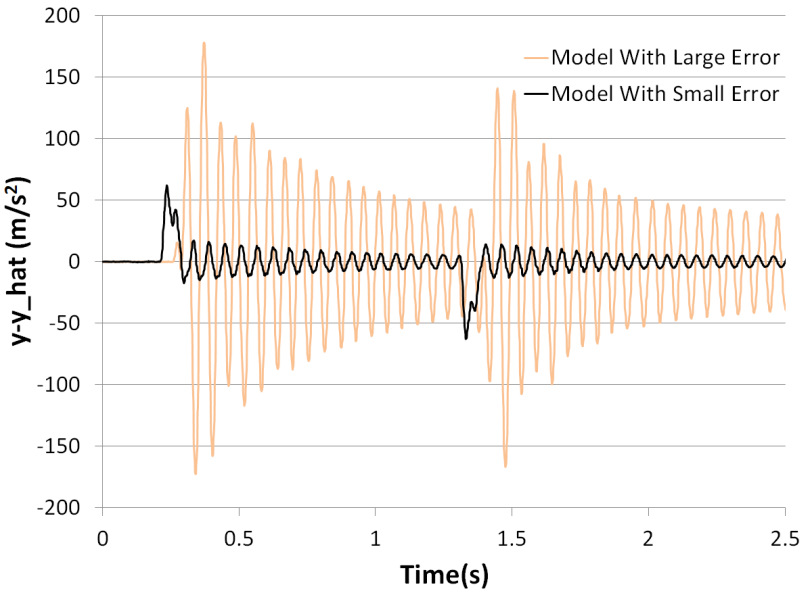


Figure 69: Proportionality of Output Error to Parametric Modeling Error (FLASHMan Nominal Model vs. Exaggerated Tip Mass (.75kg))

with the belt/cart mode of vibration. Likewise the sensor at the tip (S_3) is optimal for the 1st mode of beam vibration, and S_2 is optimal for the 2nd mode of beam vibration.

A quantity proportional to the magnitude of the modeling error can be constructed by taking the difference between two estimates of the full system states produced from disparate sensors.

State Estimate Difference Theorem

Let $\delta_z(t) = \hat{z}_{s_1} - \hat{z}_{s_2}$ then, assuming the estimator dynamics are significantly faster than the plant dynamics,

$$\delta_z(t) = \chi_1(\Delta A z(t) + \Delta B u) + \chi_2(\Delta C_{s_1} z(t) + \Delta D_{s_1} u(t)) + \chi_3(\Delta C_{s_2} z(t) + \Delta D_{s_2} u(t)) \quad (275)$$

where χ_1, χ_2, and χ_3 are constant matrices and ΔA, ΔB, ΔC_{s_1}, ΔD_{s_1}, ΔC_{s_2}, and ΔD_{s_2} are the perturbations to the system matrices attributed to parametric modeling error.

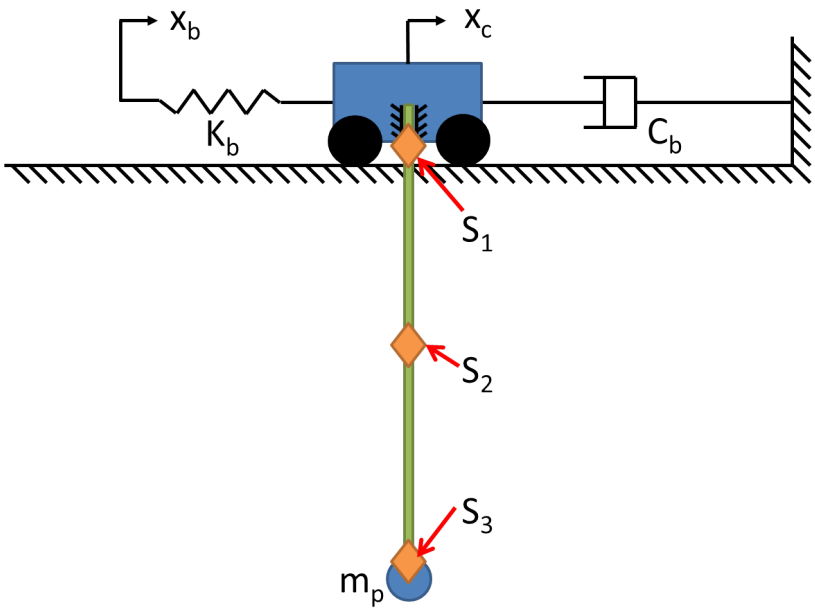


Figure 70: Flexible Gantry Robot with Multiple Accelerometer Placements

Proof. The dynamics of estimator 1, which uses sensor 1, are described in (276). A and B are the modeled plant and input matrices, $y_{s_1} = (C_{s_1} + \Delta C_{s_1})x(t) + (D_{s_1} + \Delta D_{s_1})u(t)$, and $\hat{y}_{s_1} = C_{s_1}\hat{x}(t) + D_{s_1}u(t)$.

$$\dot{\hat{z}}_{s_1}(t) = A\hat{z}_{s_1}(t) + Bu(t) + L_{s_1}(y_{s_1}(t) - \hat{y}_{s_1}(t)) \tag{276}$$

Similarly, the estimator dynamics for estimator 2 using sensor 2 are

$$\dot{\hat{z}}_{s_2}(t) = A\hat{z}_{s_2}(t) + Bu(t) + L_{s_2}(y_{s_2}(t) - \hat{y}_{s_2}(t)) \tag{277}$$

and therefore, the state estimate difference dynamics $\dot{\hat{z}}_{s_1}(t) - \dot{\hat{z}}_{s_2}(t)$ are described by (278).

$$\begin{aligned}\dot{\hat{z}}_{s_1}(t) - \dot{\hat{z}}_{s_2}(t) =& A(\hat{z}_{s_1}(t) - \hat{z}_{s_2}(t)) + L_{s_1}C_{s_1}(z(t) - \hat{z}_{s_1}(t)) - L_{s_2}C_{s_2}(z(t) - \hat{z}_{s_2}(t)) \\ &+ (L_{s_1}\Delta C_{s_1} - L_{s_2}\Delta C_{s_2})z(t) + (L_{s_1}\Delta D_{s_1} - L_{s_2}\Delta D_{s_2})u\end{aligned} \tag{278}$$

Again assuming the error states have converged to their equilibrium values (i.e. $e_{s_i} \rightarrow e^*_{s_i}$). In comparison to the estimator dynamics (very fast), the plant and estimate difference dynamics are assumed to be quasi-static ($\dot{\delta}_z \approx\rightarrow 0$).

$$\begin{aligned} 0 =& A\delta_z(t) + L_{s_1}C_{s_1}e^*_{s_1}(t) - L_{s_2}C_{s_2}e^*_{s_2}(t) + (L_{s_1}\Delta C_{s_1} - L_{s_2}\Delta C_{s_2})z(t) \\ &+ (L_{s_1}\Delta D_{s_1} - L_{s_2}\Delta D_{s_2})u \end{aligned} \tag{279}$$

Substituting for $e^*_{s_i}$ in (279)

$$\begin{aligned} \delta^*_z(t) = &-A^{-1}\left[L_{s_2}C_{s_2}(A - L_{s_2}C_{s_2})^{-1} - L_{s_1}C_{s_1}(A - L_{s_1}C_{s_1})^{-1}\right](\Delta Az(t) + \Delta Bu) \\ &-A^{-1}\left[L_{s_1} + L_{s_1}C_{s_1}(A - L_{s_1}C_{s_1})^{-1}L_{s_1}\right](\Delta C_{s_1}z(t) + \Delta D_{s_1}u) \\ &+A^{-1}\left[L_{s_2} + L_{s_2}C_{s_2}(A - L_{s_2}C_{s_2})^{-1}L_{s_2}\right](\Delta C_{s_2}z(t) + \Delta D_{s_2}u) \end{aligned} \tag{280}$$

and therefore,

$$\delta^*_z(t) = \chi_1(\Delta Az(t) + \Delta Bu) + \chi_2(\Delta C_{s_1}z(t) + \Delta D_{s_1}u) + \chi_3(\Delta C_{s_2}z(t) + \Delta D_{s_2}u) \tag{281}$$

where χ_i are independent of the true values of the system parameters. □

Thus, the magnitude of the estimated state difference is proportional to the cumulative modeling error. In the simplest cases, when no parametric error is associated with the output and feed-through matrices, $\Delta Az(t) + \Delta Bu$ can be directly calculated through an inversion of χ_1 (i.e. when the output is a direct state measurement or a known linear combination of states). While complicated when the output and feed-through matrices contain parametric error, there are only two contexts when the state estimation difference will be zero: 1) when the system parameters are identical to the true parameter; and 2) when the true state matches its equilibrium value (i.e., if the model and plant share equilibrium states).

For flexible link manipulators with modal state space control systems and accelerometer measurements, this relationship can be further refined. Recall that the state space equations can be defined as:

$$\dot{z} = \begin{bmatrix} 0_{(n\times n)} & I_{(n\times n)} \\ -\omega^2_{(n\times n)} & -\hat{C}_{(n\times n)} \end{bmatrix} z + \begin{Bmatrix} 0_{(n\times m)} \\ \hat{Q}_{(n\times m)} \end{Bmatrix} u \tag{282}$$

where ω and $\hat{C}$ are the modal frequencies and damping coefficients and $\hat{Q}$ describes the modal contribution of the input command to each state. In this form the measurement from an accelerometer at a position x_{s_1} can be described as,

$$\begin{aligned} y_{s_i} = \Big[-[1, \psi_1(x_{s_i}), \ldots, \psi_n(x_{s_i})]\, \Phi_{n\times n} \omega^2_{(n\times n)} \quad -[1, \psi_1(x_{s_i}), \ldots, \psi_n(x_{s_i})]\, \Phi_{(n\times n)} \hat{C}_{(n\times n)} \Big] z \\ + \Big[[1, \psi_1(x_{s_i}), \ldots, \psi_n(x_{s_i})]\, \Phi_{(n\times n)} \hat{Q}_{n\times n} \Big] u \end{aligned} \tag{283}$$

Let $\beta_{s_i} = [1, \psi_1(x_{s_i}), \ldots, \psi_n(x_{s_i})]$, as it was in chapter 3. Therefore, any parametric modeling error in A, B, C, and D can be attributed to the sub-matrices ω^2, $\hat{C}$, and Q.

$$(\Delta A z(t) + \Delta B u) = \begin{bmatrix} 0_{(n\times n)} \\ I_{(n\times n)} \end{bmatrix} \left(\begin{bmatrix} -\Delta\omega^2 & -\Delta\hat{C} \end{bmatrix} z(t) + \Delta\hat{Q} u(t) \right) \tag{284}$$

$$(\Delta C_{s_i} z(t) + \Delta D_{s_i} u) = \beta_{s_i} \Phi \left(\begin{bmatrix} -\Delta\omega^2 & -\Delta\hat{C} \end{bmatrix} z(t) + \Delta\hat{Q} u(t) \right) \tag{285}$$

The difference between the estimates produced by estimators using two separate accelerometer measurements is

$$\delta_z(t) = \left[\chi_1 \begin{bmatrix} 0_{(n\times n)} \\ I_{(n\times n)} \end{bmatrix} + \chi_2 \beta_{s_1} \Phi + \chi_3 \beta_{s_2} \Phi \right] \left(\begin{bmatrix} -\Delta\omega^2 & -\Delta\hat{C} \end{bmatrix} z + \Delta\hat{Q} u \right) \tag{286}$$

or more concisely

$$\delta_z(t) = \Lambda_{(2n\times n)}\gamma(t) \tag{287}$$

Where Λ is a known transformation from the estimated state difference space to the effective modal error space, $\Re^{2n} \rightarrow \Re^{n}$, where n is the number of modes considered in the state space model of the flexible manipulator. $\gamma(t)$ is therefore, at any time t a relative measure of the parametric accuracy of the modal model (i.e. the first element of γ is an indication of the accuracy of the natural frequency and damping attributed to the first mode of vibration).

Note that this error quantity can be related to the difference between estimates of a desired output quantity. For example, if accurate payload position estimates are desired, the output y_p is

$$y_{p_{si}} = C_p z_{si}(t) + 0u(t) \tag{288}$$

Therefore, the difference in position estimates from each observer is

$$y_{p_{s1}} - y_{p_{s2}} = C_p\left[z_{s1}(t) - z_{s2}(t)\right] = C_p\delta_z(t) \tag{289}$$

And consequently,

$$y_{p_{s1}} - y_{p_{s2}} = C_p\Lambda_{(2n\times n)}\gamma(t) \tag{290}$$

This means that the difference between two estimates of a projected output is also related to the parametric error in the system model. In both cases, γ serves as a quantitative metric directly linked to the level of modeling error. Since this quantity cannot be isolated from the true state vector and input, the exact errors can not be directly determined. However, if multiple estimators are utilized, γ offers an unbiased metric for comparing the relative level of parametric error between identically designed estimators with disparate underlying dynamic models. The goal, therefore, is to determine, the best state estimate in real time, given multiple estimators operating on the same sensor information and command inputs.

The least squares solution for γ (291) can be utilized directly and solved a priori, yielding a computationally efficient method for comparing state estimators.

$$\left(\Lambda^T\Lambda\right)^{-1}\Lambda^T\delta_z^*(t) = \gamma(t) \tag{291}$$

The norm of γ serves as a quantitative value on which a comparison of the accuracy of multiple estimators can be based. The end result is a systematic approach for determining the relative reliability of estimates produced with identical linear state observers with disparate models of the dynamic system. Given the same set of measurements, state estimates produced with more accurate models of the system will produce smaller $\gamma(t)$ and, thus, be given given a larger weight, w_i, in the state vector selection algorithm.

6.3 Multiple Model Adaptive Estimates and the "Sub-Estimator"

Multiple model control and estimation systems were originally applied to highly nonlinear systems like the F-16 fighter jet and stimulated muscle tissue [72, 73], where the dynamics change based on the operational mode. For the control of practical flexible manipulators it poses a simple and effective method of determining accurate system states and approximate model parameters for these often complex and marginally stable systems where the availability of accurate state information plays such a large role in control system success.

While illustrated in two dimensions for the purpose of clarity in Figure 71, the multiple model adaptive estimator can be carried out in as many dimensions as there are uncertain model parameters. A bank of estimators is generated, each built on a slightly different model, with the desired result of spanning the space of possible true parameter sets. The end goal of which is to identify the estimator, and therefore, the state estimate which best describes the observed behavior of the system. The finer the gradations in parameter variations the more accurate the end result, but accuracy comes at the expense of computational efficiency requiring large estimator banks.

In this work, each estimate is produced in real time. Further, a selection routine is used to identify the quality of the estimates, select from among the estimator bank, and fuse

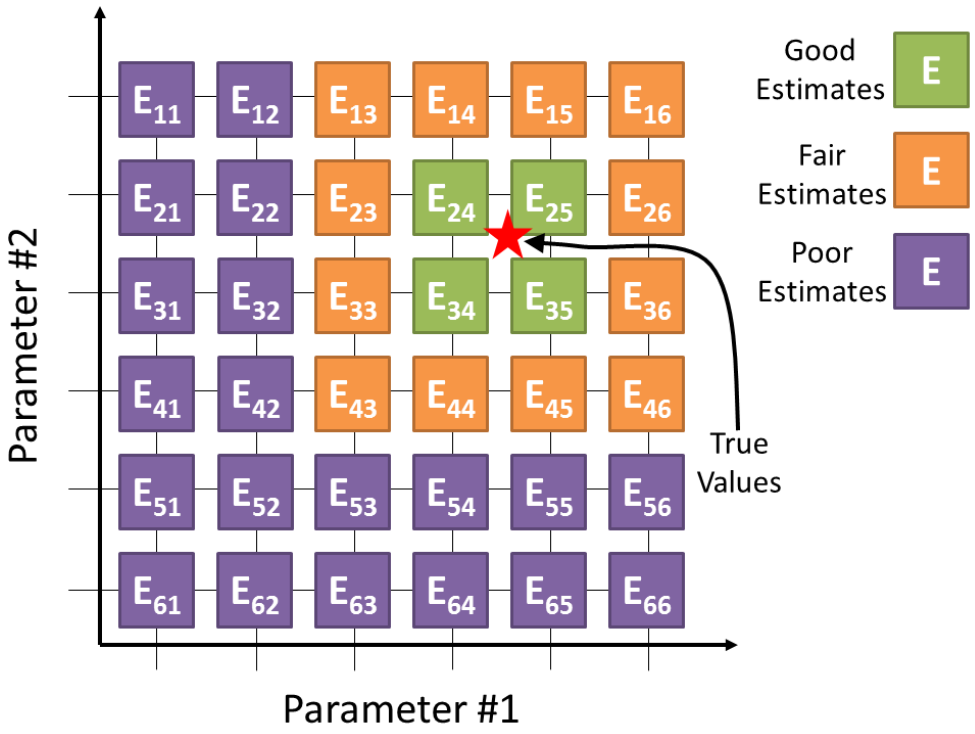


Figure 71: Multiple Model Adaptive Estimator Description

the most likely estimates to achieve the "best" state vector approximation. Often these algorithms are predicated on the conditional probability of a state estimate being correct, through the online adaptation of the system parameter, or through a priori assumptions of the system behavior.

Instead of relying on the conditional probability of a particular model outcome, estimation performance can be directly calculated by utilizing the derived relative state estimate difference weighting routine. The general form of the MMAE is therefore nested with "sub-estimators" which are used to calculate the state estimates and weights associated with each model variant. A pictorial description of this arrangement is shown in Figure 72.

Each sub-estimator consists of three independent state estimators which are used to produce the final state estimate and a weight corresponding to the reliability of the model. Two of the observers, E_2 and E_3, use independent sensors to produce estimates which are compared to form the state estimate difference weight. The third, E_T uses all the sensor measurements to produce the best possible estimate of the true state.

Therefore, each observer in the MMAE consists of a sub-estimator. Producing the true

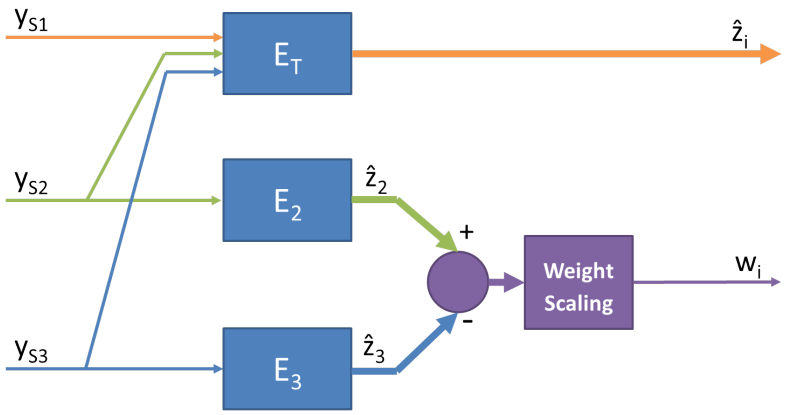


Figure 72: Nested Estimation - State Estimate and Weight Generation

estimate of the system state then becomes a matter of using the generated weights to fuse the state estimates from all of the individual observers.

For design purposes $\|\gamma\|$ is not directly utilized, instead a modified weighting quantity is defined in (292) which provides a value between 0 and 1.

$$w_i = \left(\frac{1}{e^{\rho[\gamma^T(t) W \gamma(t)]}} \right)^p \tag{292}$$

$W_{(n\times n)}$ is, therefore, a weighting matrix which determines which modes are considered to be the most important for accurate state reconstructions. For example, if the controller heavily penalizes the first mode of the system, it is important to have an accurate estimate of that mode, while the others can be weighted less significantly in the estimation routine. Increasing p provides additional separation between weights of relatively similar state estimates.

6.4 Multiple Model Switching Adaptive Estimator (MMSAE)

The simplest way to utilize the generated weights is to simply select the state estimates corresponding to the nested observer with the most desireable weight. This process is illustrated in Figure 73.

While Figure 73 illustrates payload estimation, any unknown parametric error can be corrected in the same fashion. The end products from this form of estimation are the "best fit" state estimate and an approximation of the unknown model parameter. While

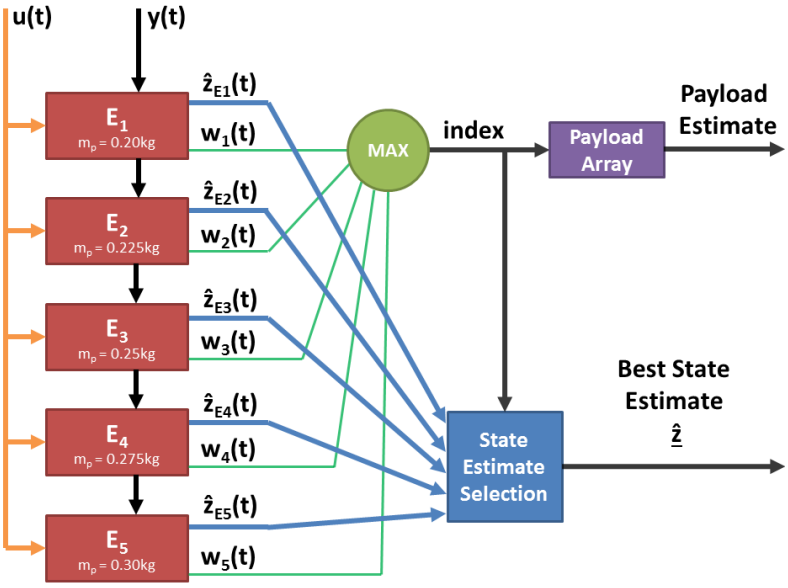


Figure 73: Multiple Model Switching Adaptive Estimator

switching between the estimates is the simplest method, it results in discrete jumps in the resulting state observations. The unknown parameter and state estimate are determined from only one nested state estimator, meaning that if the quantization of the parametric space is too coarse, poor approximations of the state vector and parameter will be chosen. If two or more observers operate with similar levels of modeling error, the selection algorithm will switch back and forth between them: this switching is exacerbated in the presence of measurement noise.

6.4.1 MMSAE Simulation

Simulation of the MMSAE algorithm for identification and compensation for uncertain payloads of the gantry style FLASHMan flexible manipulator in Figure 70 were performed. Nominal parameters for the system used in the simulations are listed in table 10.

Five potential payload values were chosen in even increments between lower and upper bound estimates of 0.2kg and 0.3kg, as listed in table 11.

The simulated manipulator was given an open loop command of the belt motion, and sensor responses (accelerations) at the positions listed in Table 12 along the flexible link were simulated as outputs to the MMSAE.

Table 10: Nominal Simulation Parameters

Parameter	Value	Unit
Payload (m_t)	0.281	kg
Cart Mass (m_c)	10	kg
Arm Length (L)	0.42	m
Elastic Modulus (E)	$7E^{10}$	N/m
Density (ρ)	2700	kg/m^3
Area Moment (I)	$1.0114E^{-10}$	m^4
Belt Stiffness (K_b)	$2.1814E^5$	N/m
Cart Damper (C_d)	100	Ns/m
Structural Damping Coefficient (γ)	0.0005	Ns/m

Table 11: MMSAE Simulation Payloads

Estimator	E_1	E_2	E_3	E_4	E_5
Payload Estimate (m_t)	0.200	0.225	0.250	0.275	0.300

In the absence of measurement noise, the resulting state estimate difference weights produced by the MMSAE are shown in Figure 74. With larger weights indicating a more accurate state estimate, it is clear that the estimators built on 0.275kg and 0.300kg payloads perform the best. However, it is also apparent that the weights peak in correspondence with state equilibrium crossings. Initially crossings are observed in all modes of vibration as state trajectories oscillate around the commanded equilibrium states (i.e. $\begin{bmatrix} -\Delta\omega^2 & -\Delta\hat{C} \end{bmatrix} z(t) = -\Delta\hat{Q}u(t)$). As the modes converge to their steady state values in accordance with their damping ratios, the frequency of the weight oscillation diminishes to $2T_1$ corresponding to the oscillation about the equilibrium state where T_1 is the period of oscillation corresponding to the 1st mode of vibration. Furthermore as all the modal energy dissipates, the system settles to its equilibrium value and $\gamma \to 0 \Rightarrow w_i \to 1$. Meaning that since the estimators share equilibrium states, as the system settles to the equilibrium all the estimators accurately predict the true state.

