MODELING, ESTIMATION AND CONTROL FOR SERIAL FLEXIBLE ROBOT ARMS

A Dissertation
Presented to
The Academic Faculty

by

Arto Kivila

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

Georgia Institute of Technology
August 2017

Copyright © 2017 by Arto Kivila

MODELING, ESTIMATION AND CONTROL FOR SERIAL FLEXIBLE ROBOT ARMS

Approved by:

Professor William Singhose, Advisor
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Professor Jun Ueda
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Wayne J. Book
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Professor Joshua Vaughan
Department of Mechanical Engineering
University of Louisiana at Lafayette

Professor Nader Sadegh
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Date Approved: 19 April 2017

ACKNOWLEDGEMENTS

I want to thank Dr William Singhose and Dr Wayne Book for the tireless support during the many years it took me to finish my degree.

I want to thank Dr Nader Sadegh, Dr Jun Ueda and Dr Josh Vaughan for their service on my dissertation committee and for the helpful comments.

I want to acknowledge the SSDSL and Dr Stephen Leadenham who helped me conduct shaker experiments a let me use their equipment. The help from JD Huggins, Jonah Burgin and Bumsoo Lee was invaluable during the design and construction phase of the test bed.

I want to thank all colleagues including: CJ Adams, Ali AlSaibie, Dr Martin Cacan, Dr Hannes Daepp, Dr Stephen Leadenham, Wayne Maxwell, Kyle Motter, Dr Heather Paxinos, Franziska Schlagenhauf and Ellen Skow for always being there to bounce ideas off of and to give their suggestions.

I would have not been able to finish this degree without the support of National Instruments, Boeing, and the Woodruff School of Mechanical Engineering through various projects.

Contents

List of Tables

List of Figures

SUMMARY

Industry demands for high precision automation equipment have led to heavy, stiff and therefore expensive, inefficient, and potentially dangerous serial link manipulators. Industry has been reluctant to move towards lighter, and therefore inherently more flexible manipulators, despite the potential benefits of lower costs, increased throughput and improved safety. Advancements in data processing capabilities, sensing capabilities and control theory during the past couple of decades can potentially solve the perceived problems with flexible serial manipulators.

In general, there is no such thing as a fully rigid manipulator: even current industrial robots exhibit small flexibilities. In addition, there are applications such as space-robotics and nuclear waste tank inspection/cleaning jobs where light and long links are the only option. Most research in the past has focused on single link manipulators and planar robot arms rather than spatial multi-link robots.

This dissertation presents a systematic approach for obtaining natural frequencies and mode-shapes for n-link spatial serial structures based on transfer matrices. The method is validated with experiments and software simulations. A low-order dynamical model for n-link flexible manipulators in spatial configurations is presented. The model is verified with finite element simulations, and hardware experiments.

The low-order model is the basis of an extended Kalman filter based estimator that allows sensor-based predictions of the flexible states. Accelerometer and strain gage based feedback is examined. Accelerometer based feedback is verified with experiments.

In order to damp out he oscillations multi-link flexible arms caused by the reference command, an optimized input shaping algorithm for multiple frequency ranges is presented. The results are confirmed with FEA analysis and experiments. The controllability of natural modes is discussed and analyzed. An inversion based closed-loop controller is presented that guarantees stable joint trajectory tracking for flexible manipulator arms. A singular

pertubation based controller is presented to actively damp out the vibrations in the arm.

A test bed that provided verification of the claims made in this dissertation was designed and constructed. The test bed has 3 actuators and 2 flexible links.

The main contributions of the dissertation are therefore:

- A systematic extension of the transfer matrix method for n-link spatial serial structures
- A low order model for flexible serial manipulator based on exact modes of the system
- Development of a 2-link, 3-joint flexible manipulator testbed
- An extended Kalman filter based estimator for flexible states based on strain and acceleration feedback
- An optimized input-shaping method based for flexible manipulators
- Modal accessibility analysis for serial flexible manipulators
- An inversion based and a singular pertubation based closed-loop controller for flexible manipulators

Chapter I

INTRODUCTION

1.1 Motivation For Research

Industry demands for high precision automation equipment led to heavy, stiff and thus expensive, inefficient, and potentially dangerous serial link manipulators. Industry has been reluctant to move towards lighter, and therefore inherently more flexible manipulators, despite great potential benefits of lower costs, increased throughput and improved safety.

In general, there is no such thing as a fully rigid manipulator and even current industrial robots exhibit some flexibility [65]. The majority of previous research on flexible robot arms has focused on single-link manipulators and planar robot arms. The class of more complex, yet more useful spatial multi-link robots has been neglected to some extent. The general view is that multi-link robots can be treated as a simple combination of single-link arms.

There are very few examples of flexible (by design) manipulators used in industry. An example application where it is vital for the mechanical structure to be light with a large workspace is space robotics. Weight must be kept to a minimum in order for them to be launched into outer space. To date, such serial arms have not used modern control techniques to reduce structural vibrations; instead they reduce the control bandwidth and have used the damping properties of the structure to slowly damp the oscillations. For example, it has been estimated that about 10 cumulative hours (about 30 % of total operational time) were spent waiting for the oscillations to damp down on the Space Shuttle remote manipulator system during the 15 flights it took to build the Space Station Freedom [115].

Flexible Serial Manipulator arms would be beneficial to a broad range of industrial applications if active and passive vibration could enable robots to be lighter, and therefore cheaper, and faster. Additional benefits include the machines being safer for humans in the workspace due to lower inertia forces.

Reducing the mass of the the robot arm could improve the performance by shifting more actuator power to moving the payload, rather than the arm structure itself. This would allow the motors to move faster, and reduce cycle times. Additionally, if the arms were made lighter per unit length then the reach of the robot can be increased to have a greater workspace. This can also allow fewer robots to be installed in a given workspace, or they can move larger objects.

If the robot arms are lighter, the foundation, to which they are attached, does not need to be as strong. This reduced requirement would decrease the cost of building factories. Similarly, accurate robot arms could be installed on moving platforms. Given that less power is being used to move the robot arms, energy costs to run the robot would also decrease. The costs to build the robot could potentially be lower as well, since less high-performance actuators would need to be used along with the accompanying power electronics.

The large mass of robot arms pose a real danger to humans or other objects in the workspace due to risks of collision. Research by the German Aerospace Center (DLR) has illustrated that collisions with current robot arms could very easily be deadly [61]. To reduce this risk, humans are generally excluded from being in the direct vicinity of robot manipulators. Reducing the weight of the arm and making the system more compliant, would enable use in more fragile environments, and even in the vicinity of humans.

In general, robots with flexible links are much harder to control compared to ones with rigid links. Even though oscillations in systems where the model can be considered linear, such as cranes, can be controlled with high precision, this linear assumption does not extend to flexible link robots. In addition robot arms with rigid links have non-linear, yet well understood dynamics, that have been utilized for years.

Industry has not widely adopted modern control techniques for vibration control of flexible structures due to research and implementation challenges. One of the biggest obstacles still lies in obtaining high fidelity models for multi link flexible manipulators that is implementable on real-time controllers. In addition, measuring flexure so that feed-back controller could be used with sensors that are affordable and usable in a variety of industrial environments is not a solved problem. Another complexity associated with flexible serial

(a) FANUC M-2000iA/2300 (b) Kuka LBR IIWA 7 R800

Figure 1: Modern Serial Manipulators

manipulators is the control theory required to accurately control them.

Advancements in data processing capabilities, sensing capabilities and control theory during the past couple of decades could potentially solve the perceived problems with flexible serial manipulators.

1.1.1 Modern Serial Manipulators

There has been continual improvement in the mainstream serial manipulators since their introduction in the 1960s. However, with a few exceptions, the end-effector location is still determined based on the joint angles. For this approach to yield accurate end-point locations, the links need to exhibit very little compliance when actuated [15, 143]. This is achieved using strong, heavy materials, which also dictates the use of powerful, and therefore, heavy actuators. Therefore, large high-performance motors and power-electronics must be used to move these arms with acceptable performance. This also leads to higher initial costs [15].

The size of the workspace is limited in order to maintain rigidity and to allow the motors to move the links sufficiently fast. The rigidity of a beam is linear to the second moment of area, I. Therefore, to keep the arm rigid I needs to be large, which necessitates heavy arms. This is a problem due to the fact that the moment torque at a joint grows linearly with the mass but with the square of the length of the arm ($\tau = \frac{1}{3}mL^2\alpha$).

Two examples of modern serial manipulators are shown in Figure 1[1]. Figure 1a shows a serial arm that has a Payload capcity of 2300 kg, reach of 3.734 m and weight of 11,000 kg, while Figure 1b displays a serial arm that has a payload capacity of 7 kg, reach of 0.8 m and weight of 24 kg. These robots represent the range of modern manipulators out there and all of them have very limited capabilities when it comes to reach and payload capacity for their mass.

Based on a survey[2] of modern manipulator arms from major robot manufacturers such as ABB, MotoMan, Fanuc, and Kuka, it can be seen that in general industrial robots have a rated maximum payload from about 5% (usually for the robots that have a rated payload of up to 5kg) to about 20% (generally robots that have a rated payload of 165 kg+) of the robots total weight. The reach of the robots varies greatly based on the payload weight and the weight of the robot.

The large motors and drivers are expensive and require substantial energy to operate. These effects drive up the cost of building and operating manipulators [15]. Significant energy is used to move the massive structure itself around, leaving only a fraction of the power to move the payload. This means the payload must be substantially lighter than the robot itself. Due to advances in material science and production capabilities the ratio of robot mass to payload capacity has come down significantly in the past decades. However, the robots are still designed to be rigid. Therefore, there is substantial opportunity for robot arms become faster and carry heavier payloads.

1.1.2 Example Use Cases For Flexible Robot Arms

One application of robotics where mass is of utmost importance is robots that get launched into space. The arms that were mounted to the space shuttles suffered from vibration problems [115]. Currently, the only robot arm on another celestial body is on Mars, where the Curiosity rover has an arm mounted on it to take measurements, as seen in Figure 2[3]. This arm currently moves very slowly so that minimal energy is used and vibrations are not

[1] Images from www.robots.com
[2] The specifications for the robots were found on https://www.robots.com/
[3] Image from http://mars.nasa.gov/msl/multimedia/images/

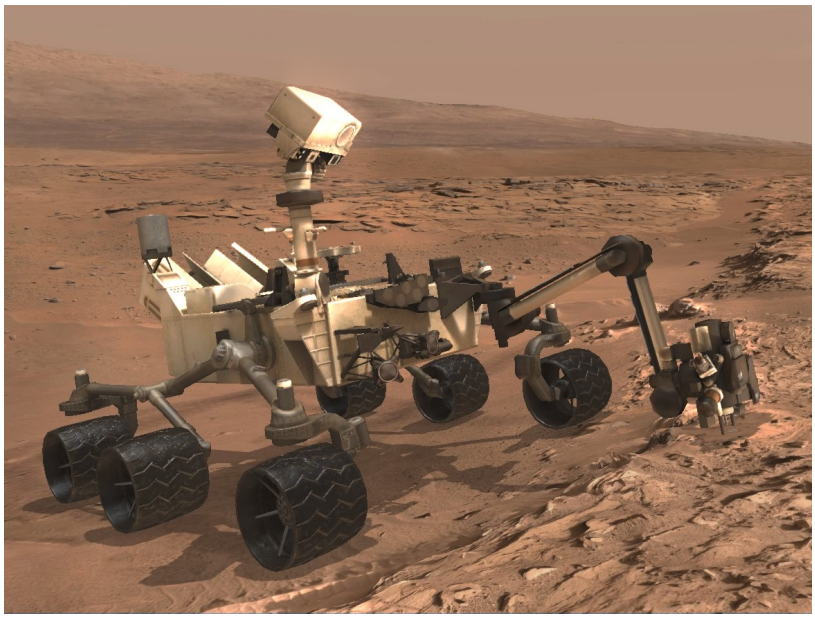


Figure 2: Curiosity Rover

excited. However, if humans attempt more substantial projects on other planets, robots will need to work faster and do more with their end-effectors.