As a result, for practical implementation purposes, weights can be low-pass filtered or averaged over time to smooth discrete peaks and avoid unnecessary switching when the predicted state estimate from a less reliable observer happens to instantaneously match the true state. Figure 75 shows the evolution of the estimator weights low pass filtered with a

Table 12: Sensor Placements

Sensor #	s_1	s_2	s_3
Location (m)	0.015	0.250	0.420

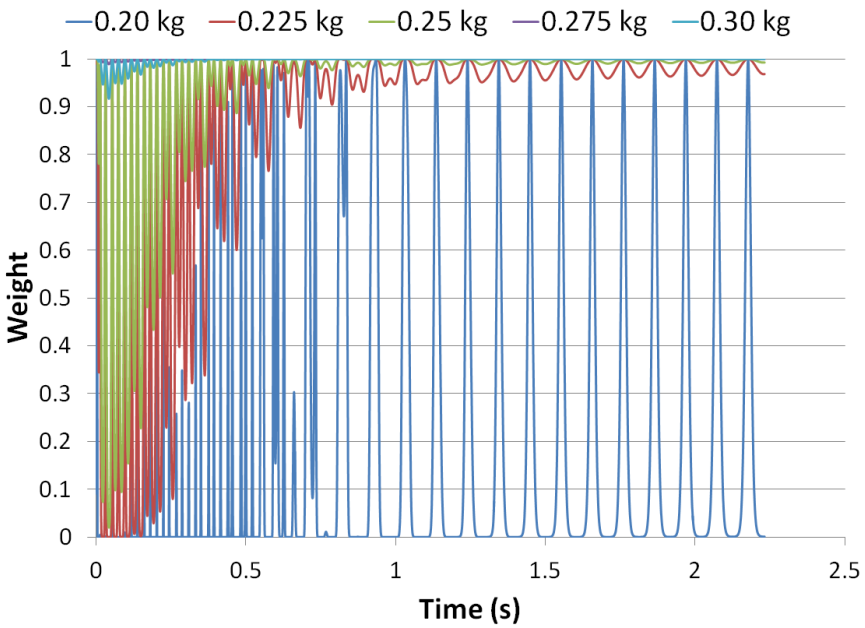


Figure 74: Unfiltered State Estimate Difference Weights Without Noise

time constant of 1 second and weighting all modes equally.

These weights are used to produce an estimate of the payload mass as shown in Figure 76. Note that the switching algorithm simply selects the state vector from the observer which receives the largest weight at the current time-step. Therefore the true parameter value is only approximated to the nearest discrete entry provided by the designer. From Figure 76 it is apparent that 0.275kg estimator is the preferred choice given the 0.281kg nominal value.

The resulting individual state estimates are presented in Figures 77(a) - 77(f) where the states z_1 - z_3 are the modal coordinates of the system and z_4 - z_6 are the derivatives of the modal coordinates. Because the payload estimate is constant over time, the selected estimates are derived from a single estimator, E_4. While not a perfect match, the estimates are the closest to the true values out of all of the provided sub-estimators.

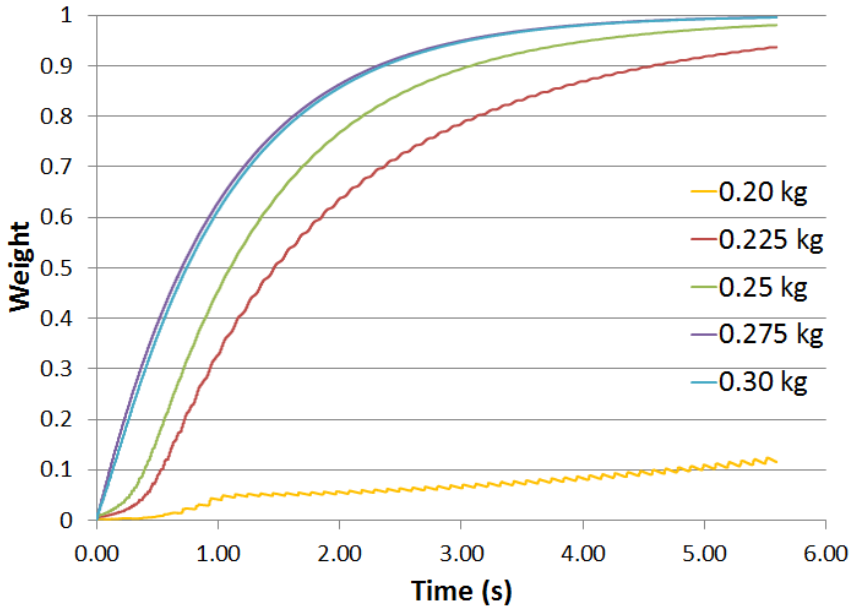


Figure 75: Filtered Weights Without Noise

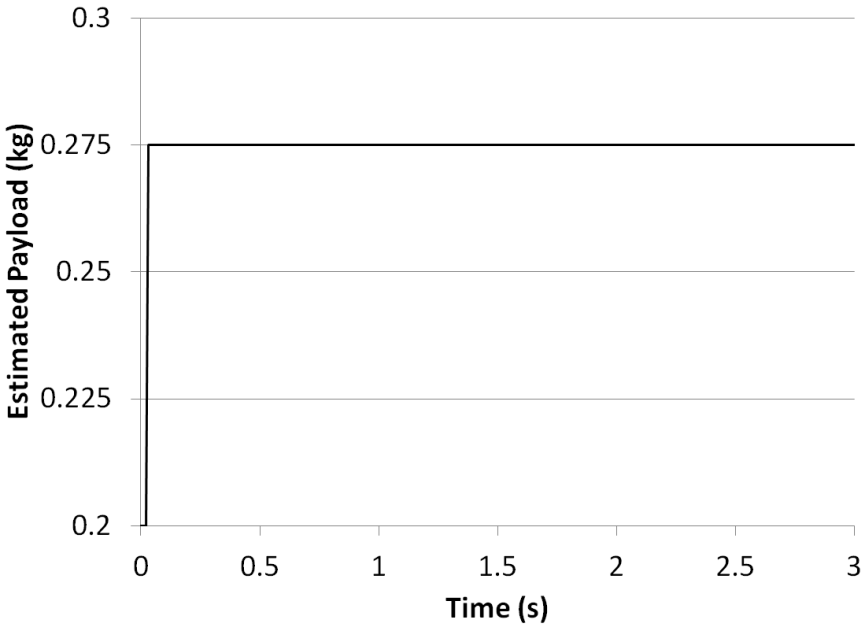
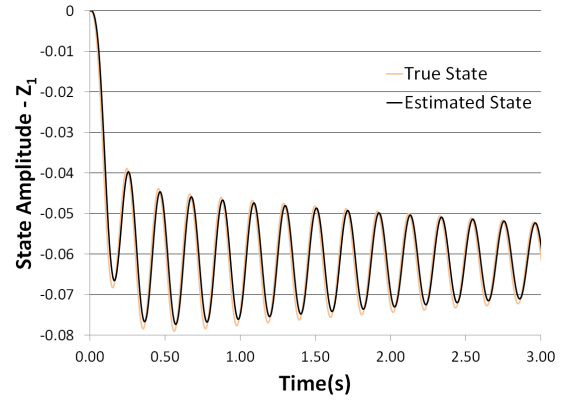
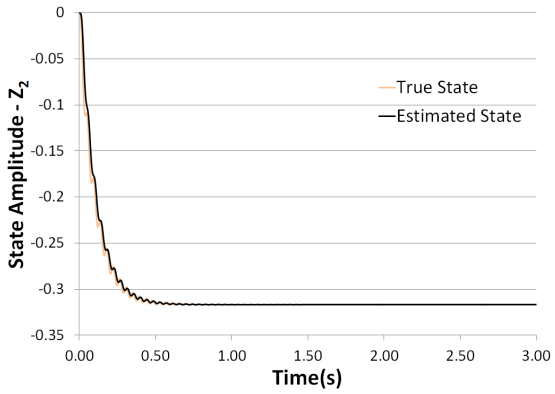


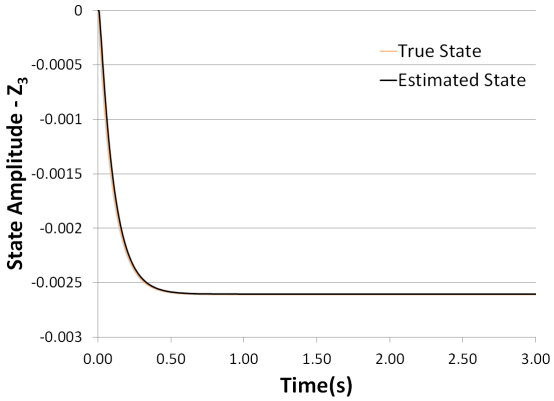
Figure 76: Simulated Payload Estimate



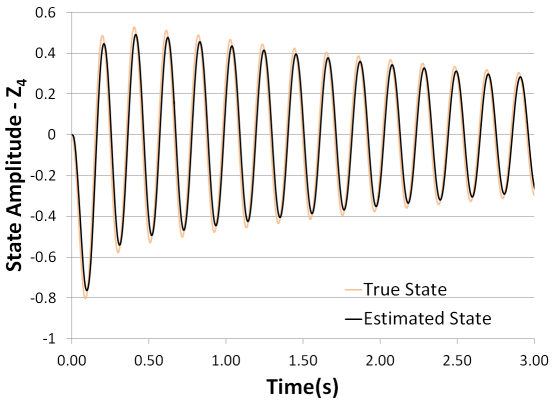
(a) State - z_1



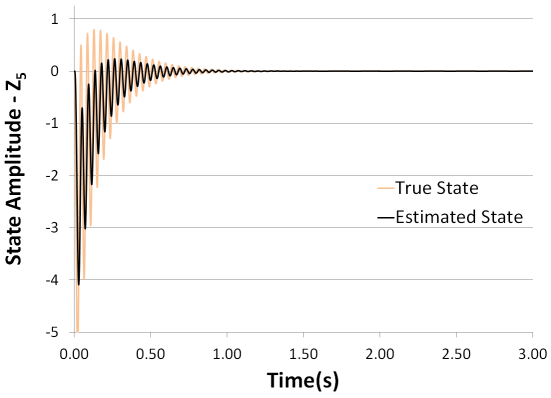
(b) State z_2



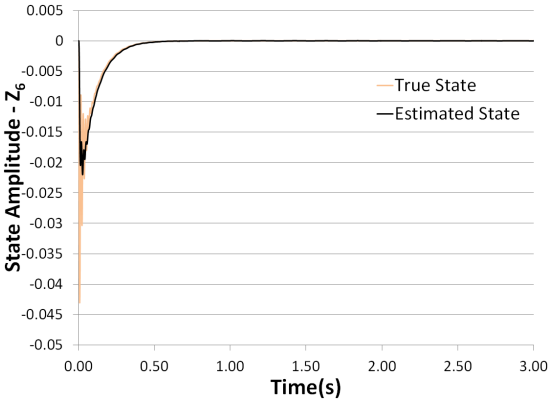
(c) State z_3



(d) State z_4



(e) State z_5



(f) State z_6

Figure 77: Simulation Results of Switching MMAE Without Measurement Noise

When measurement noise is unaccounted for in the design of the estimator, the effect will often corrupt the estimator weights resulting in poor parameter choices (Figure 78). Increasing the observer gain only exacerbates this effect by decreasing the signal to noise ratio.

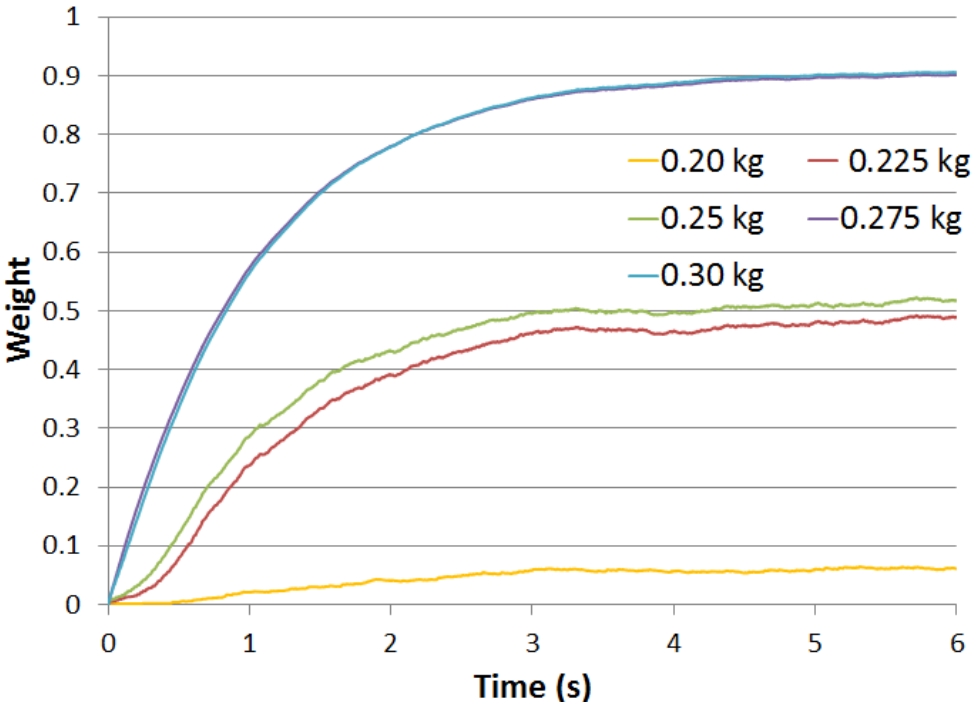
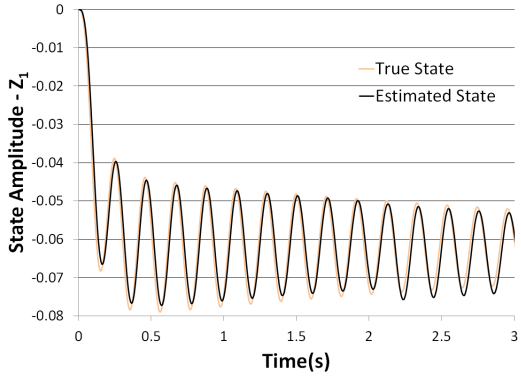
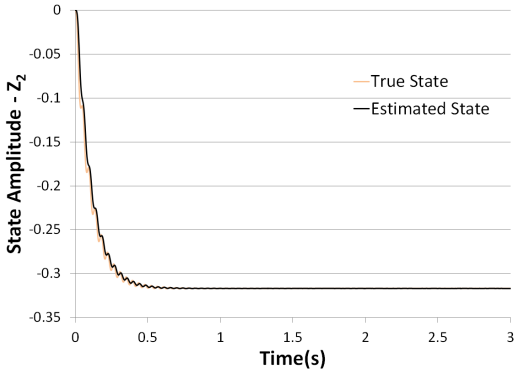


Figure 78: Weights With Simulated Measurement Noise (white noise)

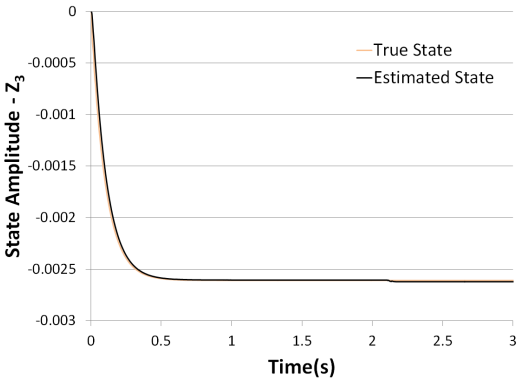
Figures 79(a)-79(f) illustrate the individual state estimates produced by the MMSAE. Note in Figure 79(a), slightly after 2 seconds, the state estimate switches from a slight over estimation to a slight under estimation of the true state as the MMSAE changes estimates from the sub-estimator with 0.275kg as its payload to the sub-estimator with the 0.30kg payload (see illustration in Figure 80). For this system, this error would not be enough of a perturbation to result in performance degradation, but if the estimators were significantly disparate, control effort resulting from the discontinuity of the state estimates could degrade system performance and result in instability. The conditions on closed loop stability are discussed further in section 6.7.



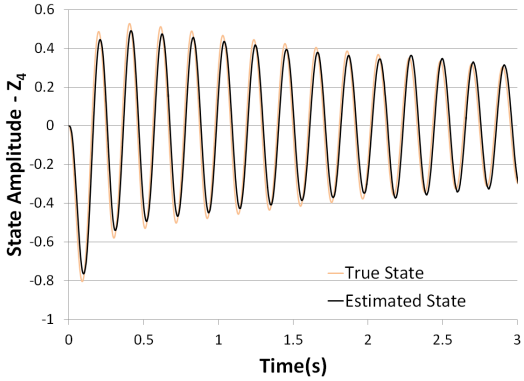
(a) State - z_1



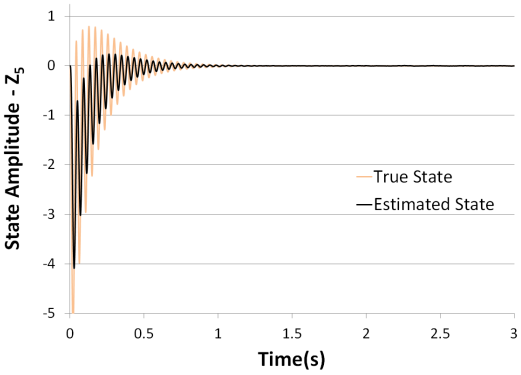
(b) State z_2



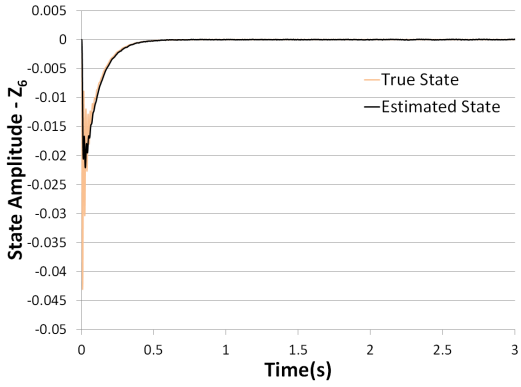
(c) State z_3



(d) State z_4



(e) State z_5



(f) State z_6

Figure 79: Simulation Results of Switching MMSAE With Measurement Noise

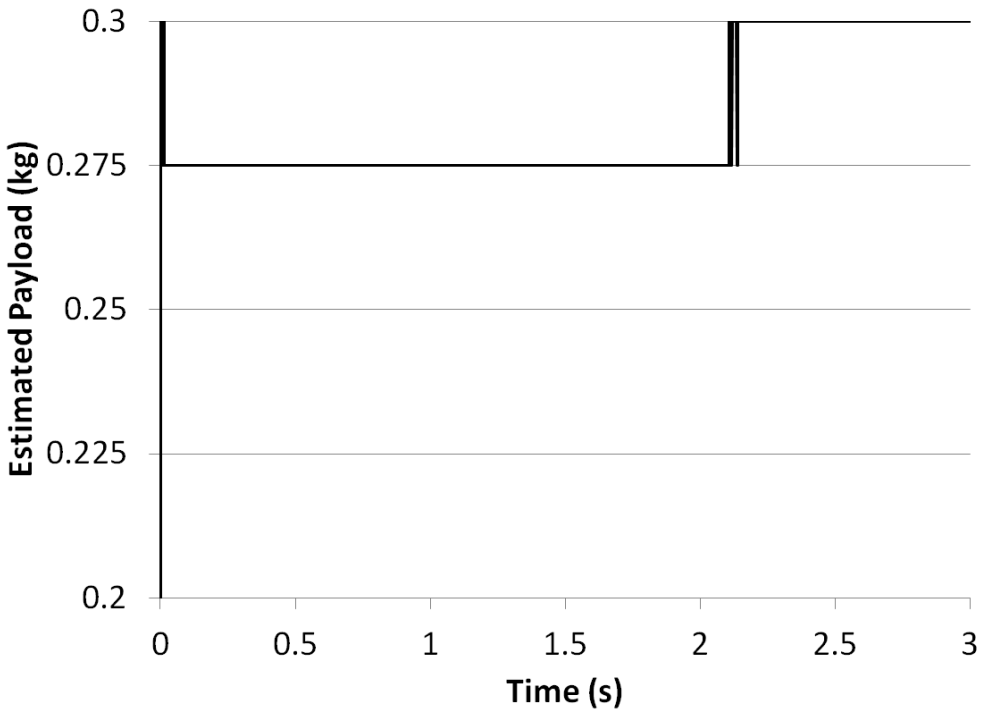


Figure 80: Payload Estimate Under Conditions of Measurement Noise

For the sake of completeness, the simulation results for a MMSAE with uncertain cart damping coefficient are below. Note that the parameters are identical to those in Table 10 from the preceding simulations with the exception that the true cart damping is 2000Ns/m rather than 100Ns/m (correction from recent experimental model verification).

Figure 81 shows the cart damping approximation produced by the multiple model adaptive estimator given the weights and parameter range demonstrated in Figure 82. The trends observed demonstrate similar characteristics as those for the payload estimates and indicate the applicability of the MMSAE for any structured parametric error.

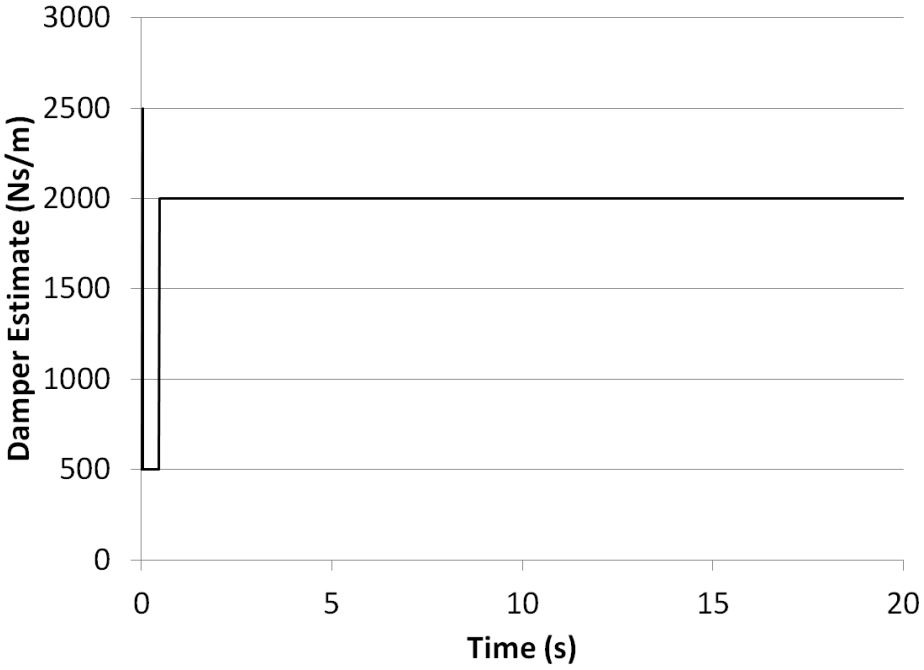


Figure 81: Cart Damping Estimate

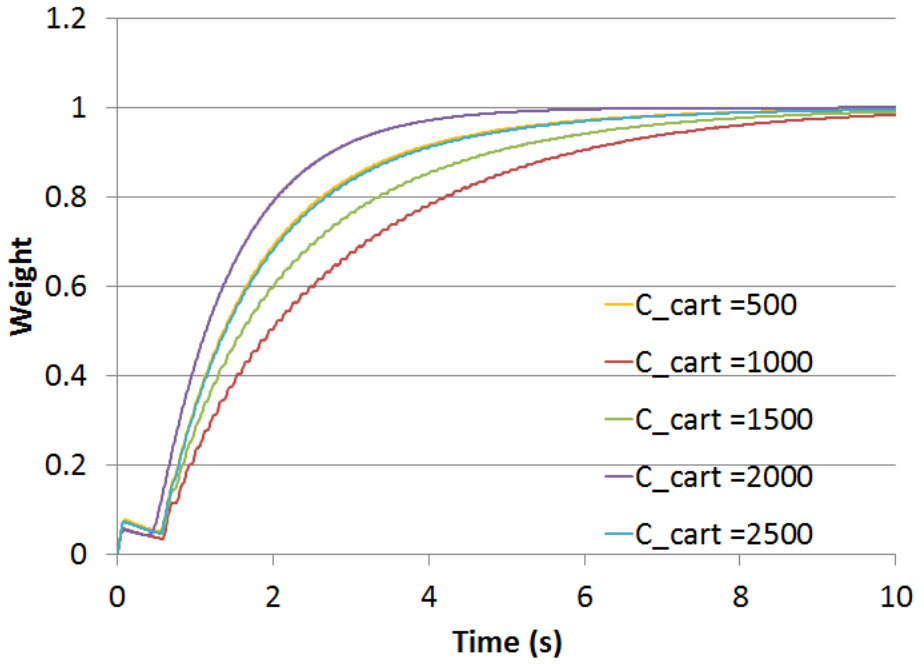


Figure 82: Weights for Cart Damping MMSAE

6.5 Multiple Model Weighted Averaging Adaptive Estimator (MMWAAE)

Often the true parameters vary slowly with time, or the discontinuity associated with the MMSAE approach results in exaggerated control signals. In these cases it may be better to avoid switching between the sub estimators as a parameter changes, and instead implement a multiple model weighted averaging adaptive estimator. The MMWAAE is designed such that the state estimate is a linear combination of all of the estimates from the sub estimators. Individual contributions are judged on the associated weights, with those receiving high weights being most important. In that manner the resultant estimated state vector is an interpolation of the sparse estimator structure allowing for the utilization of intermediate parameter values. The structure of the MMWAAE is shown in Figure 83. Care must be taken however to ensure that the weight magnitudes are significantly disparate

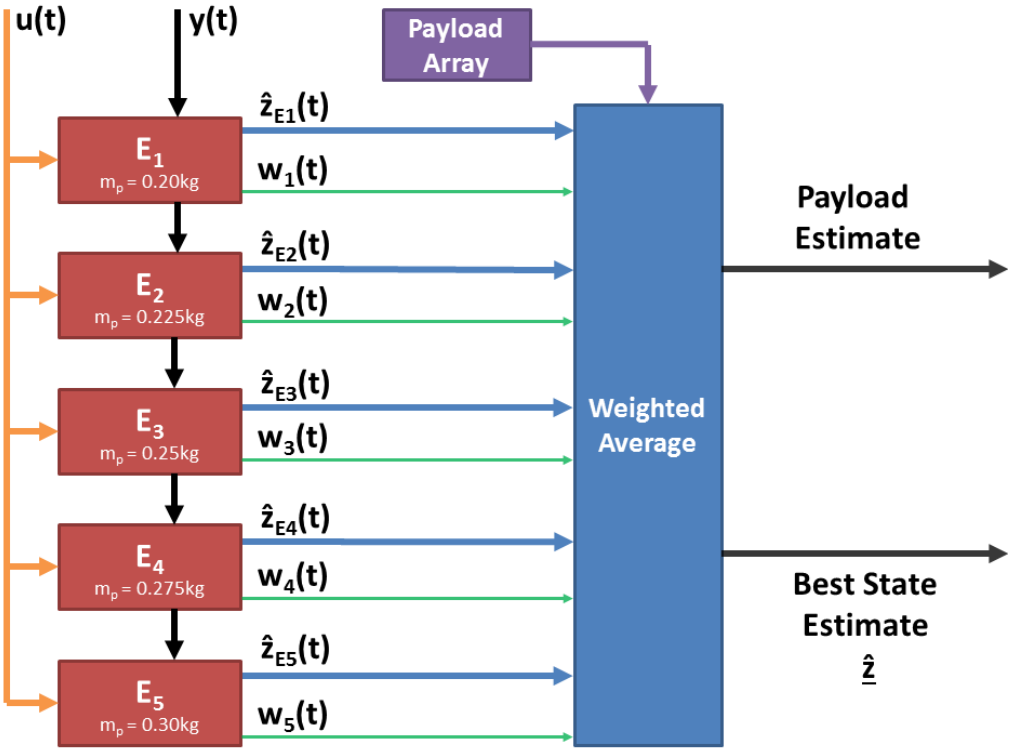


Figure 83: Multiple Model Weighted Averaging Adaptive Estimator

through conditioning of the weighting function. This requires a large enough spread such that the predicted state is not simply an average of the state vectors produced by each sub-estimator, but that the center of mass (mass in this case refers to the weighted summation

of parameter estimates) can be perturbed such that it tends toward the true parameter.