Another example applications of FSMs is inspection and maintenance of nuclear waste tanks [46]. This task is hard to accomplish with traditional manipulators due to the large workspace (arm span of 25 m or more) and small openings through which the arms can access the tanks (between 0.1 m - 1 m) [93]. Traditional robots cannot be used for such applications due to their weight and limited reach. Additionally, there are plans are to use long reach robot arms in Fusion Reactors for maintenance and inspection work [107].

1.2 Problem Statement

Past research has not addressed some important issues with multi-link spatial flexible manipulators well. A big deficiency has been how the arms have been modeled. The research on methods to estimate flexible states of spatial arms has also been very limited. Controllers rely on a good model of the system and estimation of all of the state variables to accurately position the end-effector. Very few flexible (by design) multi-link robots, where the controller and estimation algorithms could be tested on, have been built, and therefore most work has not been validated by experimental tests.

1.3 Research Overview

The research in the dissertation is broken down into three complimentary research problems. First, a suitable modeling strategy for flexible serial arms is developed. Second, an estimation algorithm that can quickly, and robustly, provide the states to a feedback controller is formulated. Finally, a controller that allows for the flexible arm to move without significant oscillations is derived.

1.3.1 Modeling Flexible Manipulators

While modeling serial robots and flexible bodies are by themselves an area that is well established, work on multi-link flexible bodies is much less developed. The models for single link flexible arms are fairly straight forward and can, in many cases, be assumed to be linear.

The mathematical models are most often derived from energy based methods. The resulting equations governing multi-link flexible robot arms are nonlinear, coupled, ordinary and partial differential equations. Most commonly the finite dimensional dynamic equations are derived with the Assumed Modes Method (AMM) or the Finite Element Method (FEM). Use of exact infinite dimensional models are not practical in the real world.

The FEM is a fairly straight forward method and has been proven to give fairly accurate results. The big problem with this method is that numerous boundary conditions have to be considered, which makes the model hard to realize in real-time.

The AMM is a much more elegant method that uses modal amplitudes as the flexible state variables. The major hurdle with this method is that when the arm configuration changes, the mode-shapes change as well. The problem lies with recomputing the modes and making sure that the model is consistent from one configuration to the next. In addition, methods for finding a good basis of modes in spatial configurations has not been well established.

The modeling method used in this dissertation makes an extension to the Transfer Matrix Method (TMM) to utilize the natural modes for any spatial serial structure. These modes are then used in a dynamical model, that was derived using energy methods, where the natural modes associated with each configuration have to be scaled and ordered properly to make

the model continuous. The TMM and the dynamical model are then both independently verified.

1.3.2 Estimation for Flexible Manipulators

Although open-loop controllers by definition work without knowing the internal states of the dynamical model, they cannot, guarantee performance when the model of the system is not known well or when disturbances exist. For closed-loop control to work estimates of states are required.

Serial robot arms typically have sensors at the joints, which are sufficient to use for full state feedback in the rigid link case. However, with these sensors it is very difficult, or even impossible, to sense the flexibilities in the system. This deficiency in turn, leads to poor end-effector position accuracy. Flexible states themselves are generally not directly measurable, therefore estimates of the flexible states are needed for the accurate positioning of the end-effector. However, accurate and efficient sensing of the flexible states generally requires a good model of the system.

Since the dynamical models for flexible robot arms are nonlinear, the estimators for the flexible coordinates need to be nonlinear as well. Unfortunately, nonlinear estimation is much less researched when compared to the linear counter part. Extended Kalman filtering was chosen as the estimator in this work due to the inherent capability to deal with both noise in the system and noise in the measurements.

Strain gages are commonly used for estimation of flexible variables. However, strain gages require careful conditioning and signal amplification. In addition, estimating flexibilities in 3D would make the placement of strain gages very difficult, which is amplified further when the links do not have flat surfaces. Therefore, strain gages, might not be the most practical sensors to use when dealing with vibrations. Low-cost MEMS accelerometers are used as the practical alternative due to their cost and ease of incorporation into the overall controller architecture.

The performance of the estimator is proven both in simulation and experiments.

1.3.3 Control of Flexible Manipulators

Controllers can be divided into the categories of open-loop controllers and closed-loop controllers. Open-loop controllers work by generating the command signals so that the unwanted dynamics of the system are canceled out. Closed-loop controllers, on the other hand, work by measuring or estimating the current states and forcing them to follow desired trajectories.

If the dynamics of the flexible manipulator are known with high accuracy, then an open-loop controller could satisfy performance criteria. The biggest advantage of open-loop controllers is the lack of required sensors. Open-loop controllers, however, cannot guarantee performance if the dynamics of the system are not known well, or problematic disturbances exist.

The big disadvantage of closed-loop control is that the use of (often noisy) sensors is necessary. In addition, some state variables may need observers, which adds in extra computational difficulties. In the case of flexible manipulators, the measurements of the joint positions are readily available from encoder readings. The deflection variables, however, are typically not directly measurable. Since models for flexible serial manipulators are nonlinear, the controllers will need to be nonlinear as well. In addition, the stability of many closed loop controllers depend on knowing system dynamics relatively well.

Open and closed loop controllers can be used concurrently and independently from one-another. This design allows for the unwanted dynamics caused by following the reference trajectory to be minimized, while still being able to handle disturbances and some unknown dynamics. An additional benefit is the actuator effort lowering due to the closed loop controller not needing to suppress all of the unwanted dynamics.

This dissertation will explore both open-loop and closed-loop control independently. The performance of the open-loop controller is evaluated both in simulations and experiments, while the closed loop controller is examined in simulations only.

1.4 Dissertation Contributions

This dissertation presents a new way to efficiently model both static serial structures through an extension to TMM, as well as the complex dynamics of flexible serial manipulator by using the true modes of a flexible manipulator as the basis. The dynamical model is computationally efficient and therefore suitable for real-time controls applications. Both of the modeling techniques are verified by comparisons to commercial FEA software and experimental data.

An estimation algorithm based on accelerometer data is developed and analyzed. An extended Kalman filter is used to estimate the flexible system states.

An open loop controller based on Specified Insensitivity shaping techniques is presented. This controller uses the predicted frequency variation along the desired trajectory to create a optimized sequence of impulses that greatly reduces the amount of residual and transient vibration.

A closed loop nonlinear controller is presented to further reduce vibrations in the flexible arms. The controller uses the estimation data from the Kalman filter. Proofs on the system stability are given.

The contributions from this dissertation can potentially be applied to other fields where structural vibration is a problem and there are limited methods to sense the flexible states directly. Examples of these fields include building structures [133], machining [5] and piezoelectric energy harvesting [27].

1.5 Dissertation Roadmap

Chapter 2 shows a summary of the work done by researchers in the realm of flexible serial robot arms. An overview of previous modeling, estimation, and control techniques is presented. A brief discussion on the strengths and weaknesses of different methods is given.

Chapter 3 consists of two main parts: 1) modeling static n-link serial structures and 2) modeling the dynamics of n-link serial arms. An extension is made to the Transfer Matrix Modeling technique so that generic n-link serial structures can be handled algorithmically. In the dynamics part, the dynamic model for serial arms is derived using the mode obtained

from the static part.

Chapter 4 describes the design process used to create the testbed to validate the work presented in this dissertation. Chapter 5 presents estimator algorithms that are suitable for a serial flexible manipulators. An empirical evaluation is carried out for the estimators.

Chapter 6 presents the work done in both open-loop and closed-loop control. Analytical proofs are given for the stability analysis. The controllers are evaluated for different performance criteria. Finally, the work of this dissertation is concluded with Chapter 7, where the work is summarized, and potential future work is described.

Chapter II

BACKGROUND AND LITERATURE REVIEW

The majority of previous work on flexible robots has been performed on single link arms. The work on multi-link arms has often been theoretical and supported by simulations or planar experiments. Evidence for these statements can be found in [47] where a literature review of more than 400 papers in the filed of flexible robots is presented. The following sections will give a brief overview of what has been done in the areas of modeling, estimation, and control for multi-link flexible serial manipulators.

2.1 Dynamic Modeling of Flexible Robot Arms

Controller design and estimation require the availability of an accurate system model that can characterize the entire robot. Modeling classical serial robot arms can successfully omit flexible states because the stiffnesses of the links is high. However, modeling flexible robots, especially when dealing with multiple links in 3D space, is not straight forward.

2.1.1 Modeling Traditional Rigid Serial Arms

For rigid manipulators, the dynamics of the manipulator is fully described by:

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

where q is the vector of joint variables, M is the joint variable dependent inertia matrix, C is the vector containing all Coriolis and centrifugal terms, G contains the influence of gravity, and τ is the vector of input torques. Among numerous examples in literature, these equations were derived in [114, 126]. The controlled variables q and its derivatives are generally directly readable from encoder or tachometer measurements. Although (1) can be coupled, the torques τ act on q directly, and therefore, controller design is straight forward for these type of manipulators.

In reality, however, there is no structure that is absolutely rigid and, therefore, (1) is an approximation that holds under limited conditions. Research has shown that there exists

a critical ratio between the poles of the closed loop control for the joints and the first natural frequency of the structure $\frac{\omega_{cl}}{\omega_{str}}$, under which the assumption for rigid manipulators is valid [23, 24]. It was shown that the controller should have bandwidth less than 1/3 of the first natural mode of the structure, otherwise the there can be significant vibration in the end-effector motion due to structural vibrations.

2.1.2 Flexible Joint Serial Robot Manipulators

In addition to flexibilities from the structure, the joints of the serial arm could have flexibilities of their own [36]. Joint flexibilities originate from flexible elements in the drive-train, such as belt or even gears. When dealing with flexible joints, an additional variable, θ is introduced to the manipulator dynamics (1). θ represents the displacement of the actuator, while q represents the displacement of the link. The dynamics of flexible joint manipulators can, therefore, be described as:

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

where K is the joint stiffness matrix and B is the matrix of rotor inertias of the actuators. The state observability problem can be solved by using input and output side encoders. Numerous researchers have tackled this problem and have presented different controllers to decrease the effect of joint flexibility. Among the large body of work in [37] a PD regulation control algorithm was designed. In [4] a passivity based controller framework was presented to handle the problem. In [34] an adaptive controller to cope with time-varying uncertainties in flexible joints.

2.1.3 Flexible Link Serial Robot Manipulators

Generally, dynamic models of flexible link manipulators can be described by partial differential equations (PDEs) or approximate finite-dimensional ordinary differential equations (ODEs). Thus the dynamics of flexible beams is represented by an infinite dimensional variable space. In practice a truncated finite series of modes is used to represent the manipulator dynamics. However, un-modeled higher modes can cause problems and drive the

system unstable. This effect is known as modal spillover [10].

The flexible states are generally not directly measurable and observers are required to estimate the state variables [63]. In addition, in [21] it was recognized, that unlike single-link manipulators, multi-link manipulators cannot be accurately modeled using linear models. The difficulties lie with nonlinearities of the rigid-body motions and the mode shapes of the linearized model as the robot configuration varies.

In some earlier studies, the effect of elastic link deflections on the rigid-body motion was assumed negligible. The importance of coupling between the elastic and rigid-body motion was discussed in [150], where a cantilever beam attached to a rotating rigid hub was studied. It was shown that there is a significant difference in dominant resonance frequencies obtained from simulations using coupled and uncoupled equations.