6.5.1 MMWAAE Simulation

Again utilizing the nominal parameters in Table 10, simulations of the FLASHMan testbed were performed using the MMWAAE to determine both the true state vector and the estimated unknown parameter. From Figure 84 that the weights produced by the MMWAAE are nearly identical to those produced by the MMSAE. However the estimated parameter value (Figure 85) no longer switches between provided values, but takes intermediary values between the provided model increments.

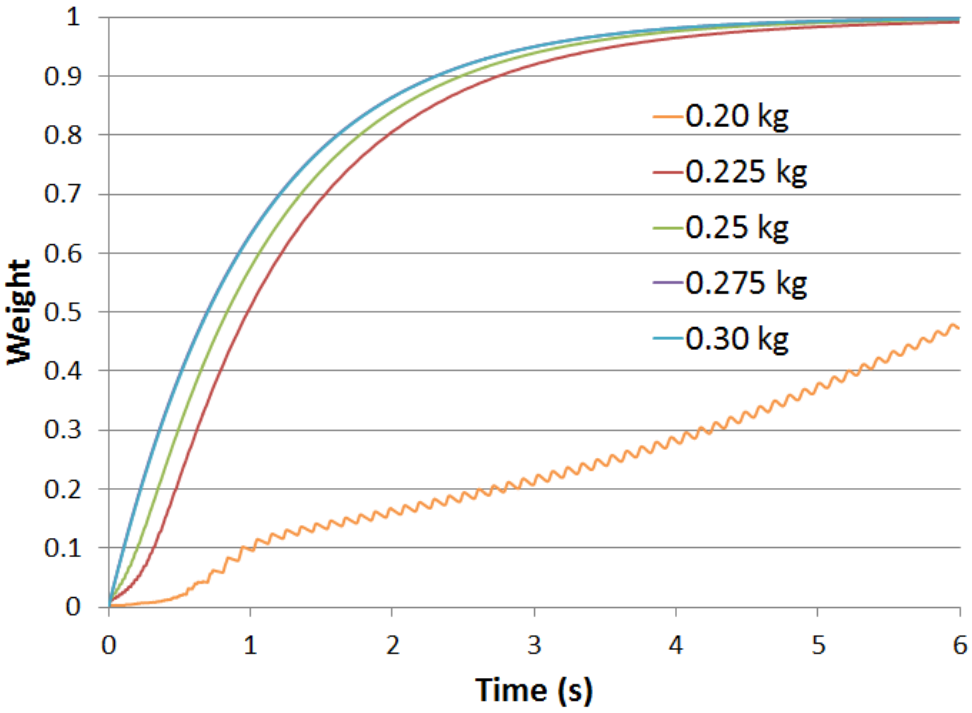


Figure 84: MMWAAE Filtered Weights Without Noise

As the vibrations subside, the weights approach 1 and each state estimate is weighted equally, pulling the estimated parameter value to the average of the specified candidates. This result illustrates the need for a persistence of excitation criterion for successful parameter identification. If the system is stationary, any of the models will produce accurate state estimates. It is only once the system is perturbed from its equilibrium state that any residual can be formed and used to identify the true parameter. Figures 86(a)-86(f) show

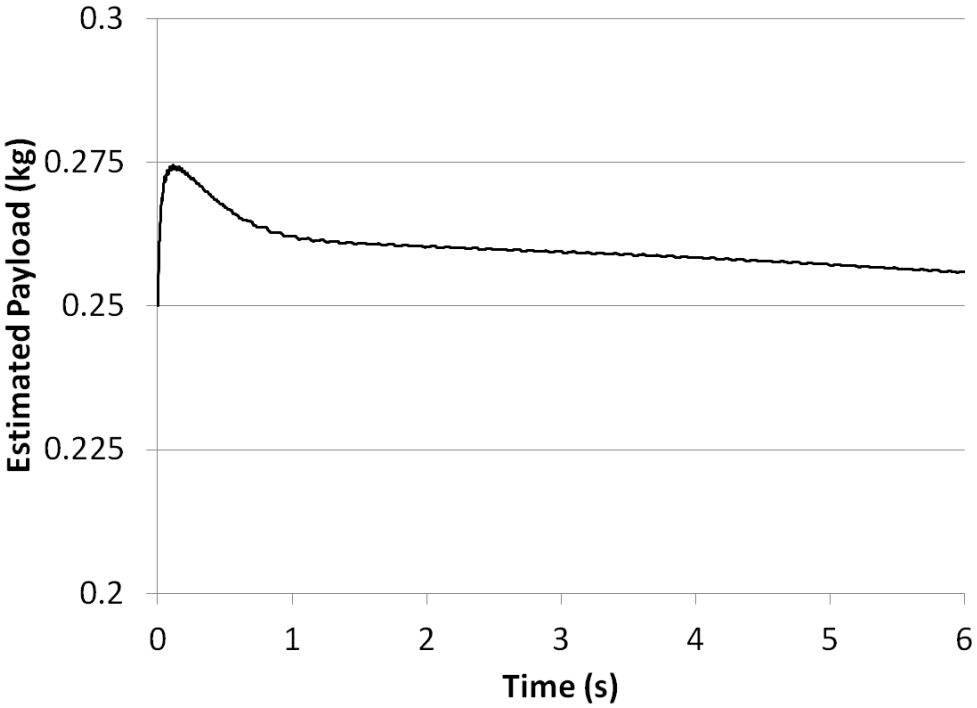
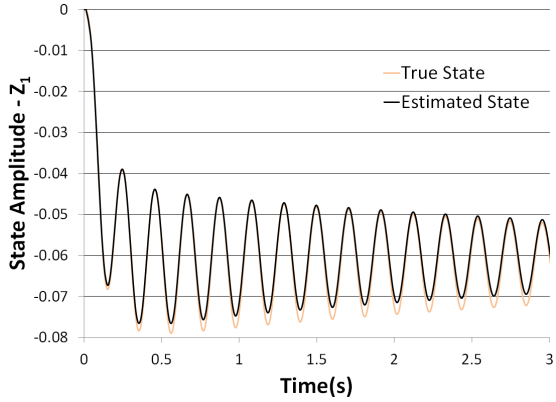
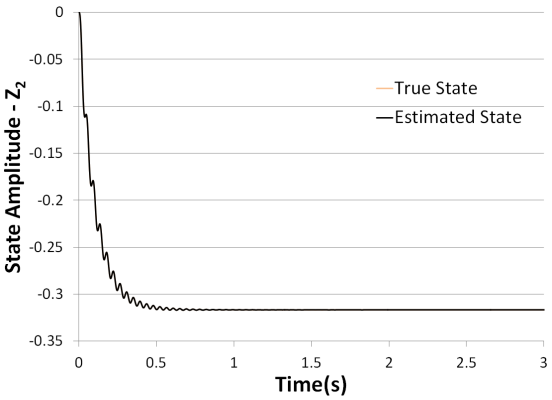


Figure 85: MMWAAE Simulated Payload Estimate

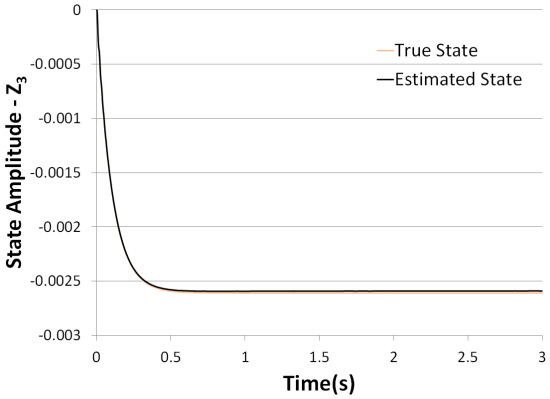
the state estimates produced by the MMWAAE. Note that the initial state estimates are relatively accurate, but as the system converges to its equilibrium state, the estimate error grows to a constant value.



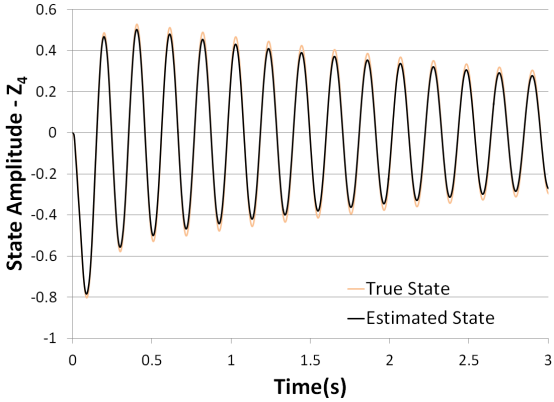
(a) State - z_1



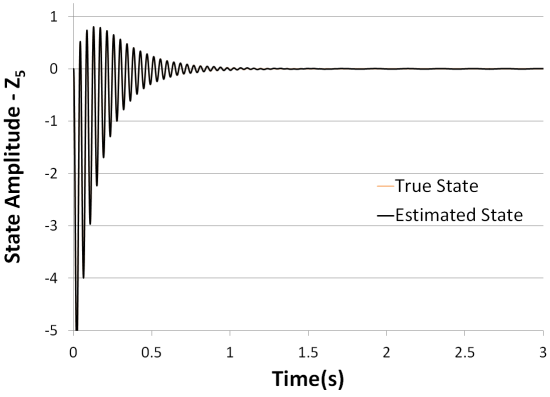
(b) State z_2



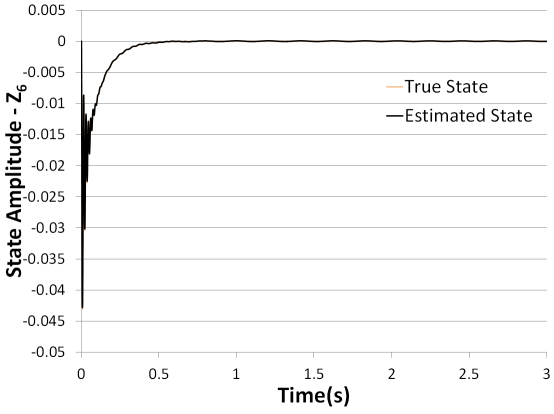
(c) State z_3



(d) State z_4



(e) State z_5



(f) State z_6

Figure 86: Simulation Results of Weighted Averaging MMAE Without Measurement Noise

The addition of noise in the acceleration measurements serves as a small excitation to the closed loop estimators, resulting in a convergence to the neighborhood of the true value rather than the average value of the parameter candidates, as demonstrated in Figure 87. Essentially eliminating very poor candidates, convergence occurs, not to the true parameter, but to a weighted average of the best candidates for the true parameter.

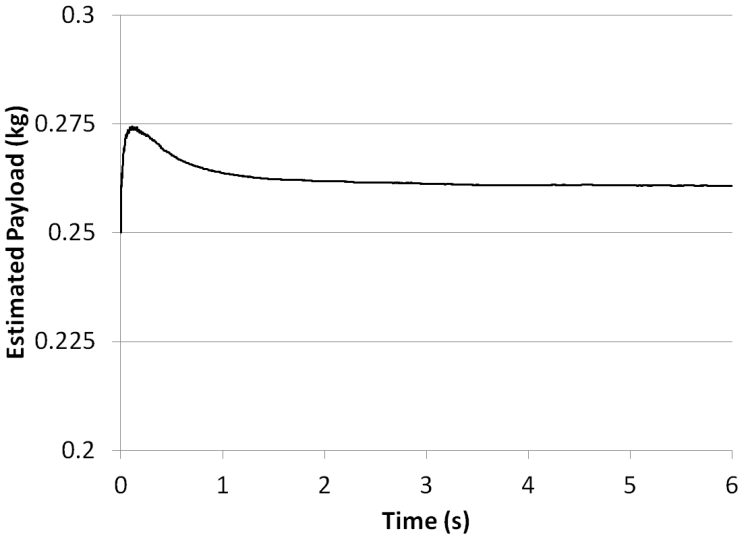


Figure 87: Payload Estimate Under Conditions of Measurement Noise

While the parameter estimate is a useful byproduct of the multiple model adaptive estimator, the desired results are accurate state estimates. As the system converges to its steady state value incorrect estimators will be chosen on occasion. However, they are chosen because they provide the state estimate with the least state error at any given time, (i.e. the sub-estimator with the best model approximation does not necessarily produce the best state estimates for all possible operating scenarios). For example, when the link is in equilibrium and no longer vibrating, the measurement noise dominates the measurement signal. If the ideal sub-estimator gains are larger than a less ideal sub-estimator, it will respond more aggressively to the noisy error signals, increasing δ_z and lowering its weight in the selection criteria. In this case it is actually better to choose a less accurate sub-estimator with a lower gains as the state estimates are smoother and more accurate. However, given a disturbance from equilibrium, the MMAE must still respond quickly to recover the best state estimate, (i.e. the selection criteria should be tuned as well as the observer gains to

achieve the desired performance over the entire operational range).

6.5.2 Output Weighted Multiple Model Estimators

For the purposes of computational streamlining, the state difference weighting metric can be replaced with an output weighting strategy. The benefit is that in comparing output estimates rather than the state predictions, the measurement signals can be used to replace one of the state estimators in the sub-estimator, provided that there are at least n measurements where n is the number of system modes. However the measurements need not be the signals which were used to close the loop on the estimator. The difference between a measured set of outputs and an estimated set is

$$\begin{aligned} y_o - \hat{y}_o(t) &= (C_o + \Delta C_o)z + (D_o + \Delta D_o)u - C_o\hat{z} - D_o u \\ &= C_o e + \Delta C_o z + \Delta D_o u \end{aligned} \tag{293}$$

where y_o is the measured output vector and $\hat{y}_o$ estimated output vector. Applying the equilibrium assumptions described in section 6.2, i.e.

$$e_* = -(A - LC_s)^{-1}\left[(\Delta A - L\Delta C_s)z + (\Delta B - L\Delta D_s)u\right] \tag{294}$$

then the difference between the measured and estimated outputs is

$$\delta_{y_o}(t) = -C_o(A - LC_s)^{-1}\left[(\Delta A - L\Delta C_s)z + (\Delta B - L\Delta D_s)u\right] + \Delta C_o z + \Delta D_o u \tag{295}$$

$$\begin{aligned} &= \left(\Delta C_o - C_0(A - LC_s)^{-1}\Delta A + C_0(A - LC_s)^{-1}L\Delta C_s\right) z \\ &+ \left(\Delta D_o - C_0(A - LC_s)^{-1}\Delta B + C_0(A - LC_s)^{-1}L\Delta D_s\right) u \end{aligned} \tag{296}$$

.

Let β_o be the basis function vector for the chosen output measurement (i.e. β_o =

$\begin{bmatrix} 1 & \psi_1(x_o) & \dots & \psi_{n-1}(x_o) \end{bmatrix}$) and β_s be the basis function vector for the sensor placements utilized by the observer. Then

$$\delta_{y_o}(t) = \left(\left[\beta_o + C_0(A - LC_s)^{-1}L\beta_s\right]\Phi - C_0(A - LC_s)^{-1}\begin{bmatrix} 0_{(n\times n)} \\ I_{(n\times n)} \end{bmatrix}\right)\begin{bmatrix} -\Delta\omega^2 & -\Delta\hat{C} \end{bmatrix} z(t) + \left(\left[\beta_o + C_0(A - LC_s)^{-1}L\beta_s\right]\Phi - C_0(A - LC_s)^{-1}\begin{bmatrix} 0_{(n\times n)} \\ I_{(n\times n)} \end{bmatrix}\right)\left[\Delta\hat{Q}\right] u(t) \tag{297}$$

and therefore, as before, the unknown parametric error quantity γ is described as a function of known system quantities according to (298).

$$\delta_{y_o}(t) = \left(\left[\beta_o + C_0(A - LC_s)^{-1}L\beta_s\right]\Phi - C_0(A - LC_s)^{-1}\begin{bmatrix} 0_{(n\times n)} \\ I_{(n\times n)} \end{bmatrix}\right)\gamma(t) \tag{298}$$

6.5.2.1 High vs. Low Gain State Observers for Multiple Model Weight Determination

Beyond simple noise rejection/amplification associated with the observer gain, the residual error between the measured and estimated output is largely dependent on the magnitude of the attributed observer gain. Large gains result in smaller error signals on which the output difference weights are determined and the lower signal to noise ratios, leading to inaccurate weight assignment. Very low gains (i.e. low eigenvalues of the observer system) prevent phase synchronization of the oscillatory estimates and true system, meaning that the zero crossings $\gamma = 0$ for each sub-estimator will occur asynchronously. This is in contrast with Figure 74, and results in weights attributed to the current estimator state and not the parametric error. Likewise, small estimator gains also invalidate the quasi-static assumption made in the state difference theorem by reducing the convergence rate of the estimator to the same order of magnitude as the system itself. Care must be taken when choosing an estimator gain for determining the residual errors δ_z and δ_{y_o} which are neither

too large or too small. For application on FLASHMan, the residual observer's poles are placed ≈ 5 times faster than the fastest system poles.

Similar to two model techniques applied in fault detection [41] a High-Low gain sub estimator can be used to determine both the residual output error for weight generation and the estimated state for control. The high gain estimator determines the state estimate, and the low gain observer is used to calculate the reliability of the state estimates, (i.e. how accurate the model is).

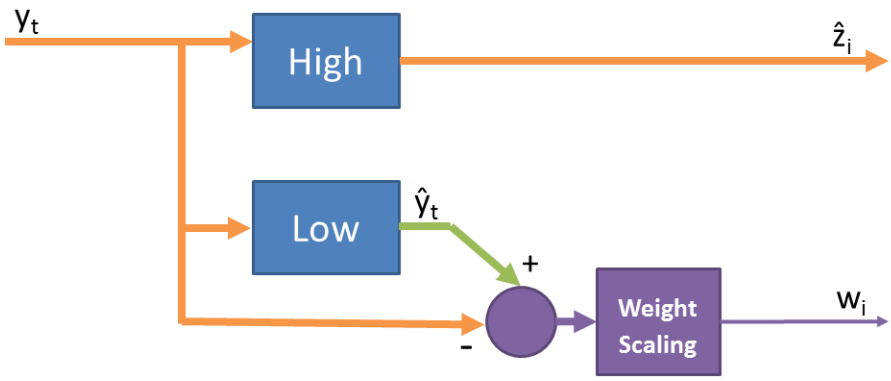


Figure 88: High + Low Gain Sub-estimator

6.6 FLASHMAN Experimental Results - Payload Identification in Multi-Waypoint Move Cycles

To test the efficacy of the multiple model for identification of true model parameters and flexible system states, a multiple model estimator was designed and implemented utilizing the output difference weighting criterion for a high/low gain sub-estimator and the switching form of the MMAE.

6.6.1 Experiment Preface - Definitions and Apparatus

Implementation on the real time system used for the experiments, required significant simplification of the methods used in the preceding simulations. First, the output difference multiple sensor strategy was employed rather than the state difference multiple estimator, single sensor strategy. This effectively eliminates one of the observers in the sub-estimator and allows the full five member estimator bank to be evaluated in real time at 1kHz. Second, using three accelerometers, weights are defined as the RMS value of the norm

of the output error, $y - \hat{y}$, rather than functions of the state estimates of two separate estimators. Third, these measurements form a common basis for comparison. Therefore, as a result of the common measurements between all estimators and because the estimators were tuned similarly, the need for the output difference scaling matrices was reduced and they were subsequently eliminated. Fourth, because the weights are based on the norm of the output estimate error, low values indicate higher accuracy. This is contrary to the weighting functions in Section 6.3 where high values indicated good estimator performance. Therefore, the selection algorithm chooses the state vector corresponding to the sub-estimator with the lowest weight. Finally, high/low gain sub-estimators are utilized. A low gain estimator is used to determine the weight associated with the state vector estimate produced by the high gain estimator.

The multiple model switching adaptive estimator control system was implemented on a LabView RealTime quad core desktop computer. Each core was responsible for separate parts of the control system, one for command generation, data acquisition, and the graphical user interface, one for feedback control, one for the state estimation routine, and the final core was reserved for data capture and storage.

A gripper was designed for the FLASHMan testbed to permit evaluation of the proposed estimation strategy resulting in a nominal end of arm mass of 0.36kg. Payloads of varying mass were also fabricated as shown in Figure 89.

As a baseline for comparison, a move cycle was designed for a pick and place task. As demonstrated in Figure 90 the manipulator moves to a desired location where it picks up a payload and after performing a series of motions, drops the payload off at a target location. All moves are trapezoidal velocity profiles with a maximum velocity of $3\frac{\mathrm{m}}{\mathrm{s}}$. And after each move, the manipulator pauses until the vibration reaches a tolerable level measured by the acceleration at the end of arm, $RMS(\ddot{y}_{tip}) < 0.5\frac{\mathrm{m}}{\mathrm{s}^2}$.

Different move lengths and starting positions were used to eliminate the likelihood of inadvertent command shaping and any asymmetries in the manipulator workspace, including nonlinear effects from track friction, changing belt length, or other sources. A list of these moves and their corresponding actions is listed in Table 13.