2.1.3.1 Field Descriptions of Vibrating Beams

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 [29] because of their large length to diameter ratios. The dynamics of a flexible Euler-Bernulli beam is described by:

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

where E is the Young's modulus, $I(x)$ is the second moment of area, $w(x,t)$ is the transverse displacement of a point, $f(x,t)$ is the force acting on a point, and $\mu(x)$ is the distributed mass of the beam. A general solution to the unforced system differential equation (3) is of the form [56, 96]:

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

where $\beta^4 = \frac{\omega^2 m}{EI}$. The natural modes are determined from the constants C that are are calculated by applying the boundary conditions to (4). The solutions like (4) to the PDEs are often analytically not available due to non-uniform mass and stiffnesses, damping, and other effects. Exact solutions, that are the solutions to (3), to planar serial static structures

have been found in [42] where work was verified with finite element analysis (FEA) with structures of 2, 3 and 4 links. In addition in [97] exact solutions were found for a planar 2 link robot arm with varying elbow angle at static location. In [137] a procedure was presented to use exact solutions for robot arms using Lagrangian formulations.

2.1.3.2 Finite Element Modeling

Finite Element modeling (FEM) discretizes the structure into smaller elements where every element is considered as a part of the continuous member of the link. The method requires that the forces and displacements be compatible at certain positions along the structure. The equations of motion for the whole system are derived by assembling the individual element's equation of motion through polynomial interpolation functions. This technique has been used in many flexible robotics projects [45, 135, 142, 148].

In [3] FEM analysis was used to study the effecitveness of viscoelestic damping augmention for active control of a large flexible space manipulator. In [100] elemental and system equations were derived for systems with both elastic and rigid links. In [80] a single-link flexible manipulator in a 3D work space using FEM was studied. The major benefit of FEM is that it allows for irregularities (such as non constant cross section in the beams) in the structure. The major disadvantage of FEM is the computational complexity. In [22] a method for solving the inverse dynamics was formulated for a model derived with FEM. In [71] a method for modeling was developed using nonlinear finite elements, that treats vibration as a first order perturbation to the rigid motion.

2.1.3.3 Assumed Modes Method (AMM)

The assumed modes approach is an energy-based method that uses basis functions, that satisfy at least the geometric boundary conditions, to 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. The accuracy of AMM increases by increasing the number of trial functions as the eigenvectors of the system become asymptotically close to the exact modes of the system. This also means that, the more modes considered, the more computationally burdensome this approach becomes. The method uses the Lagrange's

equations [56, 96] or the Newton-Euler approach to derive the systems equation of motion.

In [14] 4×4 transfer matrices from AMM were developed for manipulators with rotary joints. In general, as the number of links increases, so does the computational complexity and the number of state variables needed. In [81] 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 was proposed. A simple way to derive state-space models from the assumed modes method was shown in [62].

Dynamic models for planar two-link manipulators using a Lagrangian-based finite dimension model assumed mode method were developed in [35, 39]. A Newton-Euler and AMM based technique for deriving dynamical models for manipulators with flexible links and joints was developed in [129]. Comparisons between the assumed modes and finite element models for flexible multi-link manipulators indicated that FEM is less demanding in terms of computation [135]. In [9] the virtual work principle was used to allow n serial links to be connected together using the assumed modes approach.

Global Modal Parametrization allows static modal configurations to be used for systems where the configurations change [20]. This work is especially relevant since it provides a framework on how to incorporate flexible states with rigid body motion in a mathematically rigorous way. This method has been proven to lessen computational load, while maintaining accuracy for flexible multi-body dynamics [99].

The issue of changing mode-shapes during large motions with flexible robots has not been thoroughly investigated, and when AMM is used, the modes are considered to be constant [47]. The effect of changing mode shapes based on robot configurations was studied in [33, 77, 97, 140]. However, a systematic way to obtain a modes for varying configurations of the manipulators has not yet been presented.

2.1.3.4 Transfer Matrix Method

Transfer Matrix Method (TMM) [18,36,78] is an extension to the solutions of the PDEs governing the flexible elements. The basis of this method is that each element of the system can be represented as a transfer matrix so that states (displacements, deflections, moments

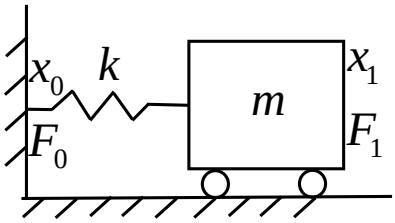


Figure 3: Mass-Spring System

and forces) can be related from one point on the system to the next, as a function of frequency. Natural frequencies of the system can be found by calculating the frequencies at which the transfer matrix for the system has a non-trivial null-space when all of the boundary conditions are taken into consideration [13, 16]. The system mode shapes can then be found by calculating the states along the system for the corresponding natural frequency. The simplest example showing how the transfer the transfer matrix metod works is the mass-spring system shown in Figure 3. The states for the system are x_i and F_i. Using Newton's second law yields the transfer matrix between the states at 0 and 1:

$$\begin{Bmatrix} x_1 \\ F_1 \end{Bmatrix} = \begin{bmatrix} 1 & 1/k \\ ms^2 & ms^2/k + 1 \end{bmatrix} \begin{Bmatrix} x_0 \\ F_0 \end{Bmatrix}, \tag{5}$$

where s represents the the derivative in the Laplace domain. The boundary conditions for the system are $x_0 = 0$ and $F_1 = 0$. Substituting in the boundary conditions into (5) and writing out the bottom row gives:

$$(ms^2/k + 1)F_0 = 0 \tag{6}$$

Therefore, the non-trivial null-space is:

$$ms^2 = -k \Longrightarrow s = \pm i\sqrt{k/m}, \tag{7}$$

where the imaginary part of s is the natural frequency of the system. For such a simple system the benefits of TMM are not apparent. What makes TMM attractive for flexible robots,

is its inherent ability to take into consideration distributed elements without discretization and incorporating lumped elements.

For flexible robot arms the base can be considered one boundary and the end-effector the second, while the links and actuators form the system transfer matrix. In [78] the 4×4 transfer matrix method was extended to three dimensional flexure in a systematic manner, and [17] developed state space models using the transfer matrix method via an iterative procedure. Bending-Torsion coupling [54] 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 that can be transformed into state space form. Recently, this method has seen use in piezoelectric energy harvesting [27]. This method has not been demonstrated to work on a general spatial n-link serial arm.

When using TMM with state space representations physical intuition is lost due to the somewhat abstract system states. This method could, however, be used as an efficient tool to generate the basis functions for a very low order model for flexible robots. Most researchers use approximated modes from polynomial series to represent the mode shapes of the system that do not give a good fit for the entire system. This problem is amplified greatly when spatial mode shapes are in question. The finite element method is another possible solution to find accurate modes for any system.

2.2 State Estimation for Flexible Serial Manipulators

For traditional rigid manipulators, encoders and tachometers are typically used to measure all of the states of the dynamical model. Flexure, on the other hand, is generally not directly measurable as a state of the system. For feedback control to be used to damp out oscillations, flexible state measurements are needed. Therefore, additional sensors should be used to capture flexure. State observers are then used to provide the estimates of the flexible states from the inputs and outputs of the system.

2.2.1 Sensor Selection

There are many types of sensors that could potentially provide data about oscillations; however, not all are practical. Sensors need to placed in correct locations along the links to assure that measurements capture the states of the system. Examples of sensors used in vibrations for feedback include: vision systems [51, 70, 103, 147], optical deflection sensors [50,59,102,103], strain sensors [2,52,63,83,139,147], Fiber-Bragg-Grating sensors [53], PZT sensors [59], and accelerometers [105, 108, 127]. In addition to the type of sensor, their location must be carefully selected because some locations, such as nodal points of mode shapes, do not provide any information about the magnitudes of those modes. This issue was thoroughly investigated in [109].

2.2.2 State Estimation Algorithms

A dynamic system can be generally represented by:

$$\begin{aligned} \dot{x} &= f(x,u,t) \\ y &= h(x,u,t) \end{aligned} \tag{8}$$

where x is the vector of unknown sates, u is the input to the system.An observer can estimate the states based on measurements y. An observer can be represented by the following system:

$$\begin{aligned} \dot{\hat{x}} &= f(\hat{x},u,t) - L(y-\hat{y}) \\ \hat{y} &= h(\hat{x},u,t) \\ \hat{x}(0) &= x_0, \end{aligned} \tag{9}$$

where $\hat{x}$ is the estimate of the states. There are numerous algorithms for selecting the observer gain L that minimizes the difference between x and $\hat{x}$.

For linear systems, the Kalman filter [84] is an optimal estimator for a stochastic system whose model and sensor noises are zero-mean, Gaussian, and whose noise covariances are known. In practice, these values are nearly impossible to find exactly, however, the Kalman filter has been proven to be powerful tool for numerous applications [146, 154].

While estimation techniques for linear systems are quite straight-forward, for non-linear systems the problem is much more difficult and fewer tools exist in the literature. By far the

most popular technique for estimating nonlinear systems is an adaptation to the Kalman filter, called the Extended Kalman Filter (EKF) [84]. The main idea is that the system can be linearized locally around the last state estimate. The EKF is not an optimal filter for a stochastic non-linear system, but it is a minimal error variance estimator. In [82] an Extended Kalman filter was to improve the tip position of a planar 2 link flexible robot arm using a infrared-light detector. In [64] and EKF was used to improve the positioning on the tool on a traditional serial robot using acceleration feedback.

Another nonlinear observer that has been proven to work for noisy non-linear systems is the H_∞ [72] type filter. While some of the requirements that exist for EKF, such as zero mean noise, have been relaxed for the H_∞ filter, its implementation is trickier because the magnitude of some of the parameters are abstract and must be chosen ad-hoc, non-linear extensions non-trivial, and the computation effort is more extensive.

Sliding mode estimators have been developed for single link and 2 link planar arms using the joint position measurement to estimate flexure [25, 91]. Poor convergence and chatter have been reported when there exists parametric error [109]. In [28] it was claimed that for non-linear systems EKF produced more accurate results than Sliding Mode observer, although the former is harder to implement. Neural network based observers were presented in [1].

Monte-Carlo based methods like the particle-filters [8] have been proven to produce more accurate estimates than EKF, and are more numerically stable. However, these type of filters are not implementable in real-time for any moderately sized state-space.

2.3 Control of Flexible Serial Manipulators

As with rigid robots, the goal of a controller for a flexible serial arm manipulator is generally to force the the end effector to follow a desired trajectory. To formulate the control and estimation problem, two physical limitations associated with flexible serial robot arms must be taken into consideration: 1) torque can only be applied at the joints, and 2) finite number of sensors with limited bandwidth can be used. In addition, robot arms that have significant flexure are under-actuated and, therefore, the typical control laws for rigid robots

such as the computed torque control law:

$$\tau = M(q)(\ddot{q}_d - K_p e - K_d \dot{e}) + C(q, \dot{q})\dot{q} + g(q) \tag{10}$$

cannot be used.

In addition, controller strategies developed for a rigid robot directly on a robot with flexible links could lead to instabilities due to unstable zero dynamics [38] because the system is non-minimum phase. Furthermore, research has shown that depending on the configuration of the robot arm, actuators that can only apply torques at the joints may not be able to have any effect on some flexible states and, therefore, the controllability of the flexible states is configuration dependent [33, 138, 153]. In those previously presented methods, the modes shapes for the spatial configurations were not rigorously analyzed.

To reduce vibrations from flexible manipulators, two approaches can be used: open-loop, and closed-loop control. Open-loop control works by modifying the control input to the plant, and closed-loop control works by applying control input based on the states of the system. Open-loop and close-loop control can be used together in one controller.