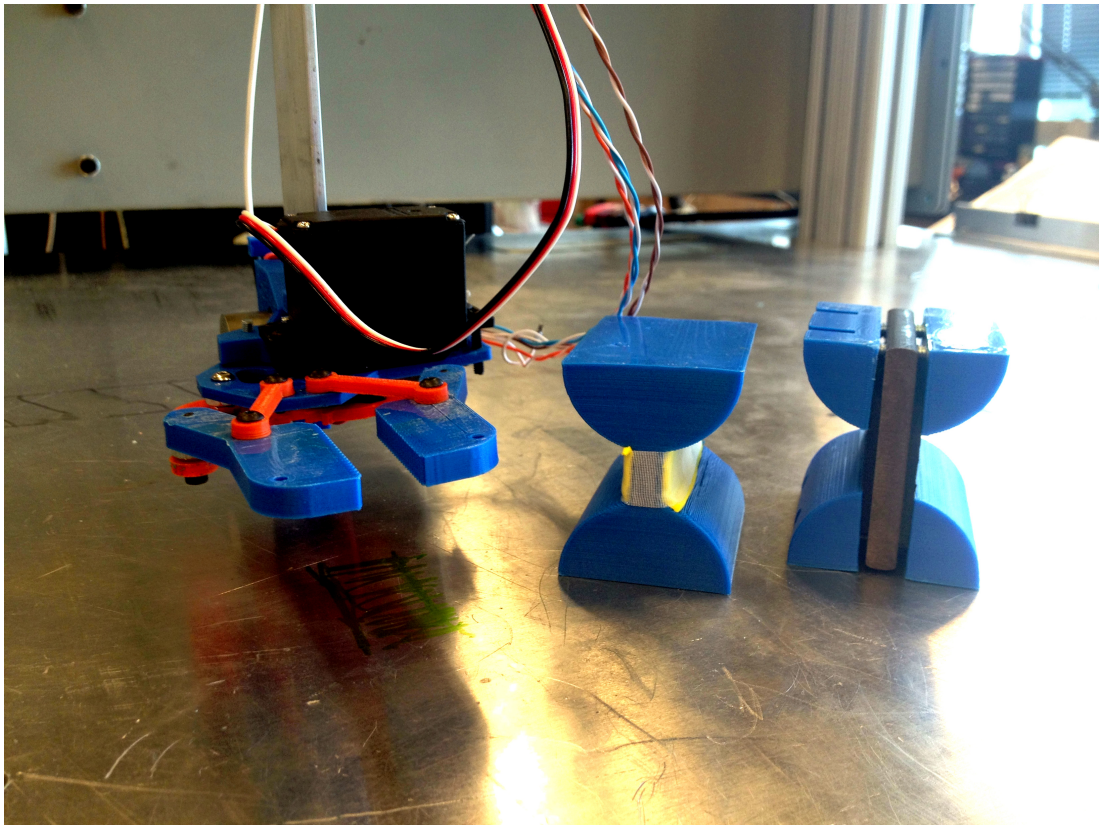


Figure 89: Gripper and Payloads (0.059kg (left) 0.143kg (right))

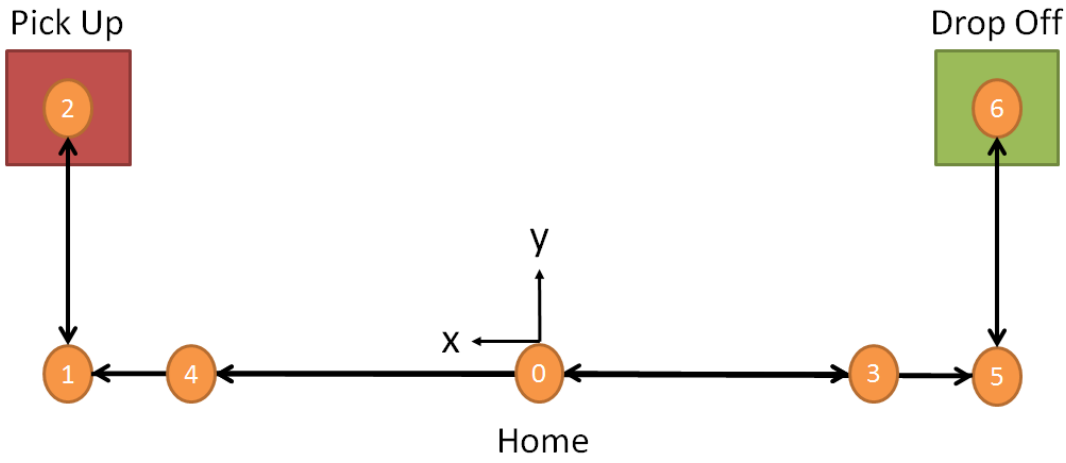


Figure 90: Multiple Waypoint Move Cycle

Table 13: Cycle Move List and Descriptions

Move #	Start Location	End Location	Move Length	Task
#1	0	1	0.20m	Wait until $RMS(\ddot{y}_{tip}) < \ddot{y}_{tol}$)
#2	1	2	0.20m	Pick up payload
#3	2	1	0.20m	Wait until $RMS(\ddot{y}_{tip}) < \ddot{y}_{tol}$)
#4	1	3	0.35m	Wait until $RMS(\ddot{y}_{tip}) < \ddot{y}_{tol}$)
#5	3	4	0.30m	Wait until $RMS(\ddot{y}_{tip}) < \ddot{y}_{tol}$)
#6	4	5	0.35m	Wait until $RMS(\ddot{y}_{tip}) < \ddot{y}_{tol}$)
#7	5	6	0.20m	Drop off payload
#8	6	5	0.20m	Wait until $RMS(\ddot{y}_{tip}) < \ddot{y}_{tol}$)
#9	5	0	0.20m	Wait until $RMS(\ddot{y}_{tip}) < \ddot{y}_{tol}$)

6.6.2 MMSAE Experimental Results

The first set of experiments was aimed at evaluating the ability of the MMAE to determine the approximate parameter value without closing the state space loop. A PID control loop around the cart position was used to drive the system through the pre-programmed way-points, and the estimator was used to verify the payload mass and corresponding system states. Five payload candidates at even increments of 0.05kg were selected to fully capture the expected parameter range (0.36kg - 0.51kg). Figure 91 shows the payload estimates for a move cycle in which no payload is picked up. The corresponding weights are shown in Figure 92.

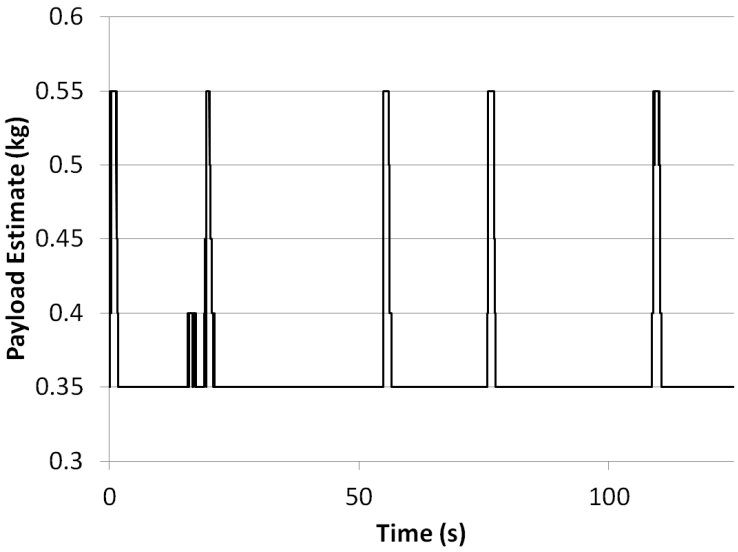


Figure 91: Payload Estimate (0.36kg + no additional payload)

Because no payload is picked up, for each motion the best weight is assigned to the estimate from the m_t = 0.35kg estimator. The sharp jumps in Figure 92 indicate motion from one station to another where the control effort injected instantaneously drives the estimated acceleration to a large value. The measurements of the acceleration, because of the filtering used, follow, but delayed slightly, resulting in large output errors.

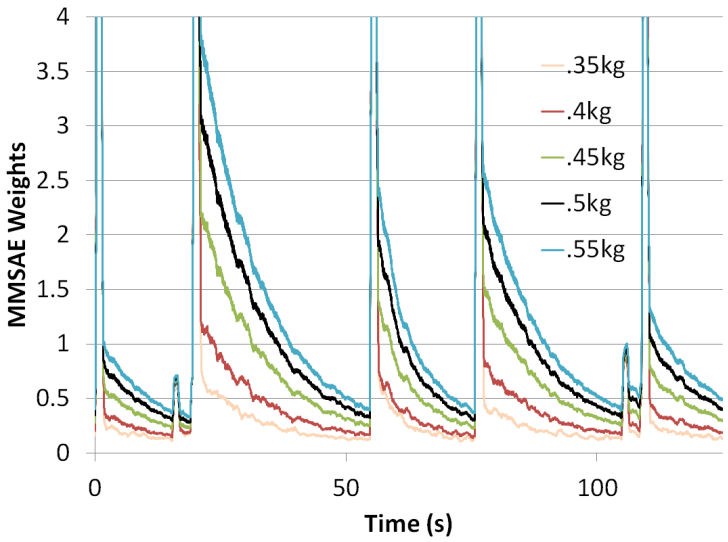


Figure 92: Estimate Weights (0.36kg + no additional payload)

Picking up a 0.059kg payload, Figure 93 illustrates the switch in the multiple model adaptive estimator at Station 2 when the payload is grasped. When the payload is released at Station 6, the corresponding parameter estimate drops back to the nominal gripper mass.

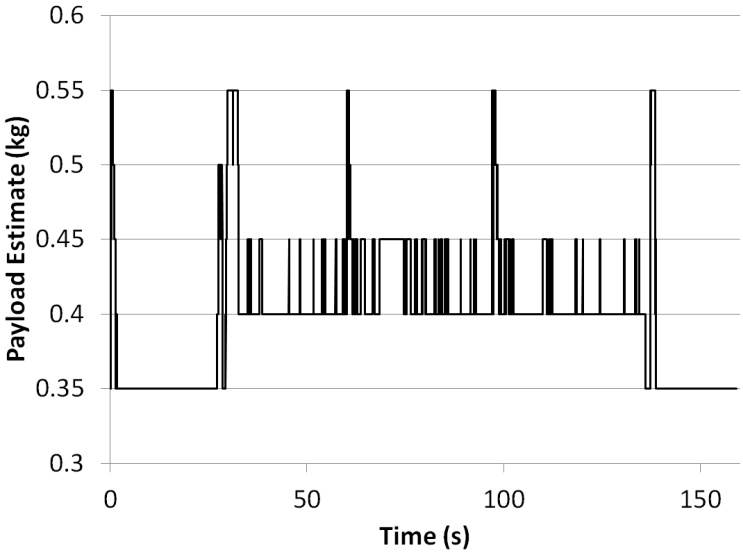


Figure 93: Payload Estimate (0.36kg + 0.059kg payload)

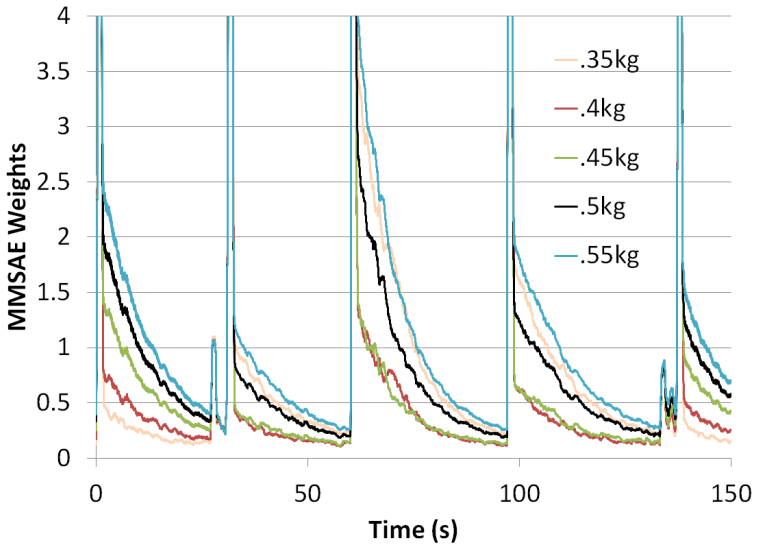


Figure 94: Estimate Weights (0.36kg + 0.059kg payload)

Finally the test was repeated with a larger 0.143kg payload, the results of which are shown in Figures 95 and 96

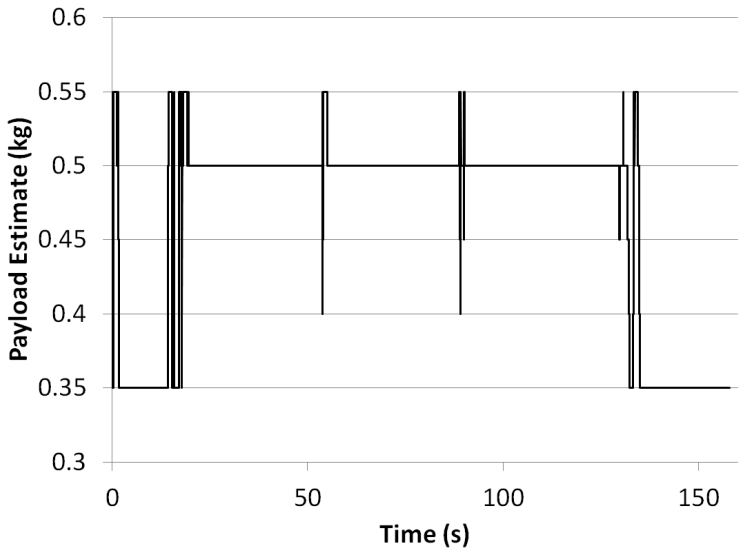


Figure 95: Payload Estimate (0.36kg + 0.143kg payload)

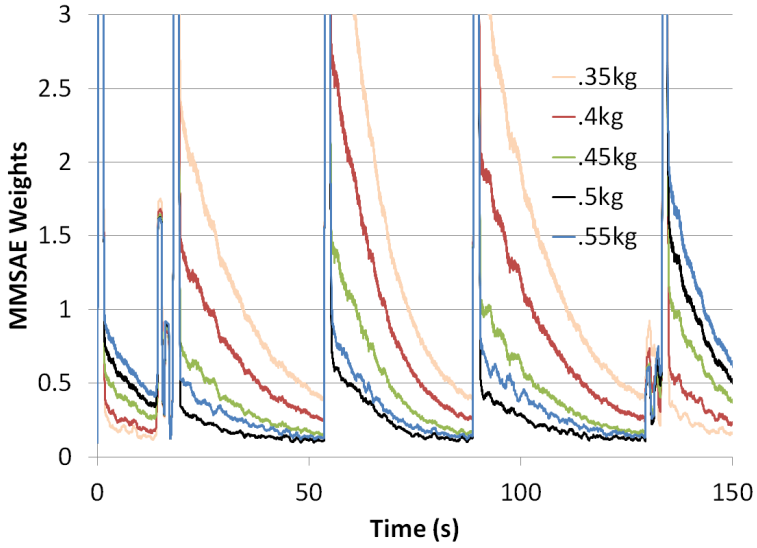


Figure 96: Estimate Weights (0.36kg + 0.143kg payload)

Again the true parameter value m_t = 0.503kg is approximated by the closest value, 0.5kg from Station 2 when the payload is grasped to Station 6 where it is dropped off, elsewhere the estimator which utilizes the nominal end of arm mass receives the best weight.

From the results it is clear that the MMSAE structure is capable of accurately determining the true system parameter and select the appropriate sub-estimator for the current operating regime. At a minimum, this forms the basis of an open loop parameter identification scheme for tuning the state space control systems of flexible manipulators. However, the desired goal is to use the produced state information to close the loop around the flexible state variables and reject residual vibration and disturbance inputs. In the following sections a closed loop, state space control system is designed and implemented to satisfy this objective.

6.7 Closing the Loop - Multiple Models for Parametrization of the Operating Regions of a Dynamic Control System

The incorporation of control based on state estimates produced from the multiple model adaptive estimator can be approached as variations of gain scheduling type approaches. From [73] the operating range of a complex system can be decomposed into a number of "operating regimes" (Figure 97) for which local controllers/observers are developed. While not necessarily limited to identical control structures, the analysis is simplified if each

control system shares the same state space. Furthermore, the finer the graduation between parametrized models, the more accurate the result will be [73].

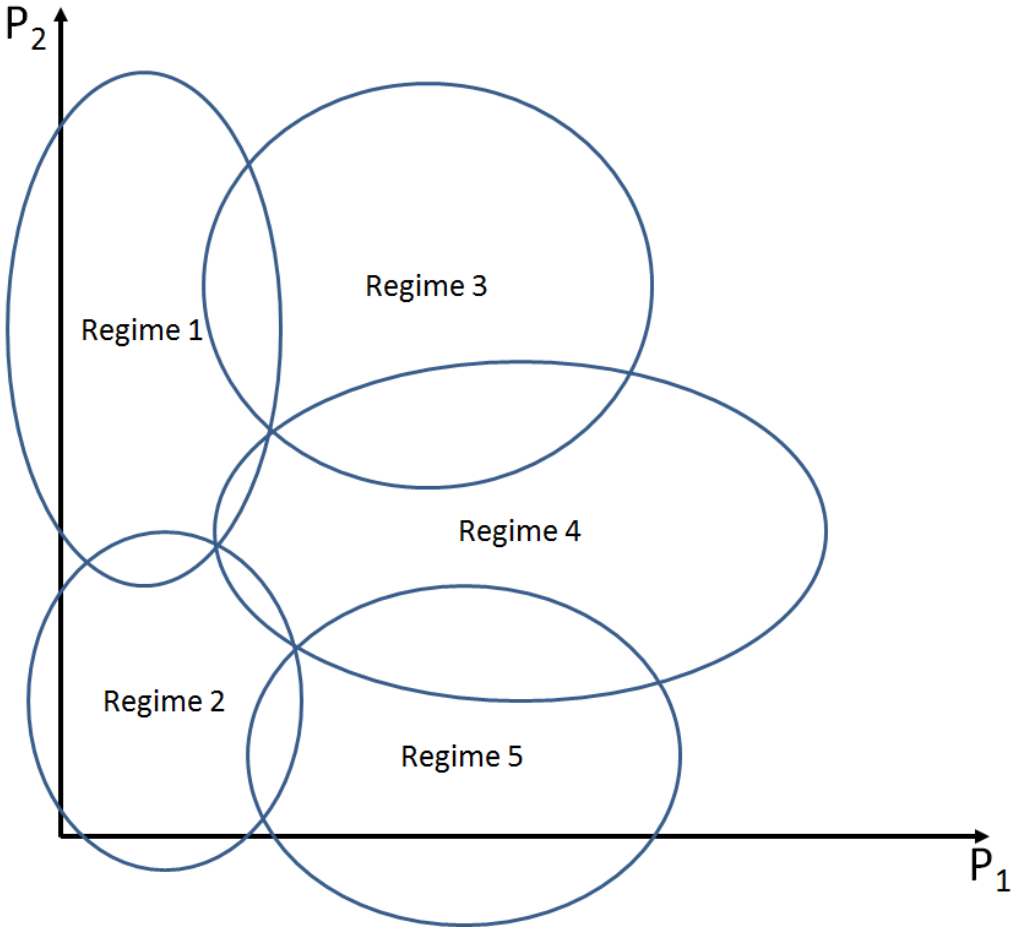


Figure 97: Decomposition of Full Operating Range into Operating Regimes about the Operating Point and Gain Scheduling Type Control

From [30,73], moving from one local control regime to another is a form of gain scheduling procedure and can be analyzed as a nonlinear system which is described by a weighted combination of linear subsystems.

$$z_{k+1} = f(z_k, u_k) = \sum_{i=1}^{N} \left(z_i + A_{di}(z_k - z_i) + B_{di}(u_k - u_i) \right) w_i(z_k) \tag{299}$$

where

$$u_k = u_i + \sum_{i=1}^{N} K_i(z_k - z_i) w_i(z_t) \tag{300}$$

and w_i is defined by arbitrary positive definite functions such that $\sum_{i=1}^{N} w_i z_k = 1$. Then for a slowly varying scheduling variable (i.e. parameter estimate) relative to the closed loop bandwidth, global stability can be asserted by designing each local controller to have poles inside the unit circle [73, 84]. The result is that by ensuring a fine enough graduation of parameter estimates around the expected working range, global stability can be inferred from local linear analysis. The proof of which is replicated from [30] in Appendix A.

6.8 Closed Loop State Space Control of FLASHMan Using a Multiple Model Estimation Strategy

The proposed multiple model control structure consists of independent linear quadratic regulator (LQR) gains with penalties on cart position and tip acceleration designed for each of the candidate parameter values. Gains are selected using the parameter estimates from the MMSAE and the states produced are used to generate the control effort. This procedure is illustrated by Figure 98 where the measured signals are the accelerometer readings, the control input is the desired cart position, and the set-point is the desired end effector location.

The control structure of the x-axis, therefore, is a gain scheduling type controller and subject to the local stability requirements outlined in section 6.7. Implementation occurs on a LabView real-time desktop with sampling rates of 1kHz. In the y-direction, PID control of the axis position is utilized since the corresponding flexible states are negligible.

6.8.1 Experimental Results

Closed loop performance and robustness can be assessed through multiple measures. Cycle time comparisons demonstrate the improvement in performance made by switching to the closed loop state space control form over PID control. As illustrated in Figure 99, the closed loop flexible state feedback control system based on the MMSAE observer reduces the cycle time by $\approx 90\%$.

Comparisons can also be drawn with static state space controllers/estimators designed with respect to an estimated value of the payload. Note that given the nominal tip mass (i.e. the mass of the gripper), designed control systems with estimated static payloads above

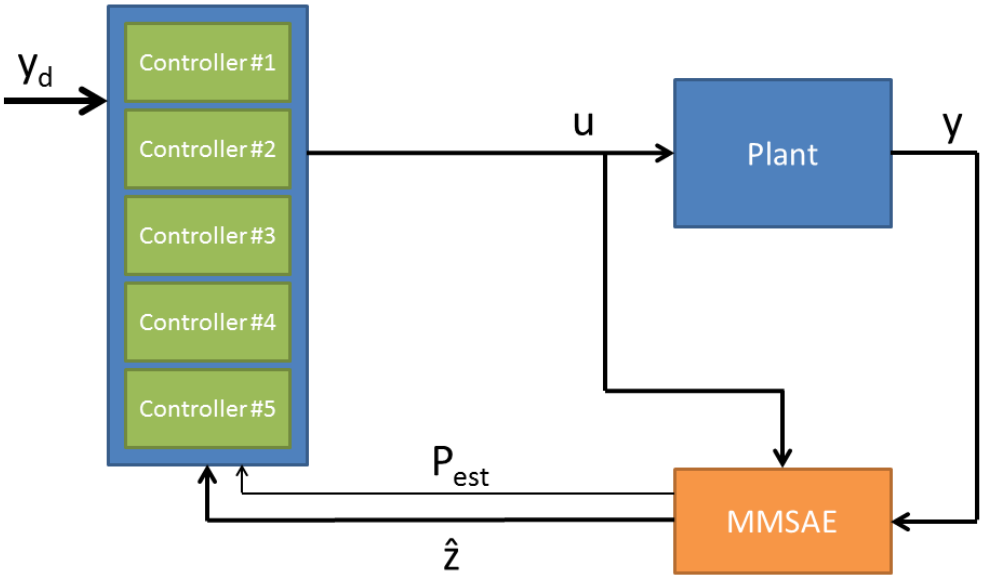


Figure 98: Control Structure for FLASHMan Control with MMSAE State Estimates

0.4kg are unstable for the chosen LQR penalties which are held constant for all designs. Therefore, for example, if the controller/estimator was designed using the 0.143kg payload, the control system would be unstable if the payload was removed. However, given the MMSAE based control system, only a range of potential payloads must be specified for stability to be assured. This reduces the potential for poor control design by essentially allowing the designer to identify a bound for the system parameter instead of a single static value. Conversely, if by chance the designer was to chose a nominal payload estimate of 0.25kg, the system would outperform the MMSAE based control system with respect to cycle time for this move cycle and LQR penalty set. However, the choice would be arbitrary and not necessarily robust to a wide range of parameter variation.

Comparing the variation in move cycle time for the manipulator with/without the 0.143kg payload (Figure 100) shows that the variability of cycle time is reduced with the MMSAE-based control system. Therefore, it can be demonstrated that the proposed control system best preserves the performance characteristics over the range of payload values.

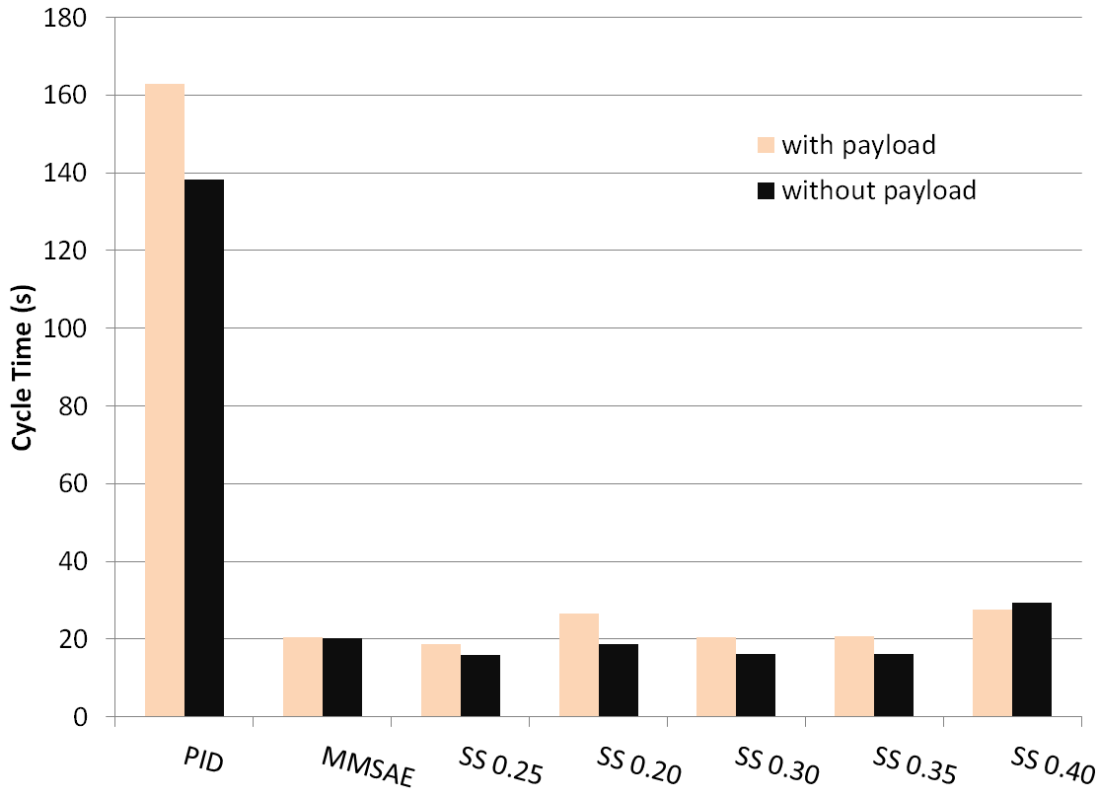


Figure 99: Cycle Time for Implemented Control Methods

The residual vibration from move to move is also reduced by ≈ 60% relative to the PID only control system. This result is demonstrated in Figure 102. These responses have been time normalized to demonstrate the relative magnitudes of the acceleration traces. However, the PID only cycle takes ten times longer to execute then the MMSAE cycle. the large accelerations indicate transitions from one station to another. The transitions, (i.e. the timing of the transitions) and residual vibration levels are far more consistent with the MMSAE approach for the three payload values relative to the PID controlled system. Thus, indicating that the changing system parameter value has limited effect on the closed loop system performance.