2.3.1 Open Loop Control of Flexible Robot Arms

When the plant of a control system is well known, the commands can be shaped to give a slower, but much less oscillating, response. A common problem for open-loop control methods is that the reference command times will lengthen. Additionally, large changes in the plant dynamics can limit the effectiveness of the control method. A popular open-loop control method is input shaping. It is a command-filtering technique where the nominal command is convolved with a series of impulses, known as the input shaper [120]. The main idea of input- shaping is shown in Figure 4, where a pulse velocity command is convolved with a series of impulse to produce a staircase command that significantly reduces the residual vibration of the plant.

A significant advantage of input shaping is the simplicity of controller design. In [122] it was shown that input shapers are shorter in length than traditional low-pass filter such as Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters. This was further

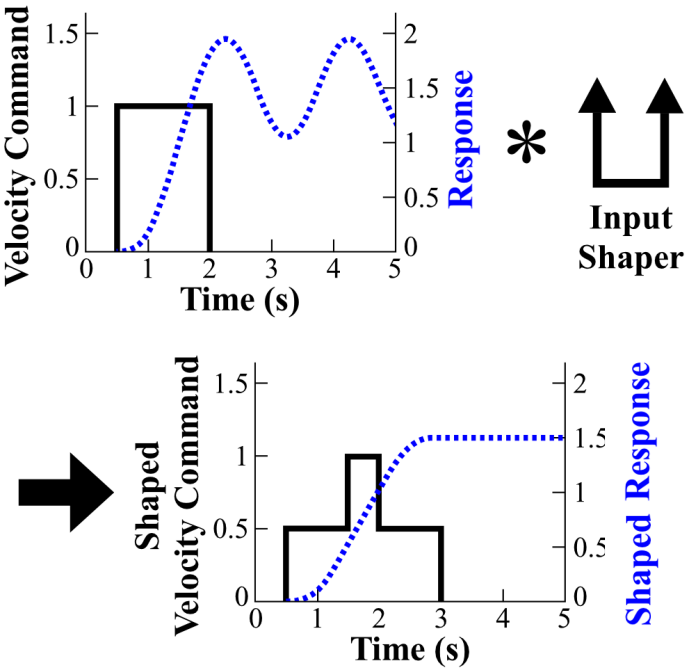


Figure 4: Input Shaping Process

expanded in [123], where it was shown that input-shapers had the shortest duration of all digital filters while having good vibration suppression characteristics.

Because flexible robots can have numerous significant modes multi modal shapers should be used [69].The main drawback with this approach is that the shaper duration will be the sum of all of the individual shapers. A more optimal approach could be the use of Specified Insensitivity (SI) shapers that can be optimized for the shaper duration [124]. In addition SI shapers can be optimized for ranges of frequencies.

Comparison of robust input shapers showed that Extra Insensitivity (EI) and SI shapers efficiently provide robustness to modeling errors and parameter uncertainty [144]. Input shaping has been successfully applied to flexible robot arms in [11, 88, 98, 111]. Reduction of multiple modes for a flexible robot arm was first shown in [88]. In [111, 132] adaptive input shapers were presented for flexible manipulators. Input-shaping techniques have been combined with position control of joints [106]. None of the works to date have explicitly taken into account the changes in the natural frequencies in the configuration space for flexible manipulators.

2.3.2 Closed Loop Control of Flexible Robot Arms

When there are significant model uncertainties and/or disturbances involved, feedback control is the only way to obtain a high-accuracy response. There are numerous feedback control methods described in literature and the choice is ultimately a compromise between the closed-loop behavior, the level of accuracy that the model has, and the ease of implementation on real manipulators. Linear state feedback control is, perhaps, the most well developed control strategy. However, for flexible arms it requires modeling simplifications of the continuum and nonlinear structure of the flexible system. All techniques developed to date rely on estimates of the flexible system states or end effector position measurements to provide the appropriate control effort.

2.3.2.1 Classical Controls

The classical control method in this dissertation encompass forms of feedback linearized control approaches. The main idea of this approach is that the nonlinear parts in the system dynamics can be negated with appropriate control effort. State feedback terms are then added to make the error dynamics stable. In [40] a PD type controller for the joint motion of a robot arm under gravity was presented. Although the traditional joint PD controller can stabilize flexible robots, the system performance is not very satisfactory because there is no explicit effort introduced to suppress the residual vibrations. In [86] a PID controller with feed forward terms for tracking the joint motion and damping vibrations was derived by using the second method of Lyapunov for a multi-link flexible manipulator. In [76] an extra term dependent on the flexible states to damp vibration was added to a PD controller for the joint motion.

In [6] inversion based tracking control was presented where a desired trajectory for flexible coordinates was computed online. Damping in the system was increased by modifying the equations for calculating the desired trajectories. Tests were conducted on a two-link planar arm. In [38] a stable inversion technique was presented for tip tracking. Three methods were provided that would always keep the joint error and torques bounded.

In [136] a two stage controller was designed, where an inversion based sliding mode

controller would move the joints of the robot along the desired joint trajectory and then an impedance based controller would reduce the vibration in the arm at the end point of the trajectory.

2.3.2.2 Two-time scale control

Singular perturbation theory [75] has been proven to be an effective method to reduce the sizes of models need to be controlled. This is known as "Reduced-order modeling." In order for singular perturbation theory to be applied on a system, the model must be able to be written in the following form:

$$\begin{aligned} \dot{x} &= f(x, z, \epsilon) \\ \epsilon \dot{z} &= g(x, z, \epsilon), \end{aligned} \tag{11}$$

where x and z are state variables, $f()$ and $g()$ are functions, and ϵ is a small parameter. This method allows the formation of a "fast" system and a "slow" system. The controller for the system in (11) can be designed in two stages: one for the fast system and one slow one. This method is applicable to the study of flexible manipulator because generally, the dynamics governing the joint motion is much slower than the fast dynamics. The goal for this approach is to simplify the the controller design and improve hardware implementation of the control algorithms. For this theory to work a sufficiently, a small parameter ϵ must be found so that the operating frequencies of the two controllers are separated and do not interfere with one another. Generally, it has been advised that the lowest structural frequency be chosen as the separation constant ϵ.

Singular perturbation based controllers have been demonstrated in [95,117,118]. In [118] the slow controller is the feedback linearized torque control and a pole placement strategy was chosen for the fast subsystem.

In [32] the rigid system was separated from the flexible system through the passivity based approach and a PID type controller was used to cancel out the oscillations. In [79,85] a two-time scale controller where the fast system is stabilized with a fuzzy logic based controller was used was presented. The time scale was based on the closed loop dynamics of the slow subsystem. Experiments were done on a single link beam.

2.3.2.3 Modern Control Methods

A number of feedback compensation approaches are available for the control of flexible manipulators. Robust control methods like H_∞ [116, 145] and sliding mode control [12, 73, 151] have been applied with moderate success. Adaptive algorithms have also been applied to improve the controller performance given model uncertainties or time varying system parameters. However, adaptive control with no compensation for the flexible behavior yields little improvement [51].

In [68] a LQR based controller was used on a flexible robot arm. More recently, fuzzy logic [19] and neural network based control schemes [113] have been applied to the problem. A Hybrid Neural-Fuzzy control was used for a robot with an uncertain model [26]. In [58] a two-link planar arm was controlled with a repetitive learning self adapting fuzzy controller to track a trajectory in simulation. An LQR controller together with the sliding mode approach was used in [48]. An adaptive controller for a flexible robot with an updating internal model was presented in [43]. In [130] hybrid fuzzy neural control for two link planar flexible manipulators was derived. In [79] a fuzzy logic based controller was used with singular perturbation based approach. A real time adaptive controller using reinforcement learning for tracking tip trajectory was shown in [110], experiments were conducted on a planar two link arm with strain feedback.

2.3.3 Other Methods to Reduce Vibrations in Flexible Manipulators

To avoid the problem of only having control input available at the joints some researchers have added other actuators to the arm or tried to change the arms themselves to be less prone to vibrations. In [44, 55, 131] piezoelectric materials were added to single link manipulators to measure and reduce vibrations in the arms. In [101] a spring and dashpot were attached to reduce the vibrations of a single link arm. In [112] a study was undertaken to find the optimal structural design of lightweight manipulators. In [3] a finite element analysis was performed to study viscoelastic passive damping augmentation on a space shuttle remote manipulator system.

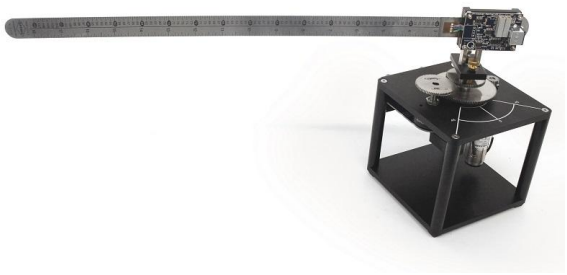


Figure 5: Quanser Flexible Link Test Bed

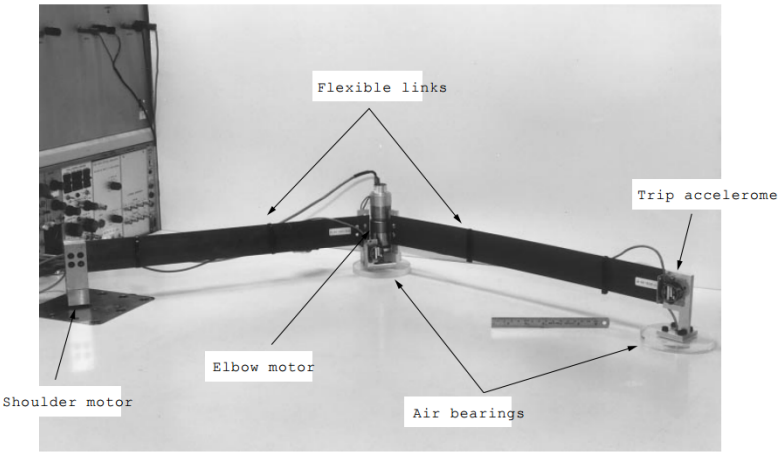


Figure 6: Typical 2 Link Planar Test Bed [97]

2.4 Test Beds

As evident from [47] the vast majority of test beds for flexible manipulators use a single link and one actuated degree of freedom. A commercial flexible link test bed by Quanser can be seen in Figure 5[1].

Typical two link planar flexible manipulator test beds are mounted on air bearings to minimize any torsional and gravity effects and provide a very low friction surface. An example of such a test bed is shown in Figure 6.

The rarest test beds have multiple links and operate in spatial configurations. TUDOR [90] at Technische UniversitÃďt Dortmund is a 2 link robot that has been used for that

[1]Image from: `http://www.quanser.com/products/rotary_flexible_link`

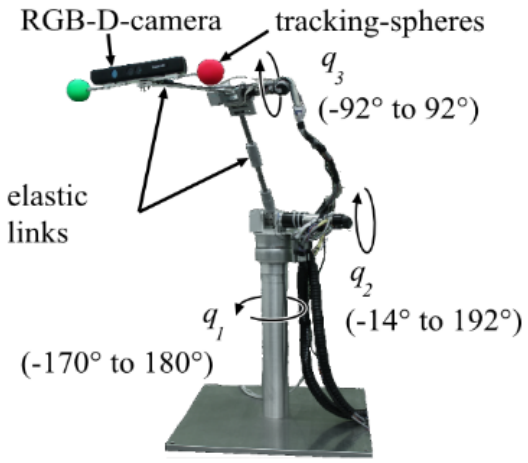


Figure 7: TUDOR

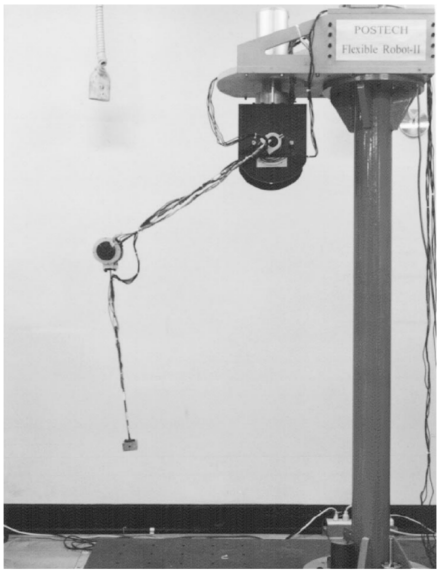


Figure 8: POSTECH Flexible Robot II [31]

has the capability to move in spatial configurations, however vibration control has only been demonstrated for the planar case. A picture of TUDOR is shown in Figure 7[2]. The POSTECH Flexible Robot II has been used for a vast number of papers from the researchers at the Pohang University of Science and Technology. The robot can be seen in Figure 8. The robot arm is one of the only test beds to have used feedback control for spatial oscillations.

[2]Image from `http://www.rst.e-technik.tu-dortmund.de/cms/en/research/robotics/TUDOR_engl/index.html`

Instead of using torque control, this manipulator arm uses high gain velocity servos.

Other two-link spatial robots include the robot from Univesidad de Castilla-La Mancha [49]. This robot's kinematics were designed after the industrial robot PUMA 560. The parallel linkage mechanism makes modeling this arm extremely cumbersome using traditional methods. The ADAM test bed is 2 link flexible manipulator from Tohoku Univeristy that can be used in spatial configurations [141]. What makes this test bed special is that it consist of two identical arms. If only one arm is used (which is the case for most of the research reported from that group) it looks like the POSTECH arm described earlier.

Chapter III

MODELING FLEXIBLE SERIAL MANIPULATORS

This chapter presents a method to derive low-order models for general n-link robot arm that adequately characterizes the behavior of the flexible serial robot arm. Low-order models are preferred because they typically are computationally less taxing and can, therefore, be used for real-time control and estimation applications. The most common approach in the literature for modeling multi-link robot arms is to assign mode shapes and modal amplitudes to each link separately while using clamped-mass boundary conditions. A small subset of papers that have used this approach include [7, 30, 39]. The approach, however, is not fully consistent with real-world dynamics.

To obtain more consistent dynamic models, a hybrid approach of both frequency and time domain modeling methods can used to form the non-linear equations of motion that describe the rigid-flexible dynamics. To model flexible serial structures in static spatial configurations, an extension to the transfer matrix method is utilized. This approach is verified with commercial FEA software and experimental testing. The varying mode shapes from the TMM approach are then used in the derivation for the low-order model for flexible serial robot arms.

3.1 Denavit-Hartenberg Parameters

Denavit-Hartenberg (DH) Parameters [41] are commonly used to describe the configuration of serial robots. These parameters can also be used to describe any serial structure. The DH parameters are used as a basis for the systematic algorithm used for the TMM analysis presented here

Each joint i, along with the base 0 and end-effector $n+1$, is assigned a frame O with the location p. Figure 9 shows how the frames and DH coordinates are related. θ_i is the joint angle measured from x_{i-1} to x_i about z_i. α_i is the link twist angle measured from z_i to

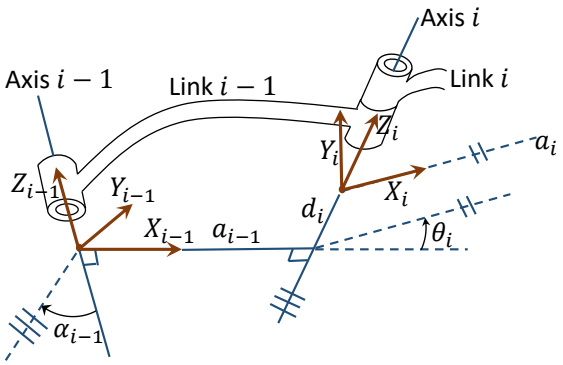


Figure 9: Link Frames

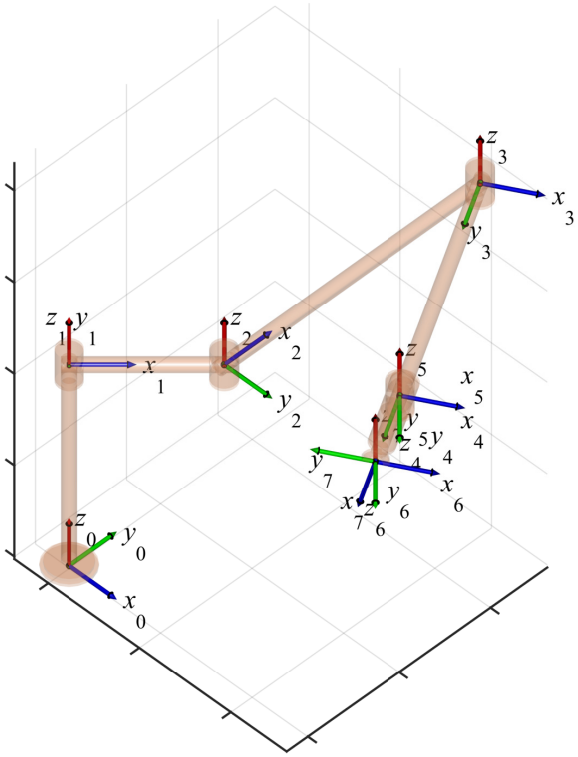


Figure 10: Coordinate Frames in DH Coordinates

z_{i+1} about x_i. d is the joint offset measured from O_{i-1} to O_i along z_i. a is the link common normal measured from O_i to $O_i + 1$ along x_i. An example of how the coordinate frames are located for a Puma-class robot is shown in Figure 10. Starting from the lower left side the 0 frame is placed at the base. After that each joint gets a frame while finally frame 7 is place at the end-effector. The z_i axis are in the direction of the joints and x_i is chosen to be orthogonal to z_i and z_{i+1}. Generally multiple options exist for placing the coordinate

frames.

3.2 The Transfer Matrix Method(TMM)

In order to incorporate the flexible states in to the dynamical model of the system the mode shapes of the flexible arm can be found. These mode shapes are used in the next section where the equations that govern the entire serial flexible arm are derived. However, the use of this method on a general spatial n-link serial arm has not been previously demonstrated.

The main idea of transfer matrix method based modeling is that a matrix $U(j\omega)$ can be derived to relate state vectors z in one location on the structure to another (for example from one joint to the next).

$$z_{i+1} = U_i(j\omega)z_i \tag{12}$$

These expressions can be combined to form a system relating one end of the structure to the other:

$$z_L = ...U_2U_1U_0z_0 = U_{tot}z_0 \tag{13}$$

In general, the solutions to the partial differential equations that govern multi-link flexible serial structures are almost always impossible to solve in closed form. However, transfer matrices of different elements can be combined together to compute the system in arbitrary poses. The roots of the resulting boundary value problem are the natural frequencies, or in general, the eigenvalues of the individual modes of vibration. The state vector that fully

describes the spatial deformable link at a point is:

$$z = \begin{Bmatrix} w_x \\ \theta_x \\ M_x \\ F_x \\ w_y \\ \theta_y \\ M_y \\ V_y \\ w_z \\ \theta_z \\ M_z \\ V_z \end{Bmatrix} = \begin{bmatrix} \text{Displacement in the x direction} \\ \text{Rotation about x axis} \\ \text{Moment about x axis} \\ \text{Shear force about x axis} \\ \text{Displacement in the y direction} \\ \text{Rotation about y axis} \\ \text{Moment about y axis} \\ \text{Shear force about y axis} \\ \text{Displacement in z direction} \\ \text{Torsion about z axis} \\ \text{Moment about z axis} \\ \text{Axial Force along the z axis} \end{bmatrix} \tag{14}$$

In the next subsections the transfer matrices for beams, rigid structures, and rotations are derived. In this work, only small deflections with negligible damping effects that can be approximated with linear behavior are considered.

3.2.1 Beam Transfer Matrix

The three dimensional transfer matrix will be of dimension 12x12 and can be represented as:

$$z_L = U_B z_0 = \begin{bmatrix} B_{xy} & 0 \\ 0 & AT_z \end{bmatrix} z_0 \tag{15}$$

where AT_z is a 4x4 matrix that includes the torsional and axial components and B_{xy} is a 8x8 matrix that has the bending components in the x and y directions.

3.2.1.1 Axial and Torsional Matrix

The well-known equation describing axial beam motion is:

$$EA\frac{\partial^2 w_z(z,t)}{\partial z^2} = \rho A\frac{\partial^2 w_z(z,t)}{\partial t^2}, \tag{16}$$

where E is the modulus of Elasticity, A is the cross-sectional area of the beam, ρ is the density of the beam, and $w_z(z,t)$ is the axial deformation. Separation of variables is used to separate the spatial and time-dependent components:

$$w_z(z,t) = Z(z)T(t) \tag{17}$$

Therefore, an ODE describing the spatial variable is:

$$Z'' - \sigma^2 Z = 0, \tag{18}$$

where,

$$\sigma^2 = -\omega^2 \frac{\rho}{E}, \tag{19}$$

where, ω is the system natural frequency.

The solution to this equation will be of the form:

$$Z = c_1 e^{\sigma z} + c_2 e^{\sigma z}, \tag{20}$$

where c_1 and c_2 are constants that are to be determined based on the boundary conditions. We also know that:

$$F_z = EAZ'. \tag{21}$$

Substituting (20) into (21) and then putting it in a matrix form yields:

$$\begin{bmatrix} w_z \\ F_z \end{bmatrix} = \begin{bmatrix} e^{\sigma z} & e^{-\sigma z} \\ EAe^{\sigma z} & -EAe^{\sigma z} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = U_{Az} c \tag{22}$$

After evaluating (22) at 0 and the beam length, L, the transfer matrix from 0 to L is found to be:

$$A = U_{Az}(L,\sigma)[U_{Az}(0,\sigma)]^{-1} = \begin{bmatrix} \frac{1}{2}\left(e^{\sigma L} + e^{-\sigma L}\right) & \frac{1}{2EA\sigma}\sigma\left(e^{\sigma L} - e^{-\sigma L}\right) \\ \frac{1}{2}EA\sigma\left(e^{\sigma L} - e^{-\sigma L}\right) & \frac{1}{2}\left(e^{\sigma L} + e^{-\sigma L}\right) \end{bmatrix} \tag{23}$$

The equation that governs a shaft in torsion is:

$$GJ\frac{\partial^2 w_z(z,t)}{\partial z^2} = \rho J \frac{\partial^2 w_z(z,t)}{\partial t^2}, \tag{24}$$

where G is the modulus of rigidity, J is the polar moment of area. The derivation of the transfer matrix is equivalent to the process used for the axial vibration, and therefore:

$$T = \begin{bmatrix} \frac{1}{2}\left(e^{\sigma L} + e^{-\sigma L}\right) & \frac{1}{2GJ\sigma}\sigma\left(e^{\sigma L} - e^{-\sigma L}\right) \\ \frac{1}{2}GJ\sigma\left(e^{\sigma L} - e^{-\sigma L}\right) & \frac{1}{2}\left(e^{\sigma L} + e^{-\sigma L}\right) \end{bmatrix} \tag{25}$$

where,

$$\sigma^2 = -\omega^2 \frac{\rho}{G} \tag{26}$$

Combining (23) and (25) produces:

$$\begin{Bmatrix} w_z \\ \theta_z \\ M_z \\ F_z \end{Bmatrix}_L = \begin{bmatrix} A_{11} & 0 & 0 & A_{12} \\ 0 & T_{11} & T_{12} & 0 \\ 0 & T_{21} & T_{22} & 0 \\ A_{21} & 0 & 0 & A_{22} \end{bmatrix} \begin{Bmatrix} w_z \\ \theta_z \\ M_z \\ F_z \end{Bmatrix}_0 = AT_x \begin{Bmatrix} w_z \\ \theta_z \\ M_z \\ F_z \end{Bmatrix}_0 \tag{27}$$