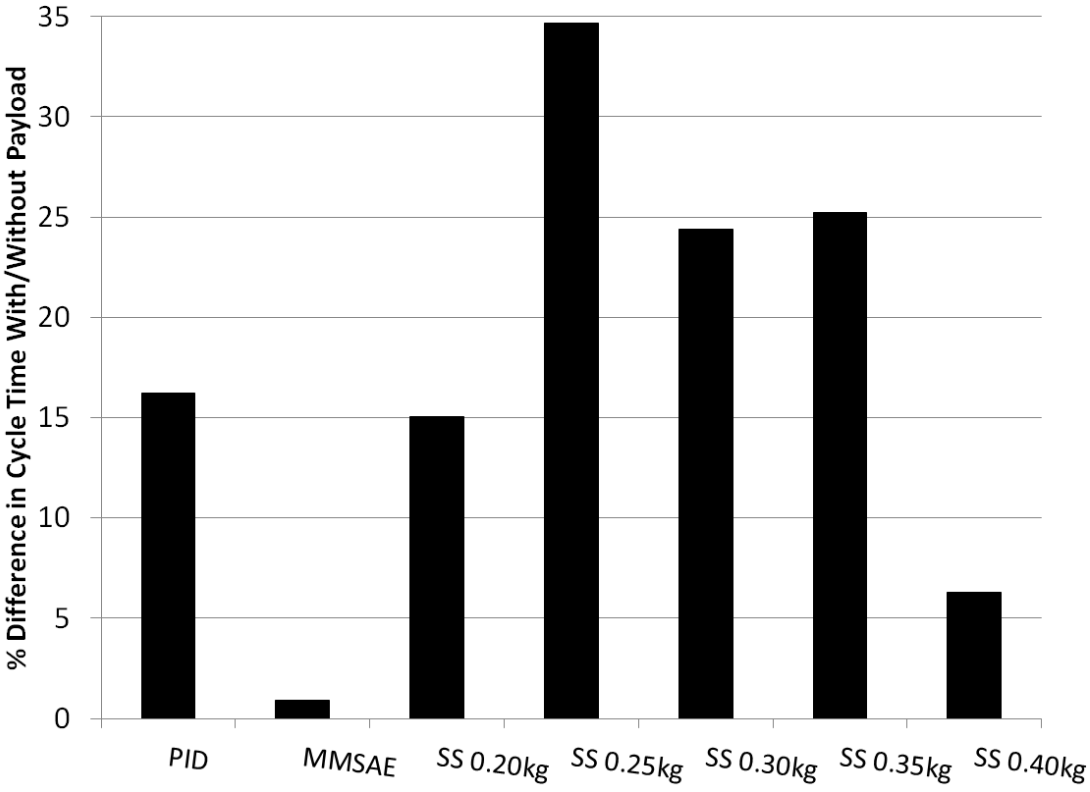
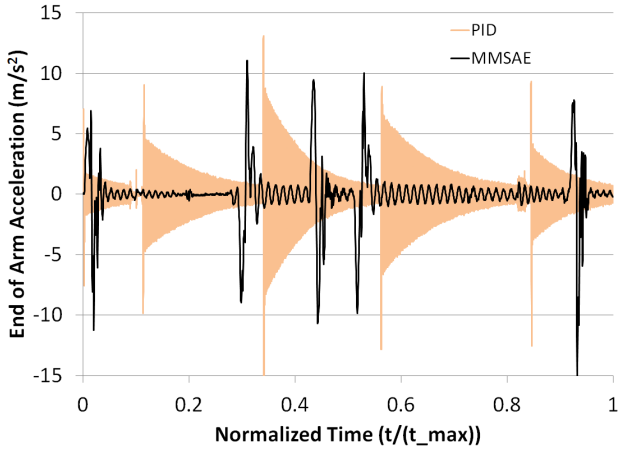
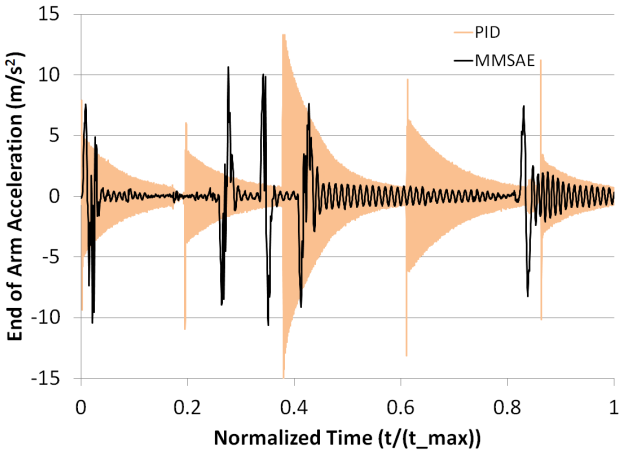


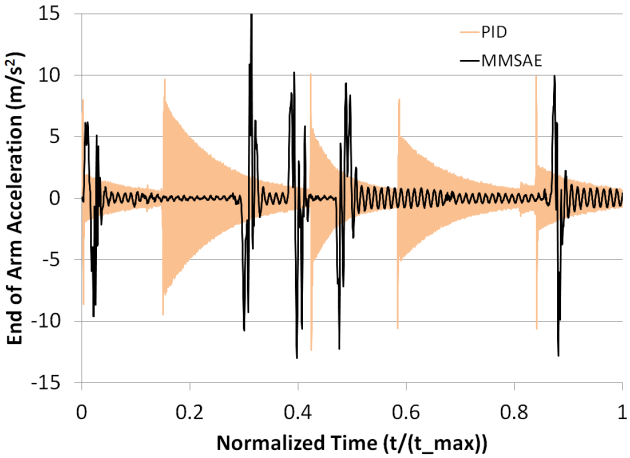
Figure 100: % Variation in Cycle Time for Implemented Control Methods



(a) Acceleration MMSAE vs. PID (0.36kg + no additional payload)



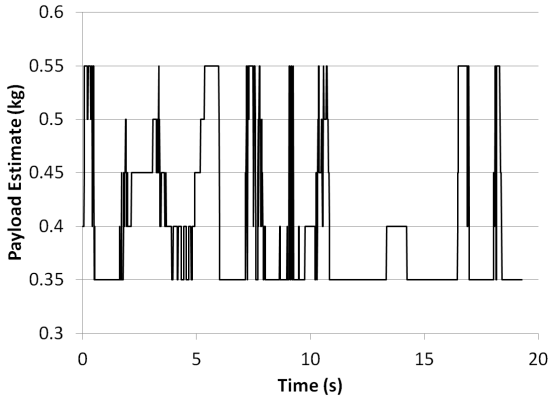
(b) Acceleration MMSAE vs. PID (0.36kg + 0.059kg payload)



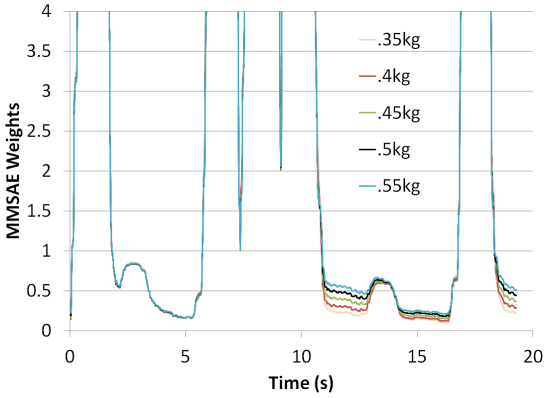
(c) Acceleration MMSAE vs. PID (0.36kg + 0.143kg payload)

Figure 101: Time Normalized End of Arm Acceleration Profiles for a Complete Cycle

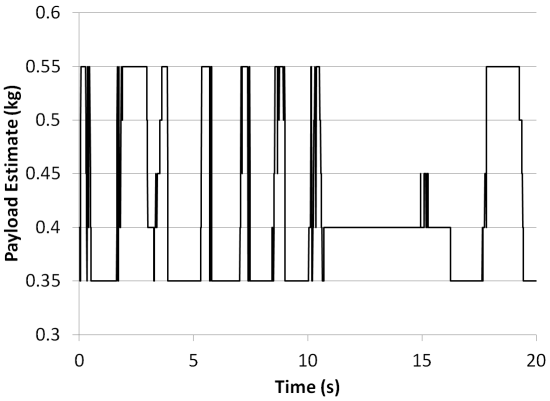
Examining the parameter estimates and weights associated with the MMSAE reveals interesting behavior. From Figure 102 it is apparent that, because the excitation of the flexible structure is minimized by the control system, the true parameter is rarely identified. In the adaptive control literature similar effects are referred to as persistence of excitation requirements. The estimator operates on the output estimation error, however if the structure is not vibrating, the accelerations are zero (or at least below a meaningful signal to noise threshold). This means that the state estimates produced by any of the estimators is equally valid. Furthermore, since the controller is designed to eliminate the residual vibration, once the oscillation is damped below certain threshold the control effort is too small to overcome the cart stiction. Excitation of the flexible modes is required in order to determine the true parameter value. In a particular part of the track (encountered at approximately 11 seconds in Figures 102(a), 102(c), and 102(e)), the stiction is significant and results in a small residual oscillation that the controller is unable to damp. In this residual period, the true parameter is more closely approximated.



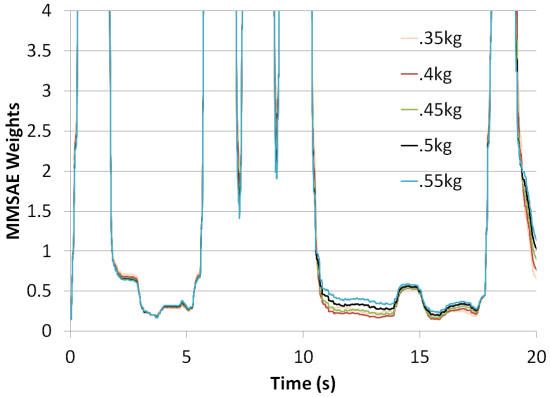
(a) Payload Estimate (0.36kg + no additional payload)



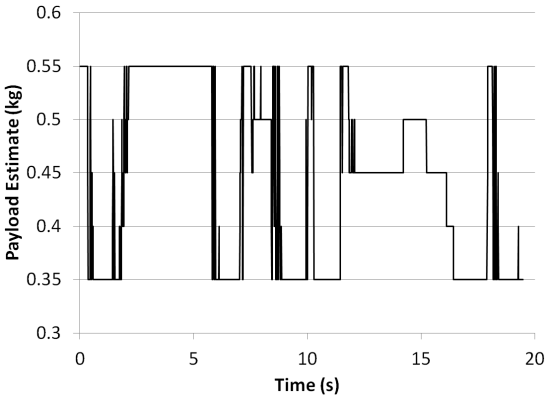
(b) Estimate Weights (0.36kg + no additional payload)



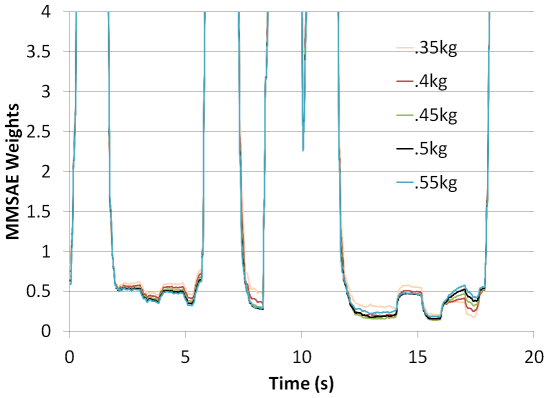
(c) Payload Estimate (0.36kg + 0.059kg payload)



(d) Estimate Weights (0.36kg + 0.059kg payload)



(e) Payload Estimate (0.36kg + 0.143kg payload)



(f) Estimate Weights (0.36kg + 0.143kg payload)

Figure 102: MMSAE Payload Estimates and Weights

6.9 Notes on Sensor Redundancy and Parallelization

The multiple model estimator structure is ideal for developing fault tolerant control systems for flexible manipulators with redundant sensors. Requiring only the generation of additional sub-estimators associated with the redundant sensors or sensor groups, if sensor failure were to occur, the state difference or output difference metrics would grow significantly. Thus, the offending sub-estimators would be eliminated from consideration in the selection of the resulting state estimate vector.

The many advances in computing technology have made low cost, high speed parallel computing much more feasible today then it was in the early 1990's when multiple model techniques were initially developed and largely discarded because of the associated high computational burden. Graphics processing units (GPU) and low cost microprocessors could be utilized to parallelize the multiple model estimation strategy, thereby, boosting performance and expanding the range and number of parametric modeling errors tolerated by the control system.

Chapter VII

SUMMARY, CONCLUSIONS, AND EXTENSIONS

7.1 Summary and Conclusions

A novel robust estimation strategy for observing the flexible states of a flexible robotic manipulator is established in this thesis. This multiple model adaptive estimator technique permits the state feedback control of these systems even under conditions of parametric modeling error and/or time varying system parameters. This summary serves as a concise description of the performed work and major results of the preceding thesis sections.

Motivation for the derived approach stems from the observation that lightweight, high speed, and large workspace robotic manipulators often suffer performance degradation because of inherent structural flexibility. This flexibility often results in persistent residual vibration, which must be damped before useful work can resume. Increased cycle times are the undesirable consequences of residual vibration. Traditional compensation techniques commonly used for the control of rigid manipulators can only approach a fraction of the open-loop system bandwidth without inducing significant excitation of the resonant dynamics. Therefore, the performance objectives are artificially limited (i.e., control gains are reduced and slow trajectories implemented) to avoid induced oscillation. To improve the performance of these systems, the structural flexibility cannot simply be ignored as it is when the links are significantly stiff and approximate rigid bodies. Instead, detailed models of the complete system must be used to anticipate the effects of structural flexibility and control actions applied which compensate for the undesired behavior. Nevertheless, any determined model of the system will contain parametric error. And, in the case of very lightly damped systems like flexible robotic manipulators, even small errors in system parameters can lead to instability of the control system. Furthermore, time varying changes in the manipulators pose or payload affect not only the rigid-body but also the flexible dynamics of the manipulator.

While very accurate modeling approaches have been created and verified over many years of research, modeling error is always present in practical implementations. Such errors restrict the performance and stability of these systems. Techniques for generating vibration limiting commands like input shaping and inverse dynamics are seeing more industrial use, in large part, because they improve the performance of flexible systems without impacting stability. However, feedback control of flexible manipulators in industry for the purpose of vibration damping and bandwidth improvement is, as of yet, unrealized.

A strategic modeling approach was employed in this thesis work to create an assumed modes model of the FLASHMan flexible gantry manipulator. The transfer matrix modeling method was utilized to determine the mode shapes and natural frequencies used in the assumed modes model approximation. This improved the low order approximation of the flexible system dynamics To improve the accuracy of state estimates, the flexibility of the belt drive was modeled and used to change the control domain from an estimated force input on the cart to a measured displacement of the belt. This change also allowed the full state vector to be estimated solely from acceleration measurements, whereas with force input a position measurement would also be necessary to recover rigid body motion. The equations of motion are represented in state space form through a modal transformation. This results in a three mode approximation of the joint and distributed flexibility: mode 1 corresponds primarily to the 1st clamped mode of the flexible beam, mode 2 to the belt/cart motion, and mode 3 to the second clamped beam mode. Experimental results were used to confirm the accuracy of the derived modeling approach.

Low cost MEMs accelerometers were evaluated for reconstructing the flexible system state vector. Augmenting the state space system to include acceleration measurements introduces direct feed-through behavior in the estimation routine, an effect which is often ignored in the literature. The detrimental effects of modeling error on classical state estimation with direct feed through behavior were investigated and documented through simulations and experiments of the FLASHMan system.

A systematic approach utilizing a genetic algorithm optimization was conceived in order to determine sensor placements for multi-sensor state estimation approaches. It was

determined that by using a sensor for each mode, placed at the location of highest singular value of the corresponding observability gramian (for that mode), observability is guaranteed. Multiple sensor placement provides more robust observability than a single sensor placement, where changing system parameters can lead to modal occlusion (i.e. where no part of a sensors measurement can be attributed to a specific mode of vibration).

Sliding mode observers were investigated to assess the purported robustness properties that several researchers had attributed to the estimation approach. It was determined through equilibrium analysis, simulation, and experiments that the primary benefit of the sliding mode approach over that offered by the Kalman filter or Luenberger observer, is an increased convergence rate and guaranteed stability for BIBO stable nonlinear systems. However, it offers no improved robustness to parametric error for this application, and any convergence rate improvements are tempered by the presence of significant increases in state estimate noise.

A novel strategy was created for estimating the full flexible state of flexible manipulators. It entailed the development of a multiple model representation of the expected system parameters. Independent sub-estimators are used to determine the state vector and a weight proportional to the relative accuracy of that estimated state vector for a given number of estimators: multiple sensors provide the necessary information on perturbations of the system model. These weights are then used to compare individual estimators from a static estimator bank to determine the best fit estimator. In doing so, the true plant parameter is estimated and the best model is determined. This approach was evaluated both in simulation and with experiments to determine the ability of the estimation strategy to adapt to discrete changes in model parameters. It was determined that the constructed multiple model adaptive estimation approach successfully ascertained the correct parameter when discrete changes in the system dynamics occurred (the manipulator picked up or dropped off a payload of known mass). This approach was also used for state feedback control of the flexible state to correct for disturbances and eliminate residual vibration. Results indicate that the new estimation method not only maintains control system stability for a larger range of parametric variation relative to a single estimator/controller designed for a

nominal payload, but also improves the repeatability of cycle performance over the range of experienced parametric error.

7.1.1 Summarized Contribution List

This thesis makes contributions to the areas of: modeling flexible systems, sensing and sensor placement for flexible manipulators, robust estimation strategies, multiple model estimation, and feedback control of flexible manipulators. The primary contribution of this work is the generation, fundamental analysis, and experimental verification of a robust strategy for state estimation for general flexible manipulators. Specific contributions include:

- The creation of a state space model for the FLASHMan flexible gantry robot and a generalized approach for flexible manipulators with both drive and link flexibility
    - TMM & AMM combined approach for accurate low order model approximations
    - Improved accuracy and guaranteed observability of estimators for flexible manipulators, using only acceleration feedback by leveraging flexible joint characteristics
- The generation of an approach for multiple sensor placements utilizing the singular values of observability gramians
    - Genetic algorithm formulation for sensor placements that guarantee optimal and robust observability
    - Improved robustness with respect to observability for multiple sensor placements
- The analytical, simulation, and experimental evaluation of sliding mode observation for flexible state estimation
    - Compared Luenberger, Kalman filter, and sliding mode observers for estimating the states of flexible motion systems
    - Demonstrated negligible robustness improvements vs. traditional estimators for parametric modeling error in the flexible manipulator model

- The formation and experimental verification of a novel multiple model adaptive estimator for robust state estimates of flexible manipulators
    - Derivation of sensor and output difference metrics for weighting state estimates
    - Nested sub-estimator approaches for multiple model estimation
    - Simulation and experimental evaluation of the derived approach
    - Design, and implementation of a stable multiple model control system

7.2 Extensions and Future Work

7.2.1 Sensitivity Analysis - Persistence of Excitation

The results of this work show that in order to correctly identify the system parameter, accelerations corresponding to the residual vibration must be non-zero and larger then the sensor noise amplitude. This means that if the link vibrations are suppressed (the desired end goal), the estimation algorithm will not be able to select the true parameter value. This can be considered to be a form of persistent excitation condition. It is conceivable that there is a minimal level of vibration that cannot be suppressed, because the appropriate controller cannot be selected. This, in turn, results in more vibration, allowing the estimation algorithm to select the correct model and thereby suppress the vibration to below the original threshold. The result is likely limit cycling, the extent of which depends on the aggressiveness of the controller and the precision of the chosen sensors. Note that this behavior has not been observed in the experiments reported in this work. Its absence is likely the result of significant stiction in the FLASHMan testbed, which prevents the cart from moving when small control signals are applied to the system.

7.2.2 Multiple Link Manipulators

The primary limitation of this work, in its current state, is the reliance on a linear model of the system. This limitation means the results, as derived, are only applicable for small motions about a fixed manipulator pose. Expansion to multiple links for large motions, while probably of moderate research interest, is of great practical importance. Most of the necessary components for extending the derived methods to multiple links exist in various

forms. For example, models of multiple link manipulators appear in [4, 11, 12, 15, 21, 99]. Additionally estimators suitable for implementation in the multiple model framework, like the extended Kalman filter have already been applied to multiple link flexible manipulators [60]. What remains is a way to ensure that sensors are placed such that they do not suffer modal occlusions over the entire workspace of the robotic manipulator. This could simply be accomplished by increasing the number of sensors and either switching between sensor groups as the manipulators pose changes, or by using all of the measurements in each estimator. Furthermore, the state and output difference weighting metrics were derived under the assumption of a linear model. Extensions to nonlinear models, while intuitively promising, must be studied.

7.2.3 Parallel Computing and Network Based Consensus Estimation

One of the main drawbacks of multiple model approaches, and the reason for the recent work in the literature suggesting growing?/shrinking of the estimator banks, is that running simultaneous estimators on a single processor is computationally expensive. The performance of the estimator and the system sample rate are therefore highly restricted, based on the processing capability of the real time system. Such constraints limited the number of estimators in the multiple model bank to five in the system examined in this thesis. Multiple core computing, graphics processing units (GPUs), and embedded systems pose potential solutions to the problem of increased computational burden. If each estimator is run independently on its own processor, there is no limit to the number of estimators (n) in or dimensions (p) of a multiple model estimator bank. The problem, therefore, reduces to ensuring synchronicity between the estimators and enabling communication to a central agent. The central agent then performs the decision making tasks, (i.e. selecting the best state from among the estimators). Rather than a central agent receiving np estimates and weights, a more elegant solution would be for each estimator to communicate with a limited set of other estimators. This would allow the complete set to come to a consensus of the best state estimate. Network based weighted consensus estimation has been studied for the purpose of sensor fusion [91, 106] and multiple model estimation is a natural extension.

7.2.4 A Complete Solution for Controlling Flexible Robotic Manipulators

Robust flexible state variable estimation, as introduced in this dissertation, forms one part of a complete solution with the potential to enable industrial use of light weight, high speed, and large workspace manipulators. However, any comprehensive solution must consider three distinct domains in order to be a generally applicable solution: trajectory design, control design, and hardware design.

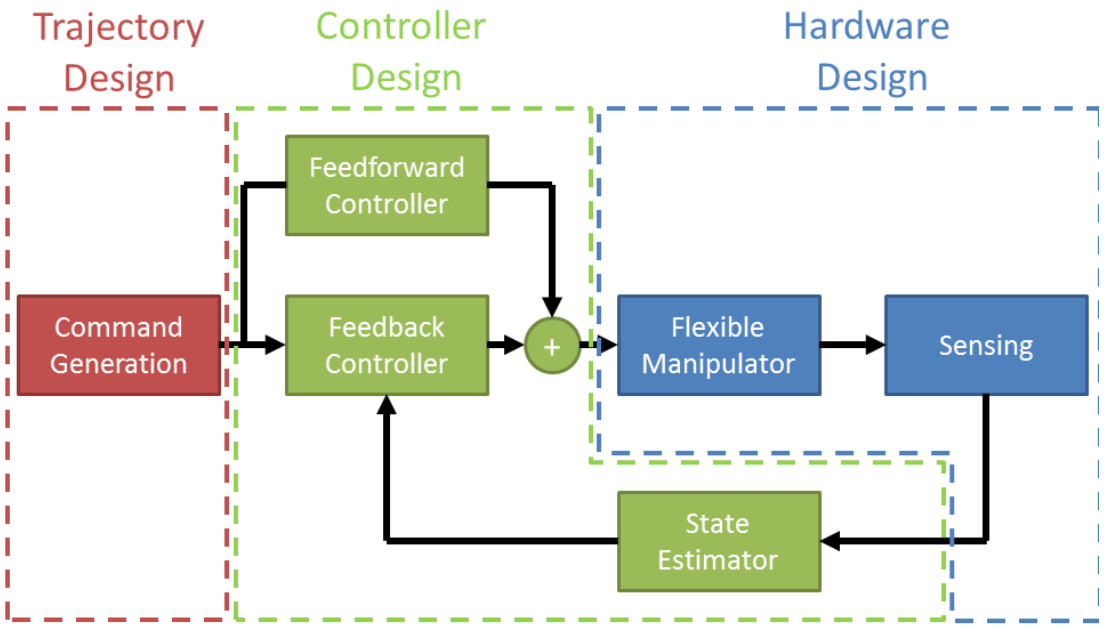


Figure 103: Complete Structure for the Control of Flexible Manipulators

Of course, when possible, robotic manipulators should be designed to be as stiff as possible while maintaining the desired levels of performance, energy consumption, and overall cost. Often however, the application dictates structures where flexibility is an unavoidable consequence of the desired workspace, performance requirements, or cost. Design solutions should be explored to raise the stiffness of the system, for example, by using composite materials or increasing the structural damping as in [2, 3]. If flexibility remains a barrier, then mitigation with active control systems becomes a viable option.