3.2.1.2 Bending Matrix

The well-known equation governing beam bending for an Euler-Bernulli beam is:

$$-EI_y \frac{\partial^4 w_x(z,t)}{\partial z^4} = \mu \frac{\partial^2 w_x(z,t)}{\partial t^2}, \tag{28}$$

where E is Young's modulus, I is the area moment of inertia, μ is the mass per length, and $w_x(z,t)$ is the bending deformation. We also know:

$$\theta_x = -\frac{\partial w_y}{\partial z} \tag{29}$$

$$M_x = -EI_x \frac{\partial^2 w_y}{\partial z^2} \tag{30}$$

$$V_y = \frac{\partial M_x}{\partial z} \tag{31}$$

The solution for (28) can again be found by using separation of variables:

$$w_x(z,t) = Z(z)T(t) \tag{32}$$

this leads to the ODE for the spatial part:

$$Z^{(4)} - \left(\frac{\beta}{L}\right)^4 Z = 0, \tag{33}$$

where,

$$\left(\frac{\beta}{L}\right)^4 = \frac{\omega^2 \mu}{EI_y} \tag{34}$$

The solution to (33) is of the form:

$$Z = c_1 \sin\frac{\beta z}{L} + c_2 \cos\frac{\beta z}{L} + c_3 \sinh\frac{\beta z}{L} + c_4 \cosh\frac{\beta z}{L}, \tag{35}$$

where c_i are constants that are dependent on the boundary conditions. By substituting (35) into (29)-(31) the following relation is found:

$$\begin{Bmatrix} w_x \\ \theta_y \\ M_y \\ V_x \end{Bmatrix} = U(z,\beta)c \tag{36}$$

By evaluating (36) at 0 and L, the following relationship is found:

$$\begin{Bmatrix} w_x \\ \theta_y \\ M_y \\ V_x \end{Bmatrix}_L = U(L,\beta)[U(0,\beta)]^{-1} \begin{Bmatrix} w_x \\ \theta_y \\ M_y \\ V_x \end{Bmatrix}_0 = B_a \begin{Bmatrix} w_x \\ \theta_y \\ M_y \\ V_x \end{Bmatrix}_0, \tag{37}$$

where,

$$B_a = \begin{bmatrix} c(1) & \frac{Lc(4)}{2\beta} & \frac{ac(3)}{2\beta^2} & -\frac{Lac(2)}{2\beta^3} \\ \frac{\beta c(2)}{2L} & c(1) & \frac{ac(4)}{2\beta L} & -\frac{ac(3)}{2\beta^2} \\ \frac{\beta^2 c(3)}{2a} & \frac{\beta L c(2)}{2a} & c(1) & -\frac{Lc(4)}{2\beta} \\ -\frac{\beta^3 c(4)}{2La} & -\frac{\beta^2 c(4)}{2a} & -\frac{\beta c(2)}{2L} & c(1) \end{bmatrix}, \tag{38}$$

where,

$$a = \frac{L^2}{EI_y}, \qquad c = \begin{bmatrix} \frac{1}{2}\cos(\beta) + \frac{1}{2}\cosh(\beta) \\ -\sin(\beta) + \sinh(\beta) \\ -\cos(\beta) + \cosh(\beta) \\ \sin(\beta) + \sinh(\beta) \end{bmatrix} \tag{39}$$

Analogously, the bending equation in the orthogonal direction is:

$$-EI_x \frac{\partial^4 w_y(z,t)}{\partial z^4} = \mu \frac{\partial^2 w_y(z,t)}{\partial t^2} \tag{40}$$

where,

$$\theta_y = -\frac{\partial w_x}{\partial z} \tag{41}$$

$$M_y = -EI_y \frac{\partial^2 w_x}{\partial z^2} \tag{42}$$

$$V_x = \frac{\partial M_y}{\partial z} \tag{43}$$

Finally, the transfer matrix can be written:

$$\begin{Bmatrix} w_y \\ \theta_x \\ M_x \\ V_y \end{Bmatrix}_L = B_b \begin{Bmatrix} w_y \\ \theta_x \\ M_x \\ V_y \end{Bmatrix}_0, \tag{44}$$

where,

$$B_b = \begin{bmatrix} c(1) & -\frac{Lc(4)}{2\beta} & -\frac{ac(3)}{2\beta^2} & -\frac{Lac(2)}{2\beta^3} \\ -\frac{\beta c(2)}{2L} & c(1) & \frac{ac(4)}{2\beta L} & \frac{ac(3)}{2\beta^2} \\ -\frac{\beta^2 c(3)}{2a} & \frac{\beta L c(2)}{2a} & c(1) & \frac{Lc(4)}{2\beta} \\ -\frac{\beta^3 c(4)}{2La} & \frac{\beta^2 c(3)}{2a} & \frac{\beta c(2)}{2L} & c(1) \end{bmatrix}, \tag{45}$$

where,

$$\beta^4 = \frac{\omega^2 \mu L^4}{EI_x},$$

$$a = \frac{L^2}{EI_x}, \qquad c = \begin{bmatrix} \frac{1}{2}cos(\beta) + \frac{1}{2}\cosh(\beta) \\ -\sin(\beta) + \sinh(\beta) \\ -\cos(\beta) + \cosh(\beta) \\ \sin(\beta) + \sinh(\beta) \end{bmatrix} \tag{46}$$

Matrices B_a and B_b are combined to form B_{xy} according to the state vector convention (14)

in the following way:

$$\begin{Bmatrix} w_x \\ \theta_x \\ M_x \\ V_x \\ w_y \\ \theta_y \\ M_y \\ V_y \end{Bmatrix}_L = \begin{bmatrix} B_{a_{11}} & 0 & 0 & B_{a_{14}} & 0 & B_{a_{12}} & B_{a_{13}} & 0 \\ 0 & B_{b_{22}} & B_{b_{23}} & 0 & B_{b_{21}} & 0 & 0 & B_{b_{24}} \\ 0 & B_{b_{32}} & B_{b_{33}} & 0 & B_{b_{31}} & 0 & 0 & B_{b_{34}} \\ B_{a_{41}} & 0 & 0 & B_{a_{44}} & 0 & B_{a_{42}} & B_{a_{43}} & 0 \\ 0 & B_{b_{12}} & B_{b_{13}} & 0 & B_{b_{11}} & 0 & 0 & B_{b_{14}} \\ B_{a_{21}} & 0 & 0 & B_{a_{24}} & 0 & B_{a_{22}} & B_{a_{23}} & 0 \\ B_{a_{31}} & 0 & 0 & B_{a_{34}} & 0 & B_{a_{32}} & B_{a_{33}} & 0 \\ 0 & B_{b_{42}} & B_{b_{43}} & 0 & B_{b_{41}} & 0 & 0 & B_{b_{44}} \end{bmatrix} \begin{Bmatrix} w_x \\ \theta_x \\ M_x \\ V_x \\ w_y \\ \theta_y \\ M_y \\ V_y \end{Bmatrix}_0 \tag{47}$$

3.2.2 Rigid-Body Transfer Matrix

Like the Beam Transformation Matrix, the Rigid Body transfer matrix can be broken down into smaller sub-matrices:

$$z_L = U_R z_0 = \begin{bmatrix} RB_{xy} & 0 \\ 0 & RB_z \end{bmatrix} z_0 \tag{48}$$

The rigid-body transfer matrices are derived by summing moments and forces about the center of gravity and deriving the displacements and rotations by kinematic relations, assuming the body is fully rigid. The local rotation of the body does not change, therefore:

$$\theta_L = \theta_0 \tag{49}$$

When assuming small angles of rotation, the translational changes are:

$$w_{xL} = w_{x0} + \theta_y L \tag{50}$$

$$w_{yL} = w_{y0} + \theta_x L \tag{51}$$

$$w_{zL} = w_{z0} \tag{52}$$

In the following relations, $s = j\omega$, m is the mass of the rigid body L is the length of the rigid body along the z axis, r is the distance to the center of mass from the beginning of the link in the z direction, and I_x, I_y, I_z are the moments of inertia in the x, y, z axis,

respectively. Summing forces and moments about the center of gravity in the y direction and moments in the x direction gives the following transfer matrix:

$$\begin{Bmatrix} w_x \\ \theta_y \\ M_y \\ V_x \end{Bmatrix}_L = \begin{bmatrix} 1 & -L & 0 & 0 \\ 0 & 1 & 0 & 0 \\ ms^2(L-r) & s^2 I_x - ms^2 r(L-r) & 1 & L \\ ms^2 & -ms^2 r & 0 & 1 \end{bmatrix} \begin{Bmatrix} w_x \\ \theta_y \\ M_y \\ V_x \end{Bmatrix}_0 = RB_a \begin{Bmatrix} w_x \\ \theta_y \\ M_y \\ V_x \end{Bmatrix}_0 \tag{53}$$

Analogously, summing the forces in the x direction and moments in the y direction, the following transfer matrix is found:

$$RB_b = \begin{bmatrix} 1 & L & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -ms^2(L-r) & s^2 I_y - ms^2 r(L-r) & 1 & -L \\ ms^2 & ms^2 r & 0 & 1 \end{bmatrix} \tag{54}$$

Finally, summing the forces and moments about the z axis gives:

$$RB_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & s^2 I_z & 1 & 0 \\ ms^2 & -ms^2 r & 0 & 1 \end{bmatrix} \tag{55}$$

The full Rigid-Body transfer matrix is found by deriving RB_{xy} from R_a and R_b with the same algorithm as used to obtain (47).

3.2.3 Rotation Matrices

To be able to use the TMM for serial arms in arbitrary positions, rotation matrices are needed for the joints. The 12×12 rotation matrices are derived directly from the standard 3×3 rotation matrices by multiplying every element with the 4×4 identity matrix $\mathbb{I}_{4\times4}$. Thus, for rotation about the z axis, the transfer matrix is:

$$R_z(\alpha) = \begin{bmatrix} \cos(\alpha)\mathbb{I}_{4\times4} & \sin(\alpha)\mathbb{I}_{4\times4} & 0 \\ -\sin(\alpha)\mathbb{I}_{4\times4} & \cos(\alpha)\mathbb{I}_{4\times4} & 0 \\ 0 & 0 & \mathbb{I}_{4\times4} \end{bmatrix} \tag{56}$$

For rotation about the y axis the, transfer matrix is:

$$R_y(\alpha) = \begin{bmatrix} \cos(\alpha)\mathbb{I}_{4\times4} & 0 & -\sin(\alpha)\mathbb{I}_{4\times4} \\ 0 & \mathbb{I}_{4\times4} & 0 \\ \sin(\alpha)\mathbb{I}_{4\times4} & 0 & \cos(\alpha)\mathbb{I}_{4\times4} \end{bmatrix} \tag{57}$$

For rotation about the x axis the, transfer matrix is:

$$R_x(\alpha) = \begin{bmatrix} \mathbb{I}_{4\times4} & 0 & 0 \\ 0 & \cos(\alpha)\mathbb{I}_{4\times4} & \sin(\alpha)\mathbb{I}_{4\times4} \\ 0 & -\sin(\alpha)\mathbb{I}_{4\times4} & \cos(\alpha)\mathbb{I}_{4\times4} \end{bmatrix} \tag{58}$$

3.2.4 TMM Analysis

The TMM can be used to find the natural frequencies and mode shapes for complex serial structures. The method requires the multiplication of the various matrices corresponding to flexible bodies, rigid bodies, and rotations:

$$z_{tip} = U_{sys}(U_B, U_R, U_S, R_i)z_0 \tag{59}$$

Partitioning U_{sys} appropriately gives:

$$\begin{Bmatrix} 0 \\ \vdots \\ 0 \end{Bmatrix} = (subU)\hat{z}_0 \tag{60}$$

The natural frequencies of the system correspond to the values of s that cause $subU$ to have a null-space. After finding the values of s, the kernel vector $\hat{z}_0$ can be found. By combining $\hat{z}_0$ with the known boundary conditions in z_0, we can form z_0. Once z_0 has been found we can use the transfer matrices along the link to find the intermediate states along the entire serial arm.