Trajectory design for flexible manipulators has been extensively studied. The three main techniques that have emerged as potential solutions for generating vibration limiting trajectories are command smoothing, where trajectories are filtered to reduce the aggressiveness

of the applied commands [89], input shaping , where the trajectories are convolved with impulses designed to eliminate residual oscillation [66,67,90], and inverse dynamics [55,80], where the complete dynamic model is utilized to generate commands which prescribe the end of arm trajectory at each time instant. While these methods have been very successful, they require accurate knowledge of the system to be effective and are entirely open loop approaches, meaning that disturbances cannot be rejected.

Feedback control can modify the dynamics of system, increasing the performance. However, errors must be observed before any corrective action is taken. Feedforeward control supplements feedback control by applying control effort in response to changing set points rather than feedback signals. This reduces the effort required from the feedback control system. Reducing the burden on the reactive feedback controller, allows the use of larger feedback gains as large errors are compensated for by the feedforeward controller, leaving the reactive controller to handle small errors and disturbances.

Often overlooked, robust state estimation forms a critical piece of a complete solution for the control of flexible manipulators. The novel robust multiple model adaptive estimator created in this work serves as a potential systematic solution to this problem, enabling the state feedback control of these systems and eventual industrial acceptance of flexible manipulator technology.

Appendix A

STABILITY PROPERTIES OF GAIN SCHEDULED STATE SPACE CONTROL

Gain scheduling in its many forms is a nonlinear control method where controller gains are precomputed. The job of the controller is to then, based on the current operational mode, switch between these gain sets. For the sake of analysis, gain scheduled controlled systems can be represented by a generic nonlinear state equation.

$$z_{k+1} = f(z_k, u_k) \tag{301}$$

Which, in turn, can be represented as a summation of individual linear plants

$$z_{k+1} = \sum_{i=1}^{N} \left(z_{di} + A_i(z_k - z_{di}) + B_i(u_k - u_{di}) \right) w_i(\nu_k) \tag{302}$$

where $\sum_{i=1}^{N} w_i(\nu_k) = 1$ is a selection function which determines the current plant based on a scheduling variable ν_k and the individual controllers are state feedback controllers of the form

$$u_k = u_{di} + \sum_{i=1}^{N} K_i(z_k - z_{di}) w_i(\nu_k) \tag{303}$$

And therefore the closed loop system is

$$z_{k+1} = \tilde{A}_k(\nu_k) z_k \tag{304}$$

which is essentially a traditional dynamic system formulation with a time varying plant matrix $\tilde{A}$

$$\tilde{A}_k(\nu_k) = A_k(\nu_k) + B_k(\nu_k) K_k(\nu_k) \tag{305}$$

If $\tilde{A}_k$ is designed to be stable (i.e. all eigenvalues in the unit circle and more specifically inside a disk of radius $1-2\epsilon$, $\epsilon > 0$)

$$max\left|\lambda_j(\tilde{A}_k)\right| \leq 1-2\epsilon < 1 \tag{306}$$

then the global control system is exponentially stable provided the sequence of matrices $\tilde{A}_k(\nu_k)$ is slowly varying, i.e.

$$sup\left\|\tilde{A}_{k+1}-\tilde{A}_k\right\| \leq \delta \tag{307}$$

The proof of the assertion in (307) is reproduced here from the work of Desoer [30]. The first step is to establish a bound on A_k^N as described below

$$\tilde{A}_k^N = \frac{1}{2\pi j}\oint_c s^N(sI-\tilde{A}_k)^{-1}ds \tag{308}$$

then the norm of A_k^N is

$$\left\|\tilde{A}_k^N\right\| \leq (1-\epsilon)^{N+1}\frac{(1-\epsilon)+(a_M)^{n-1}}{\epsilon^n} \tag{309}$$

where

$$a_M = sup\left\|\tilde{A}_k\right\| \tag{310}$$

Let $\rho = 1-\epsilon$, then

$$\left\|\tilde{A}_k^N\right\| \leq m\rho^N \tag{311}$$

where m is independent of k. For the kth sampling instant pick a Lyapunov function

$$V_k = z_k^T P_k z_k \tag{312}$$

Therefore,

$$V_{k+1} - V_k = z_k^T (\tilde{A}_k^T P_{k+1} \tilde{A}_k - P_k) z_k \tag{313}$$

Choosing P_{k+1} such that

$$\tilde{A}_k^T P_{k+1} \tilde{A}_k - P_{k+1} = -I \tag{314}$$

which has a solution:

$$P_{k+1} = I + \sum_{c=1}^{\infty} (\tilde{A}_k^T)^c (\tilde{A}_k)^c \tag{315}$$

and therefore, from (311),

$$1 \leq \|P_{k+1}\| \leq \frac{m^2}{1-\rho^2} \qquad \forall k \tag{316}$$

and from (312)

$$\|z_k\|^2 \leq V_k \leq \frac{m^2}{1-\rho^2} \|z_k\|^2 \qquad \forall k \tag{317}$$

Taking the difference between successive time steps in (314)

$$\underbrace{\tilde{A}_k^T (P_{k+1} - P_k) \tilde{A}_k - (P_{k+1} - P_k)}_{-M_k} = -\left\{ (\tilde{A}_k^T - \tilde{A}_{k-1}^T) P_k \tilde{A}_k + \tilde{A}_{k-1}^T P_k (\tilde{A}_k - \tilde{A}_{k-1}) \right\} \tag{318}$$

Given that

$$\|M_k\| \leq 2 \left\| \tilde{A}_k - \tilde{A}_{k-1} \right\| \frac{m^2}{1-\rho^2} a_M \tag{319}$$

from (310) and (311). Therefore solving (318) yields

$$P_{k+1} - P_k = M_k + \sum_{c=1}^{\infty} (\tilde{A}_k^T)^c M_k (\tilde{A}_k)^c \tag{320}$$

so

$$\|P_{k+1} - P_k\| = \left\|\tilde{A}_k - \tilde{A}_{k-1}\right\| \frac{2m^2 a_M}{1-\rho^2} \frac{m^2}{1-\rho^2} \tag{321}$$

Then if,

$$sup \left\|\tilde{A}_k - \tilde{A}_{k-1}\right\| \leq \frac{(1-\rho^2)^2}{2m^4 a_M}(1-\eta) = \delta \tag{322}$$

where $\eta > 0$. It follows that

$$\|P_{k+1} - P_k\| \leq 1 - \eta < 1 \qquad \forall k \tag{323}$$

and therefore,

$$V_{k+1} - V_k \leq -\eta \|z_k\|^2 \tag{324}$$

meaning that stability is assured if

$$sup \left\|\tilde{A}_k - \tilde{A}_{k-1}\right\| \leq \delta \tag{325}$$

i.e. the system's controlled plant changes slowly (i.e. the parametrization of the models used in the multiple model estimator is not excessively sparse, and discrete switches occur incrementally)

Appendix B

CODE - FLASHMAN MODEL GENERATION

```
clc
close all
clear all
format short

Modes = 2;                  % Number of flexible modes to consider
F_Max = 1000;               % Max Control Bandwidth Hz


% Beam Properties
L = 42/100;                                          % Length m
a = 3.81/100;                                        % Width m
t = .317/100;                                        % Thickness m
E = 7*10^10;                                         % Youngs Modulus Pa
rho = 2700;                                          % Density kg/m^3
mbase = 10;                                         % Cart Mass kg
Damper = 100;                                        % Cart Damper Ns/m
LF = 0.0025;                                         % Beam Loss Factor
mtip = .281;                                        % Mass of Payload kg
Kbelt = 7*2.05*10^11*(pi*(3.81*10^(-4))^2/4)/.75; % Belt Stiffness N/m

% Derived Quantities
I = 1/12*a*t^3;      % Area Moment
A = a*t;             % Crossectional Area
m_beam = rho*L*a*t;  % Beam Mass
mu = m_beam/L;       % Distributed Mass

% Frequency Sweep to find Natural Frequencies and Mode Shapes
% (Using Transfer Matrix Method)
omega= 0.001;          % Starting Frequency (rad/s)
tol = 0.01;             % Sweep interval (rad/s)

a = L^2/(E*I);
num_roots = 0;
i = 1;
while num_roots < Modes && omega/(2*pi) < F_Max
    %Elastic Beam Matrix
    beta = (omega^2*(mu/(E*I))*L^4)^(1/4);
    c0(i) = (cos(beta)+cosh(beta))/2;
    c1(i) = (sin(beta)+sinh(beta))/(2*beta);
    c2(i) = (cosh(beta)-cos(beta))/(2*beta.^2);
    c3(i) = (sinh(beta)-sin(beta))/(2*beta.^3);

    U_beam(:,:,i) = [c0(i) , L*c1(i) , a*c2(i) , a*L*c3(i) ;...
```

```
                    beta^4*c3(i)/L , c0(i) , a*c1(i)/L , a*c2(i)  ;...
                    beta^4*c2(i)/a , beta^4.*L.*c3(i)/a , c0(i) , L*c1(i)  ;...
                  beta^4*c1(i)/(L*a) , beta^4*c2(i)/a , beta^4*c3(i)/L , c0(i)   ];

    % Rigid Body Matrix for Tip Mass
    R_tip(:,:,i) = [       1          , 0 , 0 , 0;...
                           0          , 1 , 0 , 0;...
                           0          , 0 , 1 , 0;...
                     mtip*omega^2     , 0 , 0 , 1];

    % Complete Transfter Matrix
    Utot(:,:,i) = U_beam(:,:,i)*R_tip(:,:,i);
    % Characteristic Equation
    CHAR(i) = det(Utot(1:2,1:2,i));
    O(i) = omega;
    % Search for the Roots of the Characteristic Equation!
    if length(CHAR)>1
        if sign(CHAR(i))~= sign(CHAR(i-1)) % Detect Sign Change
            num_roots = num_roots+1;
            roots(num_roots) = (O(i)+O(i-1))/2; % Add to List of Roots
            Beta(num_roots) = (roots(num_roots)^2*(mu/(E*I))*L^4)^(1/4);
% Beta @ Natural Frequencies
            C2(num_roots) = -(sin(beta)+sinh(beta))/(cos(beta)+cosh(beta));
% Constant for Mode Functions
            indices(num_roots) = i;
        end
    end
    omega = omega+tol;
    i = i+1;
end
% Display Natural Frequencies)
NaturalFrequenciesHz = roots/(2*pi)



%Determine Mode Shapes of the first link at the desired natural frequencies
for i=1:Modes

k=indices(i);
%Step 1 - Find the initial state vector at the beginning of the link by
%applying the boundary conditions
R = Utot(:,:,k);
Z_L = [-1;R(1,1)/R(1,2);0;0]; %Based on Boundary Conditions
%Z_0 = R*Z_L %At the very beginning
Z_0 = R*Z_L % [0;0;-R(3,1)+R(3,2)*(R(1,1)/R(1,2));-R(4,1)+R(4,2)*(R(1,1)/R(1,2))];

Z_payload = R_tip(:,:,k)*Z_L;
% Z_link2 = R_beam(:,:,k)*Z_payload;
% Z_theta2 = A_theta_2(:,:,k)*Z_link2;
Z_joint2 = R_tip(:,:,k)*Z_L; %R_joint(:,:,k)*Z_theta2;

chunk = .0001; %m
num_chunk=L/chunk;
```

```
    %Elastic Beam Matrix
    a = chunk^2/(E*I);
    beta = (roots(i)^2*(mu/(E*I))*chunk^4)^(1/4);
    c_0 = (cos(beta)+cosh(beta))/2;
    c_1 = (sin(beta)+sinh(beta))/(2*beta);
    c_2 = (cosh(beta)-cos(beta))/(2*beta.^2);
    c_3 = (sinh(beta)-sin(beta))/(2*beta.^3);

    % Flexible Matrix for Beams
    U_chunk_beam = [c_0 , chunk*c_1 , a*c_2  , a*chunk*c_3 ;...
                    beta^4*c_3/chunk , c_0 , a*c_1/chunk , a*c_2    ;...
                      beta^4*c_2/a , beta^4.*chunk.*c_3/a , c_0 , chunk*c_1  ;...
                   beta^4*c_1/(chunk*a) , beta^4*c_2/a , beta^4*c_3/chunk , c_0 ];

Z_link1(:,1)=Z_joint2;
for j=1:round(num_chunk)
Z_link1(:,j+1) = U_chunk_beam*Z_link1(:,j);
end
Z_link1=flipdim(Z_link1,2);
Modeshape(i,:) = Z_link1(1,1:num_chunk+1); %/max(abs(Z_link1(1,1:num_chunk+1)));
end
x = 0:chunk:L; % Displacement Along Link


% % Determine Asssumed Modes Model of the Complete OL Plant
% tol = 0.001; % Integration Tolerance
% x = 0:tol:L; % Displacement Along Link


% UNCOMMENT TO PLOT MODE SHAPES
% figure(1)
%
% plot(x,Modeshape)

% color = ['r','g','b','k'];
% for i = 1:length(roots)
%     phi = sin(Beta(i)*x./L)-sinh(Beta(i)*x./L)+C2(i)*(cos(Beta(i)*x./L)-cosh(Beta(i)*x./L));
%     plot(x,phi,color(i))
% end

% Given the NEW basis functions, compute the ASSUMED MODES MODEL
% (Since the basis functions are the solution to the PDE we dont need to
% extend the series approximation!)

% Rigid Mass Terms
m_total = mbase+m_beam+mtip;

%Flexible Coupling Terms (Couples Rigid and Flexible Subsystems)
for i = 1:Modes
    W_0 = rho*A*Modeshape(i,:).*x;
    W(i)= (trapz(W_0)*chunk+mtip*Modeshape(i,end))
    d_dx_Modeshape(i,:)=[0,diff(Modeshape(i,:))/chunk];
    d2_dx2_Modeshape(i,:)=[0,diff(d_dx_Modeshape(i,:))/chunk];
end
```

```
for i = 1:Modes % Flexible Subsystem
    for j = 1:Modes
        M_stuff = rho*A*(Modeshape(i,:).*Modeshape(j,:));
        M(i,j) = trapz(M_stuff)*chunk+mtip*(Modeshape(i,end).*Modeshape(j,end));

        K_stuff = E*I*d2_dx2_Modeshape(i,:).*d2_dx2_Modeshape(j,:);
        K(i,j) = trapz(K_stuff)*chunk;
    end
end

M = [m_total , W ;...
        W'   , M ];
K = [    Kbelt      , zeros(1,Modes);...
     zeros(Modes,1) ,       K       ];

C = LF*K; %Compute damping coefficient matrix
C(1,1) = Damper;

Q = [    Kbelt        ;
    zeros(Modes,1)  ];

%%%%%%%%%%%%%%%%%%%%%%%%%%%% Modal Analysis %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Solve Eigenvalue Problem for natural frequencies and eigenvectors
[eig_Vec,eig_Val]=eig(K,M);
% sort in ascending order
[w,index]=sort(diag(eig_Val));
% do the same for eigenvectors
phi=eig_Vec(:,index);
RealNaturalFrequenciesHz=sqrt(w)/(2*pi)

%Normalize Eigenvectors according to "modal masses" of individual modes
for i = 1:Modes+1
    phi(:,i) = phi(:,i)./(sqrt(phi(:,i)'*M*phi(:,i)));
end


% Basis Functions @ x = L for Output Matrices
for i = 1:Modes
    Basis(i)= Modeshape(i,end);
end

N_M_r = phi'*M*phi;
N_C_r = phi'*C*phi;
N_K_r = phi'*K*phi;
N_Q_r = phi'*Q;

A = [zeros(Modes+1), eye(Modes+1);...
        -N_K_r     ,    -N_C_r];

B = [zeros(Modes+1,1); N_Q_r];
```

```
 C = [[1,zeros(1,Modes)]*phi,zeros(1,Modes+1)];

 D = [0];

 Plant = ss(A,B,C,D);


 Ctip = [[1,Basis]*phi,zeros(1,Modes+1)];

 TipPlant = ss(A,B,Ctip,D);


 C_accel = [[1,Basis]*phi*-N_K_r, [1,Basis]*-N_C_r];
 D_accel = [1,Basis]*phi*N_Q_r;

AccPlant = ss(A,B,C_accel,D_accel);
```

Appendix C

CODE - GENETIC ALGORITHM FOR ROBUST SENSOR PLACEMENTS

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%              Genetic Algorithm For Robust Sensor Placement                    %
%                                   Settings                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc
clear all
close all

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                               Genetic Algorithm                               %
%                                   Settings                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

mode_num = 1; %Selected mode for sensor placement

% Define Population Size
Pop_num = 20;
% Define Number of Elites to Keep
Elites = 2;
% Define Crossover Probability
Crossover = 25; % Chance of occurance (%)
% Define Mutation Probability
Mutation = 80; % Chance of occurance (%)
Mut_bits = 25; % Percentage of genes to mutate (%)
% Define the number of generations
num_generations = 100;

% Define Chromosomes Precision
Sig_Dig = 6;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                 Plant Model                                   %
%                                   Settings                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

L_p = 42/100; % Length m
M_p = 10;% Cart Mass kg
m_p = .281; % Mass of Payload kg
D_p = 100; % Cart Damper Ns/m
F_p = 0.0005; % Beam Loss Factor
```

```
stdev_m_p = 0.15; % Small Perturbation to tip mass

num_plants = 10; % # of gaussian distributed plants to consider in fitness metric
m_p_vec = abs(m_p+stdev_m_p*randn(num_plants,1));
% Calculate Plant Models
for i = 1:num_plants
    [N_M_r{i}, N_C_r{i}, N_K_r{i}, N_Q_r{i}, phi{i},Beta{i}, C2{i},...
RealNaturalFrequenciesHz{i}] = Single_Link_Model_Func(L_p,M_p,D_p,F_p,m_p_vec(i));
end

% Define Search Bounds
max_val = L_p;
min_val = 0;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              Inital Populaton                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Create Initial Random Population Chromosomes
Pop_dec =1/(10^Sig_Dig)*randi([ceil(min_val*10^Sig_Dig),...
    floor(max_val*10^Sig_Dig)],[Pop_num,1]);
Before_dec = floor(Pop_dec./1);
After_dec = mod(Pop_dec,1)*10^Sig_Dig;
%convert to binary strings
% Step 1: Convert number before Decimal
if max_val>1
    LS_p_of_2 = nextpow2(max_val);
else
    LS_p_of_2 = 1;
end
LS_quant = quantizer([LS_p_of_2+1,0]);
LS = num2bin(LS_quant,Before_dec);
RS_p_of_2 = nextpow2(10^Sig_Dig);
RS_quant = quantizer([RS_p_of_2+1,0]);
RS = num2bin(RS_quant,After_dec);
Pop = strcat(LS,RS);
chromosome_length = length(Pop(1,:));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                         Iterate Through Generations                           %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for j = 1:num_generations
    % Convert Binary Population to Decimal
    int_val = bin2num(LS_quant,Pop(:,1:LS_p_of_2+1));
    dec_val = bin2num(RS_quant,Pop(:,LS_p_of_2+2:end))/10^Sig_Dig;
    value = int_val+dec_val;

    % Evaluate Fitness of Each Chromosome
    for i = 1:Pop_num
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              Fitness Function                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Find location of lowest average condition number for the plants
```

```
        for l = 1:num_plants
        [Accplant] = AssemblePlant_Single_Link(N_M_r{l}, N_C_r{l}, N_K_r{l},...
N_Q_r{l}, phi{l}, Beta{l}, C2{l}, value(i), L_p);
         [U,S,V]= svd(gram(Accplant,'o'));
         [~,loc] = min(1-abs(U(mode_num,:))); %Select Mode to Consider
         temp_fitness(l) = S(loc,loc);
        end
        fitness_values(i,1) = mean(temp_fitness);
    end
    % Sort Chromosomes by Fitness
    [sorted_fitness,indices] = sort(fitness_values,'descend');
    sorted_chromosomes = Pop(indices,:);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              Population Plot                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    figure(1)
    hold off
    plot(value(indices),sorted_fitness,'g*',value(indices(1)),sorted_fitness(1),'r*')
%,min_val:.01:max_val,...


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                 Reproduction                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Assign Selection Probabilitys Based on Rank (Higher Rank -> better chance
    % of reproduction)
    Selection_Prob = [length(sorted_chromosomes(:,1)):-1:1]...
        ./length(sorted_chromosomes(:,1)); % Linear Weighting

    % Keep desired number of elites
    New_Pop(1:Elites,:) = sorted_chromosomes(1:Elites,:);
    index = Elites+1;

    while index <= Pop_num
        % Select 2 individuals for reproduction
        Parent_1 = randsample([1:Pop_num],1,true,Selection_Prob);
        Parent_2 = randsample([1:Pop_num],1,true,Selection_Prob);

        % Identify the Rank the Parents
        Dominant_Parent = sorted_chromosomes(min(Parent_1,Parent_2),:);
        Recessive_Parent = sorted_chromosomes(max(Parent_1,Parent_2),:);

        % Do not allow asexual reproduction
        if Parent_1 ~= Parent_2
            % Determine Genetics of offspring
            % Check for Crossover Event
            if randsample([0,1],1,true,[1-Crossover/100,Crossover/100])
% T = Crossover Event
                % Select Crossover Point (get more genes from dominant parent)
                CO_point = randsample([round(chromosome_length/2):chromosome_length],1);
                % Perform Crossover
                New_Pop(index,:) = [Dominant_Parent(1:CO_point),...
Recessive_Parent(CO_point+1:end)];
            elseif randsample([0,1],1,true,[1-Mutation/100,Mutation/100])
```

```
% T = Mutation Event
                inrange = 0; % Only allow mutations in valid range
                while inrange == 0
                    % Select Mutation indices (stay away from really early indices)
                    M_indices = randsample([1:chromosome_length],...
ceil(chromosome_length*rand()*Mut_bits/100)...
                        ,true,[1:chromosome_length]./chromosome_length);
% Prefer random modifications to end bits
                    % Perform Mutation
                    New_Pop(index,:) = Dominant_Parent;
                    New_Pop(index,M_indices) = num2str(Dominant_Parent(M_indices)=='0','%u');
                    % Check that the new mutation is in the search range
                    int_check = bin2num(LS_quant,New_Pop(index,1:LS_p_of_2+1));
                    dec_check = bin2num(RS_quant,New_Pop(index,LS_p_of_2+2:end))/10^Sig_Dig;
                    check_val = int_check+dec_check;
                    if check_val < max_val && check_val > min_val
                        inrange = 1;
                    end
                end
            else % No Crossover or Mutation (select most dominant parent and copy)
                New_Pop(index,:) = Dominant_Parent;
            end
            index = index+1;
        end
    end
    Pop = New_Pop;
    %pause(0.1)
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                             Display Solution                                  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Solution_Chromosome = Pop(1,:)
int_sol = bin2num(LS_quant,Pop(1,1:LS_p_of_2+1));
dec_sol = bin2num(RS_quant,Pop(1,LS_p_of_2+2:end))/10^Sig_Dig;
Solution_val = int_sol+dec_sol
Fitness = sorted_fitness(1)

figure(1)
hold on
plot(Solution_val,Fitness,'ro')


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                           FLASHMAN MODEL FUNCTION                             %
%                                                                               %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function [N_M_r, N_C_r, N_K_r, N_Q_r, phi,Beta, C2, RealNaturalFrequenciesHz] ...
= Single_Link_Model_Func(L,mbase,Damper,LF,mtip)

Modes = 2;                  % Number of flexible modes to consider
```

```
F_Max = 1000;               % Max Control Bandwidth Hz


% Beam Properties
% L = 40/100;                                          % Length m
a = 3.81/100;                                         % Width m
t = .317/100;                                         % Thickness m
E = 7*10^10;                                          % Youngs Modulus Pa
rho = 2700;                                           % Density kg/m^3
% mbase = 10;                                          % Cart Mass kg
% Damper = 100;                                        % Cart Damper Ns/m
% LF = 0.001;                                          % Beam Loss Factor
% mtip = .22;                                         % Mass of Payload kg
Kbelt = 7*2.05*10^11*(pi*(3.81*10^(-4))^2/4)/.75; % Belt Stiffness N/m

% Derived Quantities
I = 1/12*a*t^3;      % Area Moment
A = a*t;             % Crossectional Area
m_beam = rho*L*a*t;  % Beam Mass
mu = m_beam/L;       % Distributed Mass

% Frequency Sweep to find Natural Frequencies and Mode Shapes
% (Using Transfer Matrix Method)
omega= 0.001;          % Starting Frequency (rad/s)
tol = 0.01;             % Sweep interval (rad/s)

a = L^2/(E*I);
num_roots = 0;
i = 1;
while num_roots < Modes && omega/(2*pi) < F_Max
    %Elastic Beam Matrix
    beta = (omega^2*(mu/(E*I))*L^4)^(1/4);
    c0(i) = (cos(beta)+cosh(beta))/2;
    c1(i) = (sin(beta)+sinh(beta))/(2*beta);
    c2(i) = (cosh(beta)-cos(beta))/(2*beta.^2);
    c3(i) = (sinh(beta)-sin(beta))/(2*beta.^3);

    U_beam(:,:,i) = [ c0(i) , L*c1(i) , a*c2(i) , a*L*c3(i) ;...
                      beta^4*c3(i)/L ,  c0(i) , a*c1(i)/L , a*c2(i)  ;...
                      beta^4*c2(i)/a , beta^4.*L.*c3(i)/a , c0(i) , L*c1(i)  ;...
                    beta^4*c1(i)/(L*a) , beta^4*c2(i)/a , beta^4*c3(i)/L , c0(i)   ];

    % Rigid Body Matrix for Tip Mass
    R_tip(:,:,i) = [       1          , 0 , 0 , 0;...
                           0          , 1 , 0 , 0;...
                           0          , 0 , 1 , 0;...
                       mtip*omega^2   , 0 , 0 , 1];

    % Complete Transfter Matrix
    Utot(:,:,i) = U_beam(:,:,i)*R_tip(:,:,i);
    % Characteristic Equation
    CHAR(i) = det(Utot(1:2,1:2,i));
    O(i) = omega;
    % Search for the Roots of the Characteristic Equation!
```

```
    if length(CHAR)>1
        if sign(CHAR(i))~= sign(CHAR(i-1)) % Detect Sign Change
            num_roots = num_roots+1;
            roots(num_roots) = (O(i)+O(i-1))/2; % Add to List of Roots
            Beta(num_roots) = (roots(num_roots)^2*(mu/(E*I))*L^4)^(1/4);
% Beta @ Natural Frequencies
            C2(num_roots) = -(sin(beta)+sinh(beta))/(cos(beta)+cosh(beta));
% Constant for Mode Functions
        end
    end
    omega = omega+tol;
    i = i+1;
end
% Display Natural Frequencies)
NaturalFrequenciesHz = roots/(2*pi);

% Determine Asssumed Modes Model of the Complete OL Plant
tol = 0.001; % Integration Tolerance
x = 0:tol:L; % Displacement Along Link


% Given the NEW basis functions, compute the ASSUMED MODES MODEL
% (Since the basis functions are the solution to the PDE we dont need to
% extend the series approximation!)

% Rigid Mass Terms
m_total = mbase+m_beam+mtip;

%Flexible Coupling Terms (Couples Rigid and Flexible Subsystems)
for i = 1:Modes
    W_0 = rho*A*(sin(Beta(i)*x./L)-sinh(Beta(i)*x./L)+C2(i)*...
(cos(Beta(i)*x./L)-cosh(Beta(i)*x./L)));
    W(i)= (trapz(W_0)*tol+mtip*(sin(Beta(i)*L/L)-...
sinh(Beta(i)*L/L)+C2(i)*(cos(Beta(i)*L/L)-cosh(Beta(i)*L/L))));
end



  for i = 1:Modes % Flexible Subsystem
      for j = 1:Modes
          M_stuff = rho*A*(sin(Beta(i)*x/L)-sinh(Beta(i)*x/L)...
+C2(i)*(cos(Beta(i)*x/L)-cosh(Beta(i)*x/L)))...
              .*(sin(Beta(j)*x/L)-sinh(Beta(j)*x/L)+C2(j)*...
(cos(Beta(j)*x/L)-cosh(Beta(j)*x/L)));
          M(i,j) = trapz(M_stuff)*tol+mtip*(sin(Beta(i)*L/L)...
-sinh(Beta(i)*L/L)+C2(i)*(cos(Beta(i)*L/L)-cosh(Beta(i)*L/L)))...
              .*(sin(Beta(j)*L/L)-sinh(Beta(j)*L/L)+...
C2(j)*(cos(Beta(j)*L/L)-cosh(Beta(j)*L/L)));

              K_stuff = E*I*(Beta(i)/L)^2*(-sin(Beta(i)*x/L)-...
sinh(Beta(i)*x/L)-C2(i)*(cos(Beta(i)*x/L)+cosh(Beta(i)*x/L)))...
              .*((Beta(j)/L)^2*(-sin(Beta(j)*x/L)-sinh(Beta(j)*x/L)-...
C2(j)*(cos(Beta(j)*x/L)+cosh(Beta(j)*x/L))));
          K(i,j) = trapz(K_stuff)*tol;
```

```
      end
  end

  M = [m_total , W ;...
          W'   , M ];
  K = [    Kbelt      , zeros(1,Modes);...
       zeros(Modes,1) ,       K       ];

  C = LF*K; %Compute damping coefficient matrix
  C(1,1) = Damper;

  Q = [    Kbelt        ;
      zeros(Modes,1)  ];

  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Modal Analysis %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  % Solve Eigenvalue Problem for natural frequencies and eigenvectors
  [eig_Vec,eig_Val]=eig(K,M);
  % sort in ascending order
  [w,index]=sort(diag(eig_Val));
  % do the same for eigenvectors
  phi=eig_Vec(:,index);
  RealNaturalFrequenciesHz=sqrt(w)/(2*pi);

  %Normalize Eigenvectors according to "modal masses" of individual modes
  for i = 1:Modes+1
      phi(:,i) = phi(:,i)./(sqrt(phi(:,i)'*M*phi(:,i)));
  end


  % Basis Functions @ x = L for Output Matrices
  for i = 1:Modes
      Basis(i)= (sin(Beta(i)*L/L)-sinh(Beta(i)*L/L)+C2(i)*...
(cos(Beta(i)*L/L)-cosh(Beta(i)*L/L)));
  end

 N_M_r = phi'*M*phi;
 N_C_r = phi'*C*phi;
 N_K_r = phi'*K*phi;
 N_Q_r = phi'*Q;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              Assemble Plant                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


function AccPlant = AssemblePlant_Single_Link...
    (N_M_r, N_C_r, N_K_r, N_Q_r, phi, Beta, C2,x, L)

Modes = 2;

  for i = 1:Modes
      Basis(i)= (sin(Beta(i)*x/L)-sinh(Beta(i)*x/L)+C2(i)*...
          (cos(Beta(i)*x/L)-cosh(Beta(i)*x/L)));
```

```
  end


 A = [zeros(Modes+1), eye(Modes+1);...
         -N_K_r     ,    -N_C_r];
 B = [zeros(Modes+1,1); N_Q_r];
 C_accel = [[1,Basis]*phi*-N_K_r, [1,Basis]*-N_C_r];
 D_accel = [1,Basis]*phi*N_Q_r;