The mode shapes of the system can be calculated by iterating the coordinate x along the beams:

$$z(x) = U_B(x)\hat{z}_0, \tag{61}$$

where $z(x)$ is the state vector at position x and z_0 is the state vector at the beginning of the beam. Note that for a serial arm, each beam will have a separate z_0 that can be found by finding the $\tilde{U}_{sys}$ that corresponds to the system matrix up to the start of the beam:

$$\hat{z}_0 = \tilde{U}_{sys}(U_B, U_R, U_s, R_i)z_0 \tag{62}$$

Conventionally, in continuous vibration problems, the mode shapes consist of one component. For example, for a planar bending problem, the mode shape consists of the displacement in the direction perpendicular to the beam axis. The complete mode shape for a serial arm in 3D space consists of 4 components: w_x, w_y, w_z, θ_z. (Recall that θ_x is a function of w_y, θ_y is a function of w_x, and we can extract those components from the state vector z calculated along the arm.)

By using the DH parameters from Section 3.1, an algorithm can be used to form the transfer matrix of the system. This algorithm is required because the traditional DH parameters use different coordinate systems than those required by the TMM. If $dh.x$ represents the array of parameters starting with the index 0, then the algorithm is:

function GETSYSTEMMATRIX

$U_{sys} = \mathbb{I}_{12x12}$

for link=1 to NumberOfLinks **do**

if dh.a(link)==0 **then**

$U_{sys} = R_x(dh.\alpha(link))U_{sys}$

else

$U_{sys} = R_y(\frac{\pi}{2})U_{sys}$

end if

$U_{sys} = R_z(link \cdot \frac{\pi}{2})U_{sys}$

$U_{sys} = GetBeamTM(link) \cdot U_{sys}$

$U_{sys} = GetRigidBodyTM(link) \cdot U_{sys}$

$U_{sys} = R_z(link \cdot (-\frac{\pi}{2}))U_{sys}$

if dh.a(link)!=0 **then**

$U_{sys} = R_z(dh.f(link))U_{sys}$

$U_{sys} = R_y(-\frac{\pi}{2})U_{sys}$

end if

$U_{sys} = R_z(dh.\theta(link+1))U_{sys}$

end for

return U_{sys}

end function

This algorithm requires that flexible beams are located either i) between joints or ii) between the base and the joint or iii) a joint and the tip or the arm. The rigid attachments are An illustration of how the coordinate frames for TMM are oriented is shown in Figure 11. The frames for the robot are given to each link in the system. Note that some links can have zero length and are only included to make the DH convention work. z_i axis is always pointed towards the direction of the link. These coordinate frames are different than those used in the DH notation in Figure 10, where the z_i axis was always in the direction of the joint.

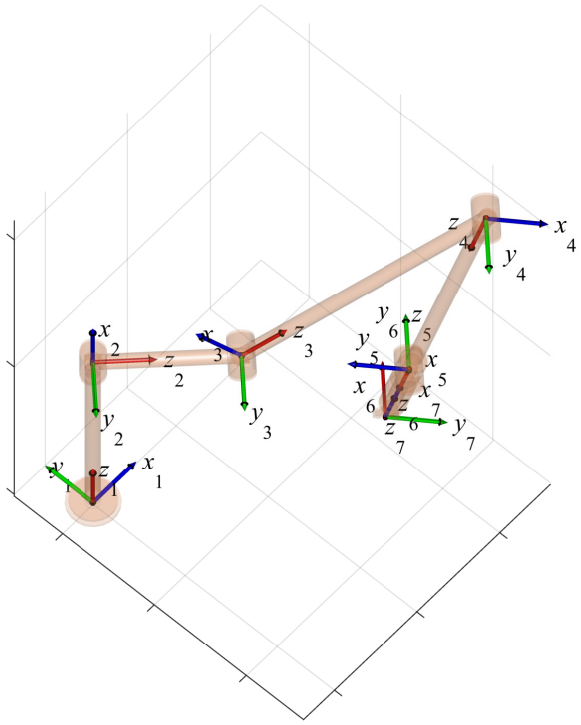


Figure 11: Coordinate Frames for TMM analysis

Table 1: Comparison with Castri et al [42]

	2 Link			3 Link			4 Link		
	Castri	TMM	%dif.	Castri	TMM	%dif.	Castri	TMM	%dif.
ω_1	6.687	6.687	0	3.218	3.217	0.03	2.034	2.033	0.05
ω_2	30.371	30.374	0.01	17.168	17.170	0.01	11.567	11.568	0.01
ω_3	124.25	124.27	0.02	33.815	33.818	0.01	21.060	21.062	0.01

3.2.5 Model Verification

In order to verify the spatial TMM technique, comparisons with previous work, FEA modeling, and hardware testing were performed. The previous studies were for planar n-link structures. Comparisons were made with previous work by Castri et al. [42] who verified his work with FEA analysis with planar structures of 2, 3 and 4 links. The results are compared in Table 1, where 2 link represents the 2 link structure presented in their paper at configuration $\theta_2 = 45°$; 3 Link represents the 3 link structure with $\theta_2 = -30°$, $\theta_3 = 60°$; 4 link represents the 4 link structure in configuration $\theta_2 = 45°$,$\theta_2 = -60°$, $\theta_3 = 90°$, w_i represents the i^{th} natural frequency. As can be seen from Table 1 the results match well . The small error could be attributed to the root solving algorithms or the numerical precision of terms, such as π used in the calculations.

Comparisons were also made with Milford et al. [97] who verified their work with experiments using a 2-link planar robot arm at various elbow joint displacements. The results obtained in [97] are shown in Figure 12. The results obtained with the TMM technique are shown in 13. Since the y-axis is in the logarithmic scale it is clear that the two methods produce equivalent results. Researchers in [42] and [97] used methods that were similar in principle (solving for the constants in the partial differential equations in a matrix fashion). However, unlike the method proposed in this chapter, their methods were limited to the planar structures.

In a PhD dissertation from Malzahn [89] impact hammer tests were done in one configuration with different tip masses as shown in Figure 14. The modeling in that dissertation was done using the assumed modes method using clamped-mass boundary conditions. It

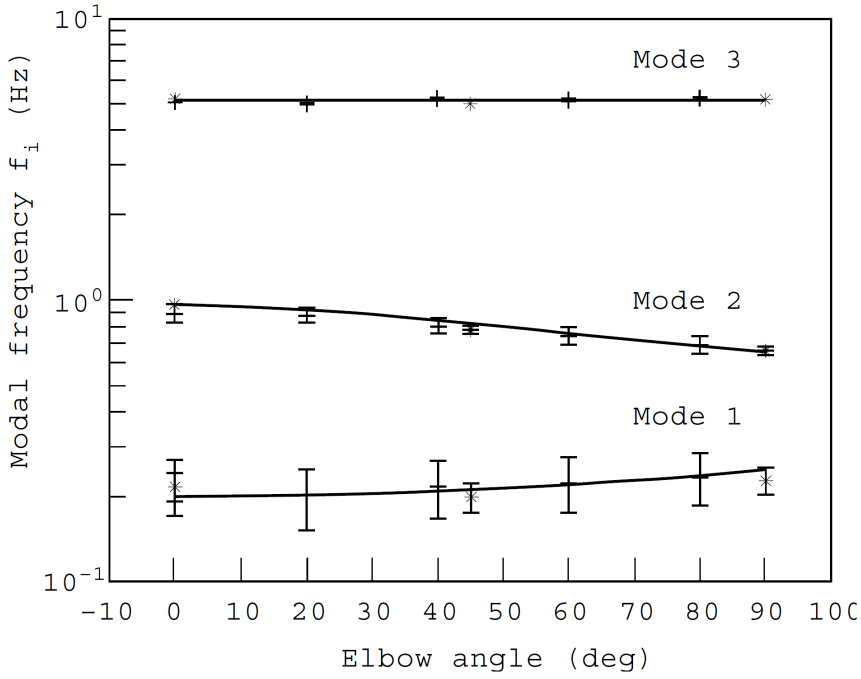


Figure 12: Elbow Angle Dependent Natural Frequencies From [97]

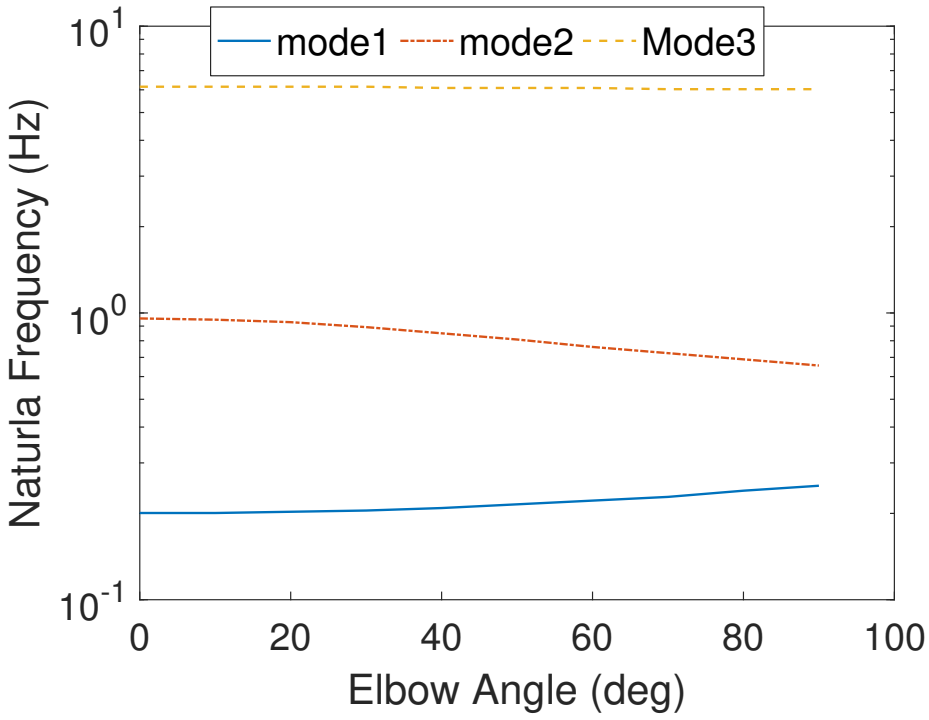


Figure 13: Elbow Angle Dependent Natural Frequencies From TMM

can be seen that TMM produces results that are much closer to the actual values. The discrepancy in the second mode could have come from improper clamping. Knowing the