AccPlant = ss(A,B,C_accel,D_accel);
```

Appendix D

SLIDING MODE OBSERVER LABVIEW CODE

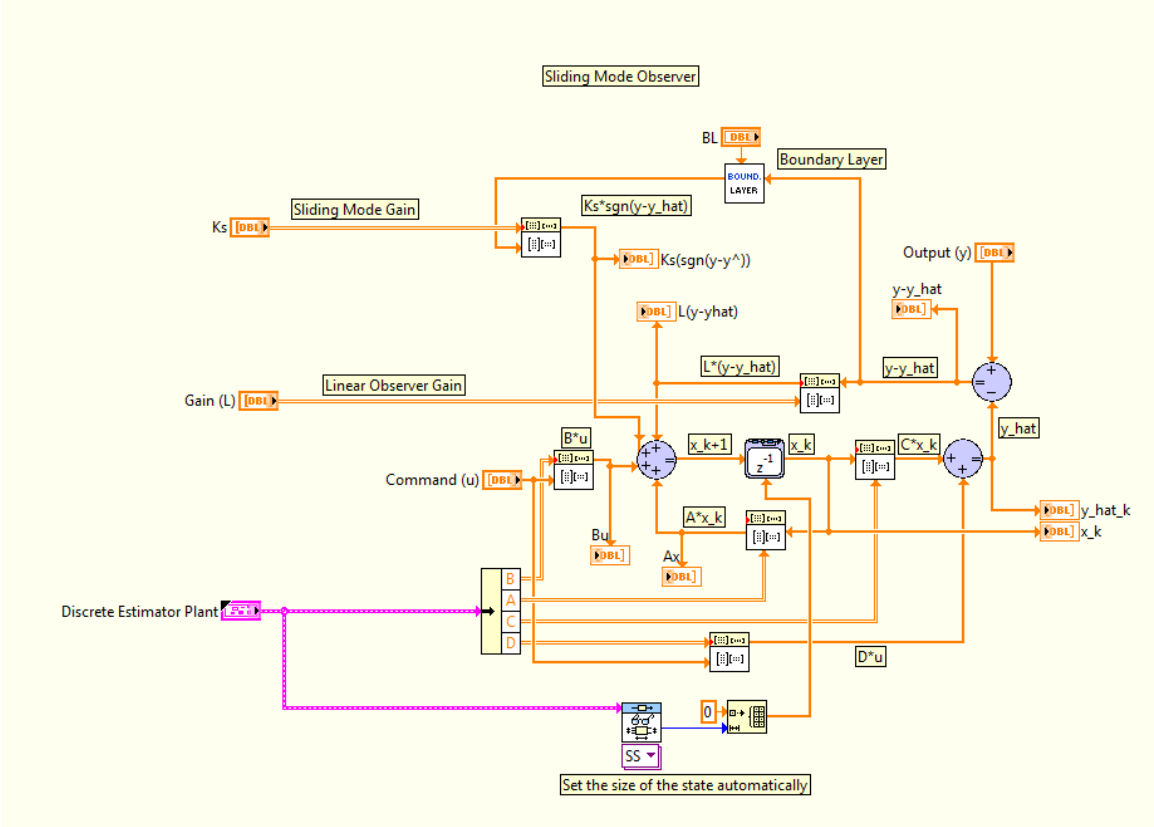


Figure 104: Sliding Mode Observer Block Diagram

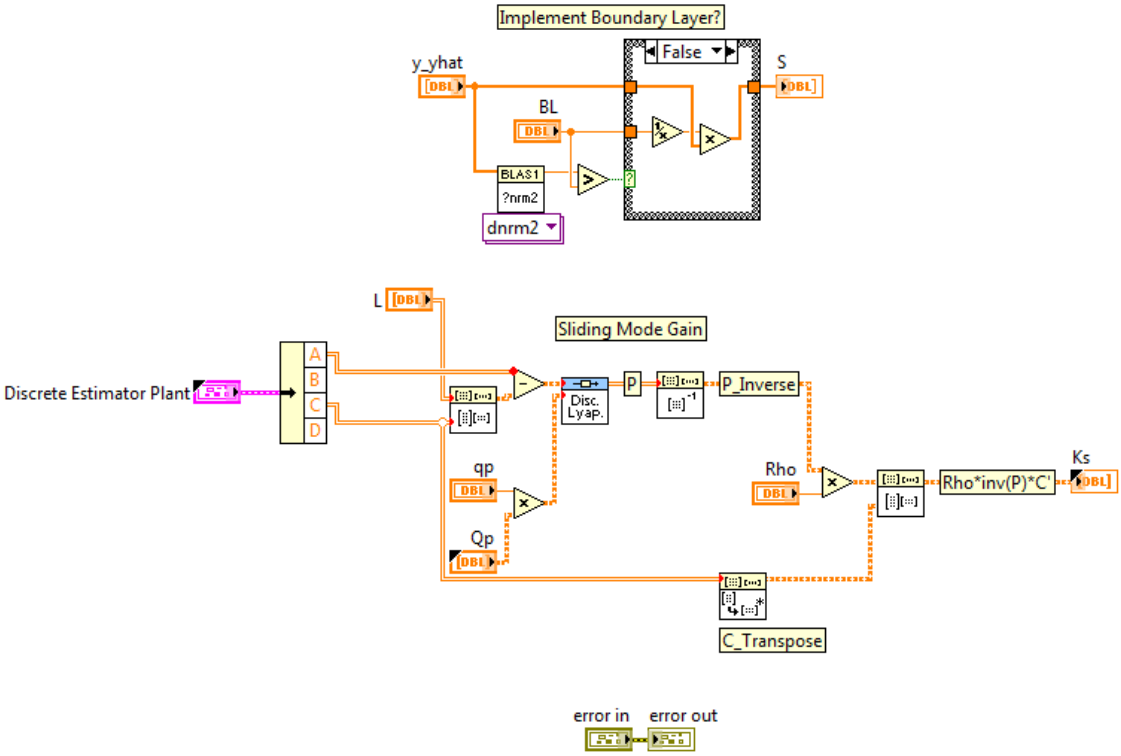


Figure 105: Sliding Mode Observer Gain Determiniation

Appendix E

MULTIPLE MODEL ADAPTIVE ESTIMATOR LABVIEW CODE

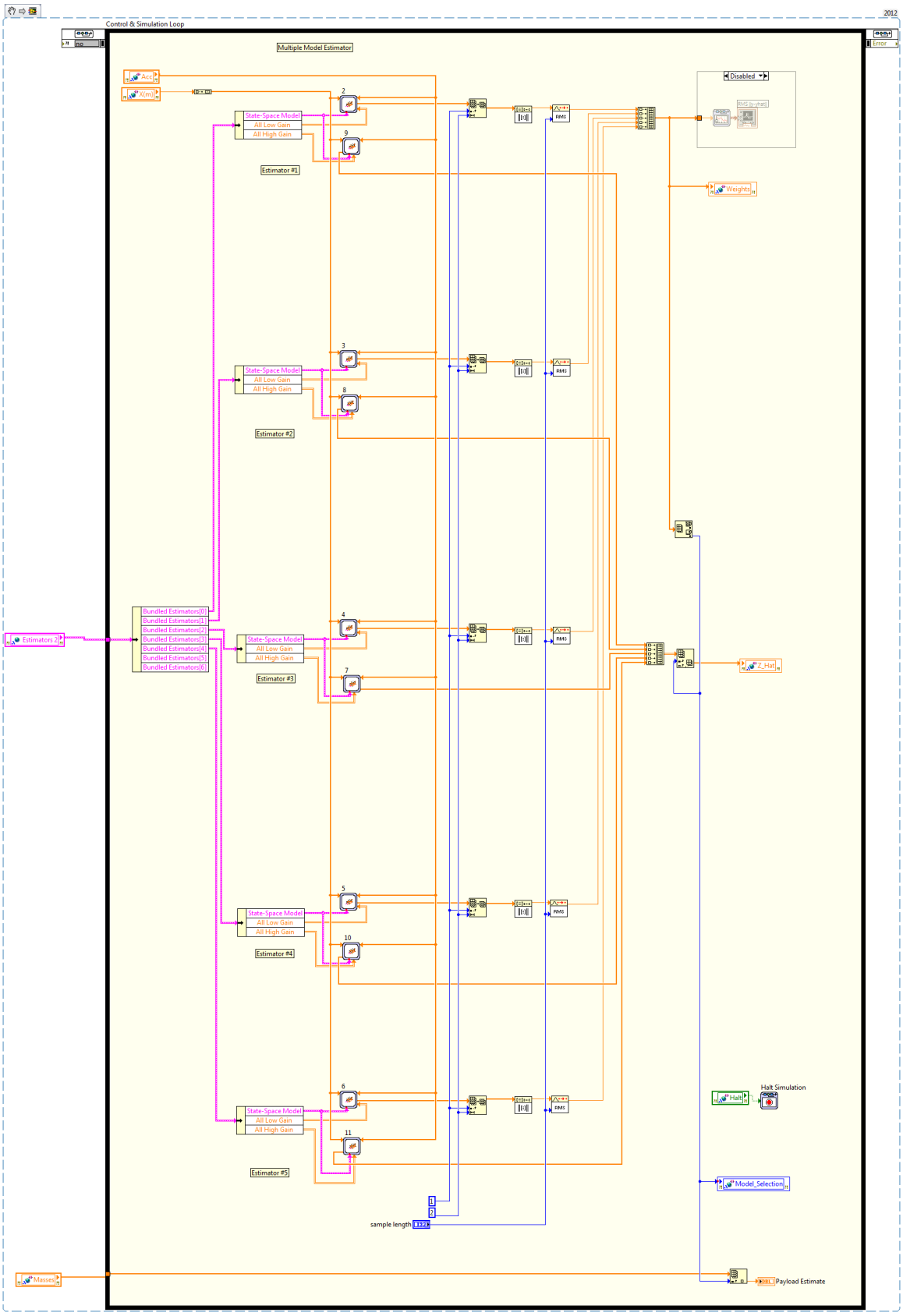


Figure 106: Multiple Model Adaptive Estimator

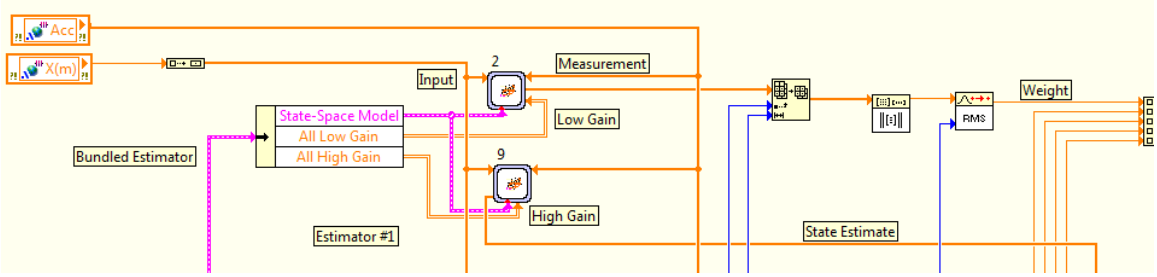


Figure 107: Sub Estimator Detailed View

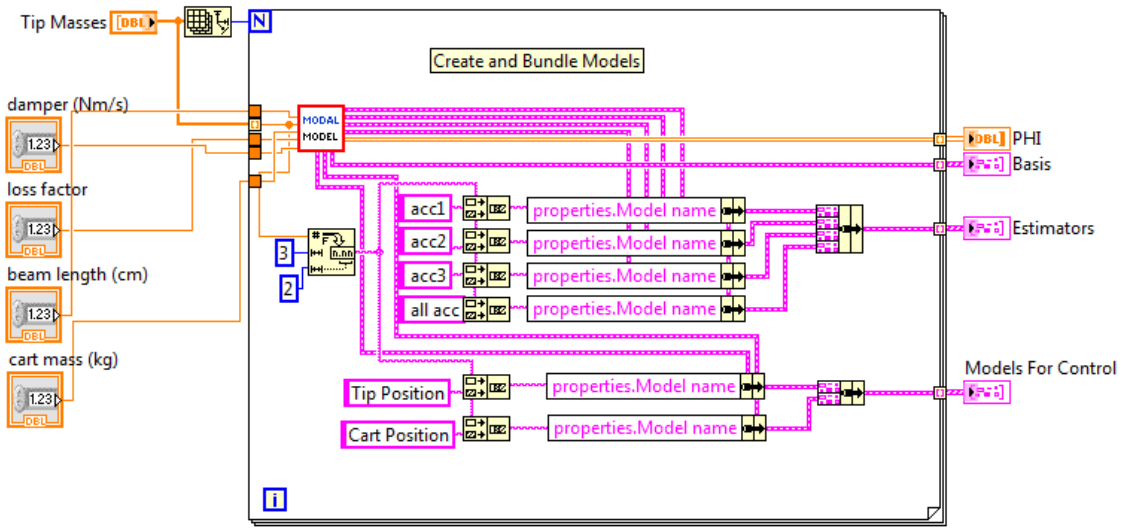


Figure 108: Model Creation and Bundling into Multiple Estimator Models

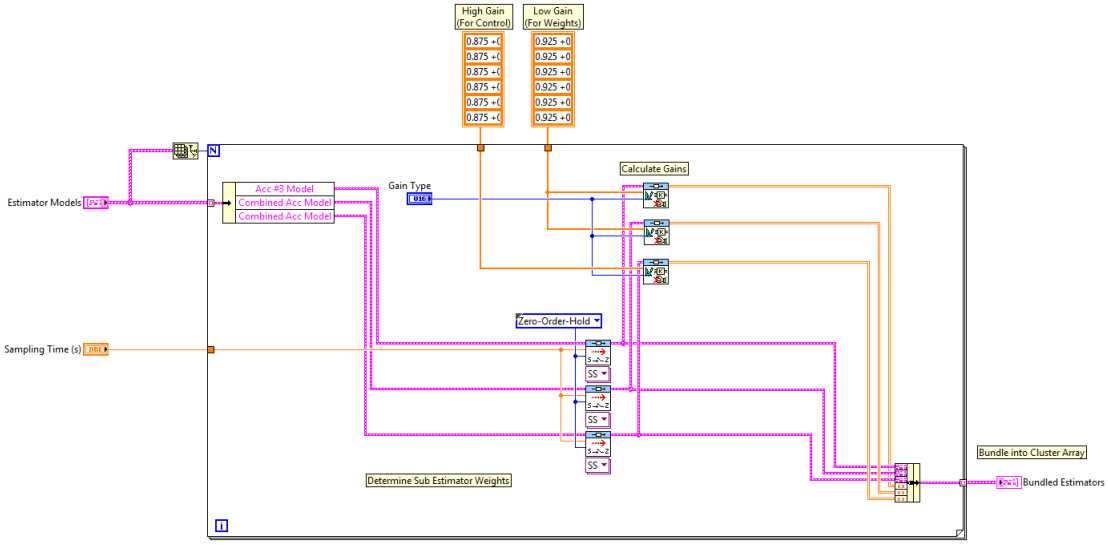


Figure 109: Sub Estimator Gain Assignment

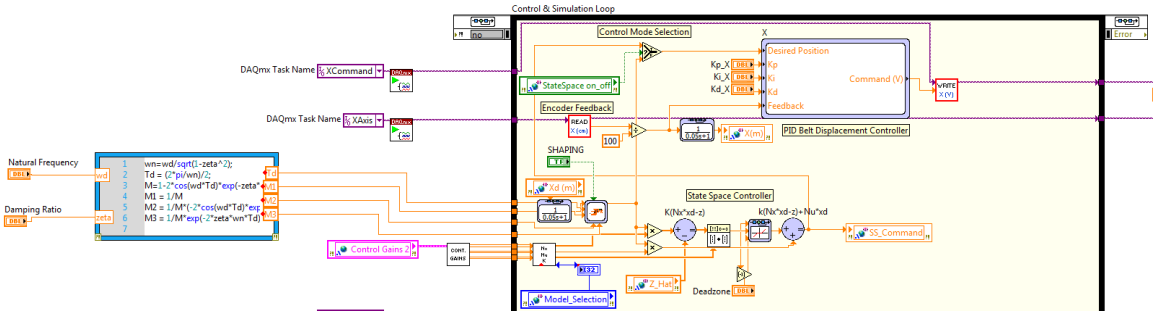


Figure 110: Multiple Model Adaptive Switching State Space Controller With Command Shaping

Appendix F

ADXL 325 - MEMS ACCELEROMETER SPECIFICATIONS

Small, Low Power, 3-Axis ±5 *g* Accelerometer

ADXL325

FEATURES

- 3-axis sensing
- Small, low profile package
  - 4 mm × 4 mm × 1.45 mm LFCSP
- Low power: 350 µA typical
- Single-supply operation: 1.8 V to 3.6 V
- 10,000 *g* shock survival
- Excellent temperature stability
- Bandwidth adjustment with a single capacitor per axis
- RoHS/WEEE lead-free compliant

APPLICATIONS

- Cost-sensitive, low power, motion- and tilt-sensing applications
  - Mobile devices
  - Gaming systems
  - Disk drive protection
  - Image stabilization
  - Sports and health devices

GENERAL DESCRIPTION

The ADXL325 is a small, low power, complete 3-axis accelerometer with signal conditioned voltage outputs. The product measures acceleration with a minimum full-scale range of ±5 *g*. It can measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration, resulting from motion, shock, or vibration.

The user selects the bandwidth of the accelerometer using the C_X, C_Y, and C_Z capacitors at the X_{OUT}, Y_{OUT}, and Z_{OUT} pins. Bandwidths can be selected to suit the application with a range of 0.5 Hz to 1600 Hz for X and Y axes and a range of 0.5 Hz to 550 Hz for the Z axis.

The ADXL325 is available in a small, low profile, 4 mm × 4 mm × 1.45 mm, 16-lead, plastic lead frame chip scale package (LFCSP_LQ).

FUNCTIONAL BLOCK DIAGRAM

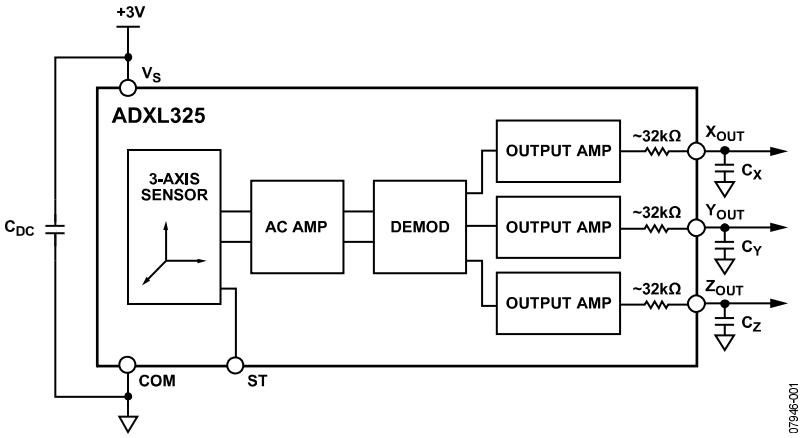


Figure 1.

Rev. 0

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 www.analog.com
Fax: 781.461.3113

SPECIFICATIONS

T_A = 25°C, V_S = 3 V, C_X = C_Y = C_Z = 0.1 µF, acceleration = 0 *g*, unless otherwise noted. All minimum and maximum specifications are guaranteed. Typical specifications are not guaranteed.

Table 1.

Parameter	Conditions	Min	Typ	Max	Unit
SENSOR INPUT	Each axis				
Measurement Range		±5	±6		*g*
Nonlinearity	Percent of full scale		±0.2		%
Package Alignment Error			±1		Degrees
Interaxis Alignment Error			±0.1		Degrees
Cross-Axis Sensitivity[1]			±1		%
SENSITIVITY (RATIOMETRIC)[2]	Each axis				
Sensitivity at X_{OUT}, Y_{OUT}, Z_{OUT}	V_S = 3 V	156	174	192	mV/*g*
Sensitivity Change Due to Temperature[3]	V_S = 3 V		±0.01		%/°C
ZERO *g* BIAS LEVEL (RATIOMETRIC)					
0 *g* Voltage at X_{OUT}, Y_{OUT}, Z_{OUT}	V_S = 3 V	1.3	1.5	1.7	V
0 *g* Offset vs. Temperature			±1		m*g*/°C
NOISE PERFORMANCE					
Noise Density X_{OUT}, Y_{OUT}, Z_{OUT}			250		µ*g*/√Hz rms
FREQUENCY RESPONSE[4]					
Bandwidth X_{OUT}, Y_{OUT}[5]	No external filter		1600		Hz
Bandwidth Z_{OUT}[5]	No external filter		550		Hz
R_{FILT} Tolerance			32 ± 15%		kΩ
Sensor Resonant Frequency			5.5		kHz
SELF TEST[6]					
Logic Input Low			+0.6		V
Logic Input High			+2.4		V
ST Actuation Current			+60		µA
Output Change at X_{OUT}	Self test 0 to 1	−90	−190	−350	mV
Output Change at Y_{OUT}	Self test 0 to 1	+90	+190	+350	mV
Output Change at Z_{OUT}	Self test 0 to 1	+90	+320	+580	mV
OUTPUT AMPLIFIER					
Output Swing Low	No load		0.1		V
Output Swing High	No load		2.8		V
POWER SUPPLY					
Operating Voltage Range		1.8		3.6	V
Supply Current	V_S = 3 V		350		µA
Turn-On Time[7]	No external filter		1		ms
TEMPERATURE					
Operating Temperature Range		−40		+85	°C

[1] Defined as coupling between any two axes.
[2] Sensitivity is essentially ratiometric to V_S.
[3] Defined as the output change from ambient-to-maximum temperature or ambient-to-minimum temperature.
[4] Actual frequency response controlled by user-supplied external filter capacitors (C_X, C_Y, C_Z).
[5] Bandwidth with external capacitors = 1/(2 × π × 32 kΩ × C). For C_X, C_Y = 0.003 µF, bandwidth = 1.6 kHz. For C_Z = 0.01 µF, bandwidth = 500 Hz. For C_X, C_Y, C_Z = 10 µF, bandwidth = 0.5 Hz.
[6] Self test response changes cubically with V_S.
[7] Turn-on time is dependent on C_X, C_Y, C_Z and is approximately 160 × C_X or C_Y or C_Z + 1 ms, where C_X, C_Y, C_Z are in µF.

REFERENCES

[1] Abdollahi, F., Talebi, H. A., and Patel, R. V., "A stable neural network-based observer with application to flexible-joint manipulators," *Neural Networks, IEEE Transactions on*, vol. 17, no. 1, pp. 118–129, 2006.

[2] Alberts, T. E., Dickerson, S. L., and Book, W. J., "Modeling and control of flexible manipulators," 1 June 1985.

[3] Alberts, T. E., Dickerson, S. L., and Book, W. J., "On the transfer function modeling of flexible structures with distributed damping," December 1986.

[4] Asada, H. and Ma, Z.-D., "Inverse dynamics of flexible robots; modeling and recursive computation using virtual rigid link coordinate systems," in *American Control Conference, 1989*, pp. 2352–2359, IEEE.

[5] Athans, M. and Chang, C.-B., "Adaptive estimation and parameter identification using multiple model estimation algorithm," tech. rep., DTIC Document, 1976.

[6] Balas, M. J., "Feedback control of flexible systems," *Automatic Control, IEEE Transactions on*, vol. 23, no. 4, pp. 673–679, 1978.

[7] Bartolini, G., Pisano, A., Punta, E., and Usai, E., "A survey of applications of second-order sliding mode control to mechanical systems," *International Journal of Control*, vol. 76, no. 9, pp. 875 – 892, 2003.

[8] Beargie, A. M. and Book, W. J., "Sliding mode control of a non-collocated flexible system," 11 November 2003.

[9] Book, W. J., *Modeling, Design and Control of Flexible Manipulator Arms*. PhD thesis, 1974.

[10] Book, W. J., "Dynamic analysis of a class of mixed lumped-distributed parameter systems via numerical techniques," July 1976.

[11] Book, W. J., "Analysis of massless elastic chains with servo controlled joints," *Journal of Dynamic Systems*, vol. 101, pp. 178–192, 1979.

[12] Book, W. J., "Recursive lagrangian dynamics of flexible manipulator arms via transformation matrices," tech. rep., Carnegie-Mellon University, 1983.

[13] Book, W. J., "Modeling design, and control of flexible manipulator arms: A tutorial review," *29th IEEE Conference on Decision and Control*, pp. 500 – 506, 1990.

[14] Book, W. J., "Controlled motion in an elastic world," *ASME Journal of Dynamic Systems Measurement and Control*, vol. 115, no. 2, p. 252, 1993.

[15] Book, W. J., Maizza-Neto, O., and Whitney, D., "Feedback control of two beam, two joint systems with distributed flexibility," *American Society of Mechanical Engineers, Winter Annual Meeting*, vol. 75-WA/Aut-26, 1975.

[16] Book, W. J. and Majette, M., "A comparison of natural frequency prediction methods for flexible manipulator arms," November 16-21 1980.

[17] Book, W. J. and Majette, M., "Controller design for flexible distributed parameter mechanical arms via combined state space and frequency domain techniques," *Journal of Dynamic Systems Measurement and Control*, vol. 105, no. 4, pp. 245 – 249, 1983.

[18] Botsali, F. M., Kalyoncu, M., Tinkir, M., and Onen, U., "Fuzzy logic trajectory control of flexible robot manipulator with rotating prismatic joint," 26-28 Feb. 2010 2010.

[19] Bruant, I., Gallimard, L., and Nikoukar, S., "Optimal piezoelectric actuator and sensor location for active vibration control, using genetic algorithm," *Journal of sound and vibration*, vol. 329, no. 10, pp. 1615–1635, 2010.

[20] Cetinkunt, S. and Book, W. J., "Performance of lightweight manipulators under joint variable feedback control: Analytical study of limitations," 11 June 1988.

[21] Cetinkunt, S. and Book, W. J., "Symbolic modeling and dynamic simulation of robotic manipulators with compliant links and joints," *Journal of Robotics and Computer Integrated Manufacturing*, vol. 5, no. 4, pp. 301 – 310, 1989.

[22] Cetinkunt, S. and Book, W. J., "Flexibility effects on the control system performance of large scale robotic manipulators," *Journal of Astronomical Sciences*, vol. 38, no. 4, pp. 531 – 556, 1990.

[23] Chalhoub, N. G. and Kfoury, G. A., "Development of a robust nonlinear observer for a single-link flexible manipulator," *Nonlinear Dynamics*, vol. 39, no. 3, pp. 217–233, 2005.

[24] Chen, F. and Dunnigan, M., "Comparative study of a sliding-mode observer and kalman filters for full state estimation in an induction machine," 2002.

[25] Chen, W., "Dynamic modeling of multi-link flexible robotic manipulators," *Computers & Structures*, vol. 79, no. 2, pp. 183–195, 2001.

[26] De Luca, A., "Feedforward/feedback laws for the control of flexible robots," in *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, vol. 1, pp. 233–240 vol.1.

[27] De Luca, A. and Book, W. J., *Robots With Flexible Elements*, pp. 287–319. Berlin: Springer, 2008.

[28] De Luca, A. and Flacco, F., "A pd-type regulator with exact gravity cancellation for robots with flexible joints," May 9-13 2011.

[29] Debus, T., Dupont, P., and Howe, R., *Contact state estimation using multiple model estimation and hidden markov models*, pp. 517–526. Springer, 2003.

[30] Desoer, C. A., "Slowly varying discrete system xi+1=aixi," *Electronics Letters*, vol. 6, no. 11, pp. 339–340, 1970.

[31] Dian, S., Arai, Y., and Gao, W., "Dynamic compensation of modeling uncertainties and disturbances of a precision planar motion stage based on sliding mode observer," *Int J Adv Manuf Technol*, vol. 46, pp. 899–912, 2010.

[32] Elbeheiry, E. M. and Elmaraghy, H. A., "Robotic manipulators state observation via one-time gain switching," *Journal of Intelligent and Robotic Systems*, vol. 38, pp. 313 – 344, 2003.

[33] Feliu, V. and Ramos, F., "Strain gauge based control of single-link flexible very lightweight robots robust to payload changes," *Mechatronics*, vol. 15, no. 5, pp. 547–571, 2005.

[34] Feliu, V., Rattan, K. S., and Brown, H. B., J., "Adaptive control of a single-link flexible manipulator," *Control Systems Magazine, IEEE*, vol. 10, no. 2, pp. 29–33, 1990.

[35] Forbes, J. R. and Damaren, C. J., "Single-link flexible manipulator control accommodating passivity violations: Theory and experiments," *Control Systems Technology, IEEE Transactions on*, vol. 20, no. 3, pp. 652–662, 2012.

[36] Furuhashi, T., Sangwongwanich, S., and Okuma, S., "A position-and-velocity sensorless control for brushless dc motors using an adaptive sliding mode observer," *Industrial Electronics, IEEE Transactions on*, vol. 39, no. 2, pp. 89–95, 1992.

[37] Gawronski, W. and Lim, K., "Balanced actuator and sensor placement for flexible structures," *International Journal of Control*, vol. 65, no. 1, pp. 131–145, 1996.

[38] Georges, D., "The use of observability and controllability gramians or functions for optimal sensor and actuator location in finite-dimensional systems," in *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*, vol. 4, pp. 3319–3324, IEEE.

[39] Ginsberg, J. H., *Mechanical and Structural Vibrations*. New York: John Wiley and Sons, inc., 2001.

[40] Gurses, K., Buckham, B. J., and Park, E. J., "Vibration control of a single-link flexible manipulator using an array of fiber optic curvature sensors and pzt actuators," *Mechatronics*, vol. 19, no. 2, pp. 167–177, 2009.

[41] Gustafsson, F., *Adaptive filtering and change detection*, vol. 1. Wiley New York, 2000.

[42] Hastings, G. G. and Book, W. J., "Reconstruction and robust reduced-order observation of flexible variables," December 1986.

[43] Hastings, G. G. and Book, W. J., "A linear dynamic model for flexible robotic manipulators," *Control Systems Magazine, IEEE*, vol. 7, no. 1, pp. 61–64, 1987.

[44] Hiramoto, K., Doki, H., and Obinata, G., "Optimal sensor/actuator placement for active vibration control using explicit solution of algebraic riccati equation," *Journal of Sound and Vibration*, vol. 229, no. 5, pp. 1057–1075, 2000.

[45] Hyun, C.-H., Park, C.-W., and Kim, S., "Takagisugeno fuzzy model based indirect adaptive fuzzy observer and controller design," *Information Sciences*, vol. 180, no. 11, pp. 2314–2327, 2010.

[46] Jung, J., Huh, K., Fathy, H. K., and Stein, J. L., "Optimal robust adaptive observer design for a class of nonlinear systems via an h-infinity approach," in *American Control Conference, 2006*, p. 6 pp., IEEE.

[47] Kalman, R. E., "A new approach to linear filtering and prediction problems," *Transactions of the ASMEJournal of Basic Engineering*, vol. 82 (Series D), pp. 35–45, 1960.

[48] Kammer, D. C., "Effect of model error on sensor placement for on-orbit modal identification of large space structures," *Journal of guidance, control, and dynamics*, vol. 15, no. 2, pp. 334–341, 1992.

[49] Kammer, D. and Yao, L., "Enhancement of on-orbit modal identification of large space structures through sensor placement," *Journal of Sound and Vibration*, vol. 171, no. 1, pp. 119–139, 1994.

[50] Karandikar, D. and Bandyopadhyay, B., "Sliding mode control of single link flexible manipulator," 19-22 Jan. 2000.

[51] Khoshnevis, B., "Automated construction by contour crafting–related robotics and information technologies," *Automation in Construction*, vol. 13, no. 1, pp. 5–19, 2004.

[52] Kim, M.-H. and Inman, D. J., "Reduction of observation spillover in vibration suppression using a sliding mode observer," *Journal of Vibration and Control*, vol. 7, no. 1087, p. 20, 2001.

[53] Krauss, R., *An Improved Technique for Modeling and Control of Flexible Structures*. PhD thesis, 2006.

[54] Krener, A. J. and Ide, K., "Measures of unobservability," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pp. 6401–6406, IEEE.

[55] Kwon, D. S. and Book, W. J., "A time domain inverse dynamic tracking control of a single-link flexible manipulator," *Journal of Dynamic Systems, Measurement, and Control*, vol. 116, no. 2, 1994.

[56] Lashlee Jr, R. W. and Maybeck, P. S., "Space structure control using moving bank multiple model adaptive estimation," in *Decision and Control, 1988., Proceedings of the 27th IEEE Conference on*, pp. 712–717, IEEE.

[57] Leahy Jr, M. and Tellman, L., "Multiple model-based control of robotic manipulators: Theory and simulation," in *Intelligent Control, 1989. Proceedings., IEEE International Symposium on*, pp. 53–58, IEEE.

[58] Lee, J. W. and Huggins, J. D., "Experimental verification of a large flexible manipulator," 11 June 1988.

[59] Lee, J. W. and Book, W. J., "Efficient dynamic models for flexible robots," 1 May 1989.

[60] Lertpiriyasuwat, V., Berg, M. C., and Buffinton, K. W., "Extended kalman filtering applied to a two-axis robotic arm with flexible links," *The International Journal of Robotics Research*, vol. 19, no. 254, 2000.

[61] Lew, J. Y. and Book, W. J., "Hybrid control of flexible manipulators with multiple contacts," 1993.

[62] Lewis, F. L., Xie, L., and Popa, D., *Optimal and Robust Estimation With an Introduction to Stochastic Control Theory*. 2nd ed., 2008.

[63] Li, Y. and Wang, G., "Sensing strategies for a flexible manipulator," *Instrumentation and Measurement, IEEE Transactions on*, vol. 49, no. 3, pp. 565–572, 2000.

[64] Liu, P., Rao, V. S., and Derriso, M., "Active control of smart structures with optimal actuator and sensor locations," *Proceedings of SPIE Smart Structures and Materials: Modeling, Signal Processing and Control*, vol. 4693, pp. 1–12, 2002.

[65] Luenberger, D., "An introduction to observers," *Automatic Control, IEEE Transactions on*, vol. 16, no. 6, pp. 596–602, 1971.

[66] Magee, D. P. and Book, W. J., "The application of input shaping to a system with varying parameters," 1992.

[67] Magee, D. P. and Book, W. J., "Control and control theory for flexible robots," *Journal of the Society of Instrument and Control Engineers*, vol. 32, no. 4, pp. 309 – 317, 1993.

[68] Maghami, P. G. and Joshi, S. M., "Sensor/actuator placement for flexible space structures," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 29, no. 2, pp. 345–351, 1993.

[69] Martinez, J. I. M., Sawut, U., and Nakano, K., "Application of non-linear observer with simultaneous perturbation stochastic approximation method to single flexible link smc," in *SICE Annual Conference, 2008*, pp. 2150–2155.

[70] Martinez, J., Nakano, K., and Umerujan, S., "Vibration suppression control of a flexible arm using a nonlinear observer with simultaneous perturbation stochastic approximation," *Artificial Life and Robotics*, vol. 14, no. 4, pp. 539–544, 2009.

[71] Meirovitch, L., *Elements of Vibration Analysis*. McGraw-Hill, Inc., 2nd ed., 1986.

[72] Menke, T. E. and Maybeck, P. S., "Sensor/actuator failure detection in the vista f-16 by multiple model adaptive estimation," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 31, no. 4, pp. 1218–1229, 1995.

[73] Murray-Smith, R. and Johansen, T. A., *Multiple Model Approaches to Modeling and Control*. Systems and Control Book Series, London: Taylor and Francis, 1997.

[74] Obergfell, K. and Book, W. J., "End-point position measurement of long-reach flexible manipulators," 11 September 1994.

[75] Obergfell, K. and Book, W. J., "Control of flexible manipulators using vision and modal feedback," 21 May 1999.

[76] Pereira, E., Becedas, J., Payo, I., Ramos, F., and Feliu, V., "Control of flexible manipulators. theory and practice," *Robot Manipulators Trends and Development, InTech, Croatia*, pp. 267–296, 2010.

[77] Pestle, E. and Leckie, F., "Matrix methods in elastomechanics," *MaGraw-Hill, New York*, 1963.

[78] Petersen, J. R. and McFarlane, D. C., "Robust state estimation for uncertain systems," in *Decision and Control, 1991., Proceedings of the 30th IEEE Conference on*, pp. 2630–2631 vol.3.

[79] Post, B. K. and Book, W. J., "A robust observation strategy for the control of flexible manipulators," May 2011.

[80] Post, B. K., Mariuzza, A., Book, W. J., and Singhose, W., "Flatness-based control of flexible motion systems," October 2011.

[81] Quinlan, M. J. and Middleton, R. H., *Multiple model kalman filters: A localization technique for robocup soccer*, pp. 276–287. Springer, 2010.

[82] Rao, S. S., Pan, T.-S., and Venkayya, V. B., "Optimal placement of actuators in actively controlled structures using genetic algorithms," *AIAA journal*, vol. 29, no. 6, pp. 942–943, 1991.

[83] Rong Li, X. and Jilkov, V. P., "Survey of maneuvering target tracking. part v. multiple-model methods," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 41, no. 4, pp. 1255–1321, 2005.

[84] Rugh, W. J., "Analytical framework for gain scheduling," *Control Systems, IEEE*, vol. 11, no. 1, pp. 79–84, 1991.

[85] Sasaki, M., Asai, A., Shimizu, T., and Ito, S., "Self-tuning control of a two-link flexible manipulator using neural networks," in *ICCAS-SICE, 2009*, pp. 2468–2473.

[86] Sciavicco, L. and Siciliano, B., *Modelling and control of robot manipulators*. Springer Verlag, 2000.

[87] Shawky, A., Ordys, A., and Grimble, M. J., "End-point control of a flexible-link manipulator using hinfinity nonlinear control via a state-dependent riccati equation," 2002.

[88] Singh, A. K. and Hahn, J., "Determining optimal sensor locations for state and parameter estimation for stable nonlinear systems," *Industrial & engineering chemistry research*, vol. 44, no. 15, pp. 5645–5659, 2005.

[89] Singhose, W., Eloundou, R., and Lawrence, J., "Command generation for flexible systems by input shaping and command smoothing," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 6, 2010.

[90] Singhose, W. and Seering, W., *Command Generation for Dynamic Systems*. ISBN 978-0-9842210-0-4, 2009.

[91] Spanos, D. P., Olfati-Saber, R., and Murray, R. M., "Dynamic consensus on mobile networks," 2005.

[92] Speyer, J. L. and Chung, W. H., *Stochastic Processes, Estimation, and Control*. Philadelphia: Society for Industrial and Applied Mathematics, 2008.

[93] Spong, M. W., Hutchinson, S., and Vidyasagar, M., *Robot modeling and control*. John Wiley & Sons New York, 2006.

[94] Staufer, P. and Gattringer, H., "State estimation on flexible robots using accelerometers and angular rate sensors," *Mechatronics*, 2012.

[95] Su, Y., Miller, P. C., and Zheng, C., "A simple nonlinear observer for a class of uncertain mechanical systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1340–1345, 2007.

[96] Szidarovszky, F. and Bahill, A. T., *Linear Systems Theory*. Boca Raton, Florida: CRC Press, 2nd ed., 1998.

[97] Tondu, B., Boitier, V., and Lopez, P., "Naturally compliant robot-arms actuated by mckibben artificial muscles," 2-5 Oct 1994.

[98] Tongco, E. and Meldrum, D., "Optimal sensor placement of large flexible space structures," *Journal of guidance, control, and dynamics*, vol. 19, no. 4, pp. 961–963, 1996.

[99] Tsujisawa, T. and Book, W. J., "A reduced order model derivation for lightweight arms with a parallel mechanism," in *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*, pp. 728–735 vol.2.

[100] Tumari, M., Ahmad, M., Saealal, M., Zawawi, M., Mohamed, Z., and Yusop, N., "The direct strain feedback with pid control approach for a flexible manipulator: Experimental results," in *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*, pp. 7–12, IEEE.

[101] Van Der Smagt, P., Groen, F., and Schulten, K., "Analysis and control of a rubbertuator arm," *Biological Cybernetics*, vol. 75, no. 5, p. 433, 1996. Biological Cybernetics Source Information: 1996, Vol. 75 Issue 5, p433; Subject Term: ROBOTICS; Subject Term: ELECTRONIC control; Number of Pages: 8p; Document Type: Article.

[102] Veluvolu, K., Soh, Y., and Cao, W., "Robust observer with sliding mode estimation for nonlinear uncertain systems," *IET Control Theory Appl.*, vol. 1, no. 5, p. 15331540, 2007.

[103] Walcott, B. and Zak, S., "State observation of nonlinear uncertain dynamical systems," *Automatic Control, IEEE Transactions on*, vol. 32, no. 2, pp. 166–170, 1987.

[104] Walcott, B. L. and Zak, S. H., "Observation of dynamical systems in the presence of bounded nonlinearities/uncertainties," in *Decision and Control, 1986 25th IEEE Conference on*, vol. 25, pp. 961–966.

[105] Wit, C. C. D. and Slotine, J.-J. E., "Sliding observers for robot manipulators," *Automatica*, vol. 27, no. 5, pp. 859–864, 1991.

[106] Xiao, L., Boyd, S., and Lall, S., "A scheme for robust distributed sensor fusion based on average consensus," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pp. 63–70, IEEE.

[107] Xie, L., Yeng Chai, S., and de Souza, C. E., "Robust kalman filtering for uncertain discrete-time systems," *Automatic Control, IEEE Transactions on*, vol. 39, no. 6, pp. 1310–1314, 1994.

[108] Xu, Y. and Ritz, E., "Vision based flexible beam tip point control," *Control Systems Technology, IEEE Transactions on*, vol. 17, no. 5, pp. 1220–1227, 2009.

[109] Yin, H., Kobayashi, Y., Hoshino, Y., and Emaru, T., "Hybrid sliding mode control with optimization for flexible manipulator under fast motion," May 9-13 2011.

[110] Yoshikawa, T., Ohta, A., and Kanaoka, K., "State estimation and parameter identification of flexible manipulators based on visual sensor and virtual joint model," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 3, pp. 2840–2845, IEEE.

[111] Zaki, A., ElBeheiry, E., and ELMaraghy, W., "Variable structure observer design for flexible link manipulator control," *Transactions of the Canadian Society for Mechanical Engineering*, vol. 27, no. 1-2, pp. 107–129, 2003.

[112] Zhu, X., Soh, Y. C., and Xie, L., "Robust kalman filter design," in *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, vol. 4, pp. 3813–3818, IEEE.

VITA

Brian K. Post originally hails from Powell, Tennessee. He earned his bachelors degree in mechanical engineering from Purdue University in West Lafayette, Indiana in 2007 with a focus in surgical robotics. In the summers of 2006 and 2007, Brian was an intern for the Robotics and Energetic Systems Group at the Oak Ridge National Laboratory. Motivated by his experiences at the lab, he decided to pursue a Ph.D. in the robotics field with the Intelligent Machine Dynamics Laboratory at Georgia Tech.

In 2010 Brian earned his masters degree in mechanical engineering from the Georgia Institute of Technology. Brian married his high school sweetheart Sarah Beam in the spring of 2012. It remains the best thing he's ever done. Brian and Sarah will be moving back to Knoxville, Tennessee where Brian will be a postdoctoral fellow at Oak Ridge National Laboratory with the Automation Robotics and Manufacturing Group and Sarah will be a practicing audiologist.