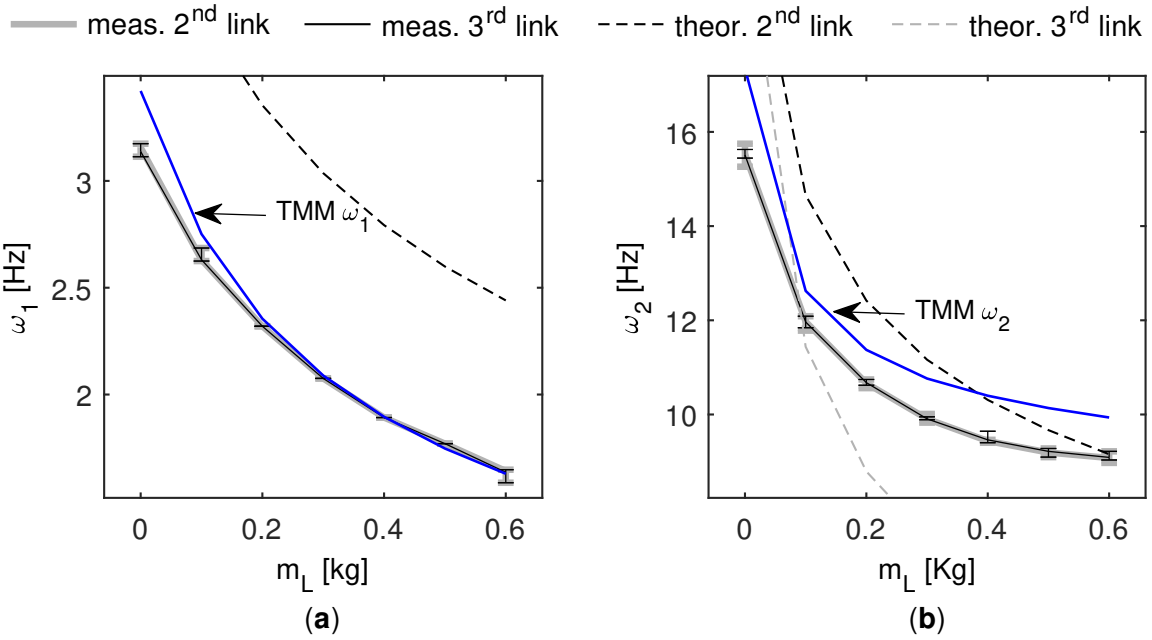


Figure 14: First (a) and second (b) dominant frequencies determined for varying payloads from strain measurements on the second (solid, gray) and third link (solid, black). The averages, minima and maxima are obtained from ten hammer excitation experiments per payload. Additionally the theoretical values for the first two natural frequencies of the second link (dashed, black) and the first natural frequeny of the third link (dashed, gray) are shown. [89] The natural frequencies obtained with TMM are shown with the blue line.

effective stiffness of the joint controller would have potentially yielded better results.

3.2.6 Experimental Verification of TMM

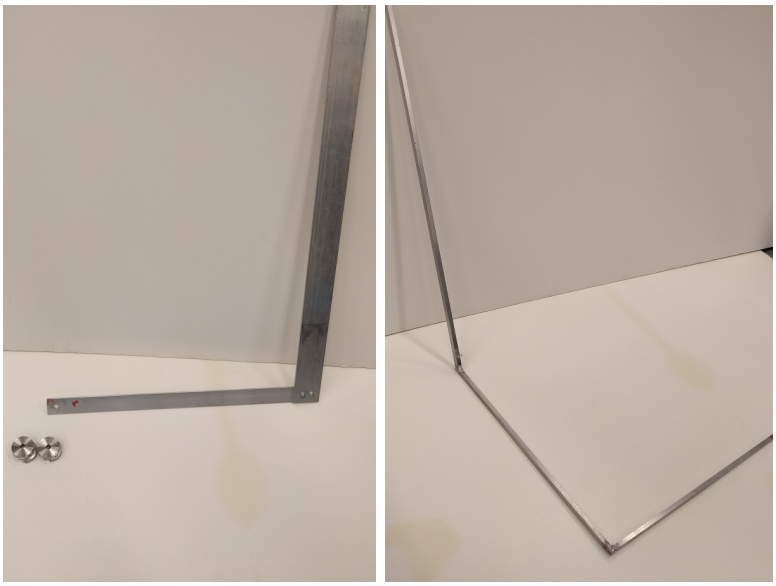
Validation in the previous section was in planar configurations, therefore, experiments in spatial configurations are in order. Figure 15 shows three different test fixtures that were constructed to show TMM working for different beam sizes and configurations. The fixture parameters are listed in Table 2. a_i is the link thickness in the y_i direction and b_i is the thickness in the x_i direction. For fixtures 2 and 3 the "second" link is of length 0 in order to do the proper rotations. Two additional configurations were tested: Fixture 1 with an added tipmass (all parameters the same as Fixture 1, except $m_t = [0.01, 0.1323]$) and Fixture 2 with two concentrated masses at the end of beams 1 and 2 (all parameters are the same as Fixture 2, except $m_t = [0.0125, 0, 0155, 0]$).

Table 2: Fixture Parameters

	Fixture 1			Fixture 2					Fixture 3				
i	0	1	2	0	1	2	3	4	0	1	2	3	4
$dh.\theta$ (rad)	-	$-\frac{\pi}{2}$	0	-	0	$\frac{\pi}{6}$	- 0	0	-	0	2.32	0	0
$dh.d$ (m)	-	0.432	0	-	0.442	0	0	0.3429	-	0.508	0	0	0.406
$dh.a$ (m)	0	0.381	-	0	0	0.3446	0	0	0	0	0.4775	0	-
$dh.\alpha$ (rad)	0	0	-	0	0	$\frac{\pi}{2}$	0	0	0	$\frac{\pi}{2}$	0.8727	0	-
a (m)	-	0.032	0.024	-	0.0095	-	0.0063	0.0063	-	0.0318	-	0.019	0.0024
b (m)	-	0.0381	0.0254	-	0.0063	-	0.0063	0.0063	-	0.0032	-	0.0024	0.019
ρ kg/m^3	-	2700	2700	-	2700	-	2700	2700	-	2700	-	2700	2700
G (GPa)	-	26.3	26.3	-	26.3	-	26.3	26.3	-	26.3	-	26.3	26.3
E (GPa)	-	69.0	69.0	-	69.0	-	69.0	69.0	-	69.0	-	69.0	69.0
m_t (kg)	-	0.01	0	-	0.001	-	0.004	0	-	0.01	-	0.005	0
I_{x_t} (kgm^2)	-	0	0	-	0	-	0	0	-	0	-	0	0
I_{y_t} (kgm^2)	-	0	0	-	0	-	0	0	-	0	-	0	0
I_{z_t} (kgm^2)	-	0	0	-	0	-	0	0	-	0	-	0	0

Note that the tipmasses also include the material added by the welds. These fixtures were placed in a shaker. Laser Doppler velocimeters (LDV) were directed at reflective tape on the fixtures. The shaker excited the fixtures with white-noise displacement at the clamp. Frequency response functions (FRF) from the accelerometer at the base to the displacements on the beam were obtained by averaging test data from 10 trials using Hanning windowing. National Instruments Signal Express software was used to run the experiments and calculate the FRFs.

Figure 16 shows the experimentally obtained FRFs. The peaks in FRF correspond to the natural frequencies of the modes. The natural frequencies can be found from the FRFs with 0.01 Hz accuracy. The y axes are in log scale. The FEA program used in this study to compare with TMM was COMSOL. The structures were modeled as they appear in real world and not as beam elements, to investigate weather welds and tip mass geometry has a



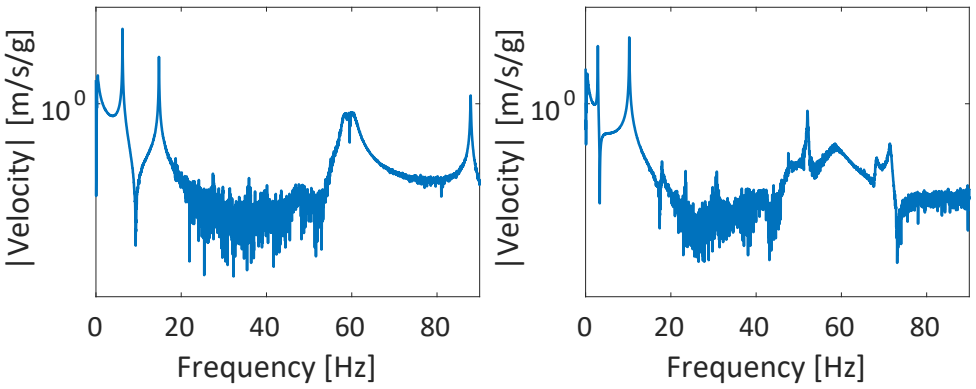
(a) Fixture 1 With Tip Masses

(b) Fixture 2

(c) Fixture 3

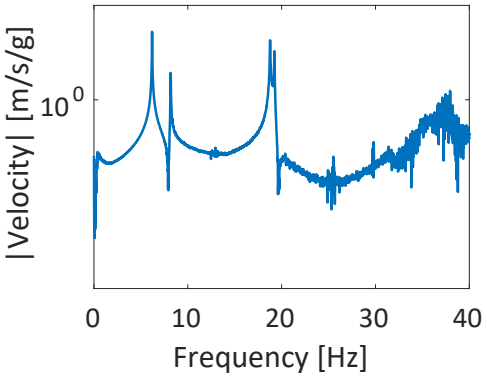
Figure 15: Fixtures for TMM Verification

significant effect on the results. Table 3 shows the natural frequencies obtained from Fixture 1 testing with experimental results from Figure 16a along with FEA and TMM results. All results match up fairly well, except ω_4 did not show up on the FRF because that mode was perpendicular to the LDV and the fixture was too thin to measure it from the side. Table 4 shows the results when the tipmass was attached at the end of fixture 1. The results still match up very well and there was no completely orthogonal mode to the LDV measurement

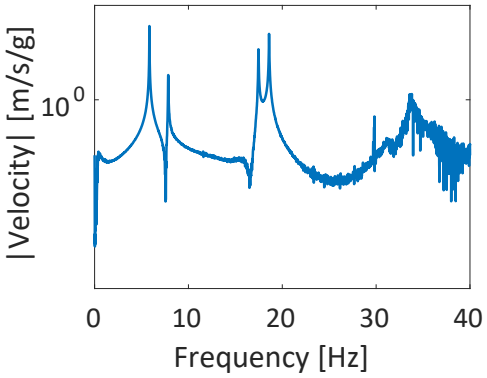


(a) Fixture 1

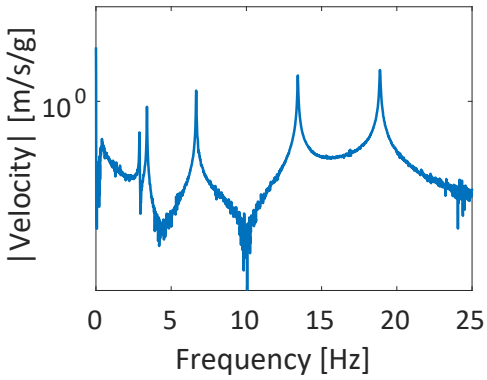
(b) Fixture 1 with Tipmass



(c) Fixture 2



(d) Fixture 2 with mass loads



(e) Fixture 3

Figure 16: Experimental FRFs

direction. As expected the natural frequencies went down with the addition of the mass. Table 5 shows the results from Fixture 2, with all predictions matching up very closely to the experimental data. Table 6 show the results when 2 masses we attached to Fixture 2. TMM and FEA predicted the results very accurately. As expected the natural frequencies went down very slightly, since the added masses were not big relative to the structure itself. Table

Table 3: Fixture 1 Natural Frequencies (Hz)

	Experiment	FEA	%error	TMM	%error
ω_1	6.22	6.84	9.97	6.71	7.87
ω_2	14.78	15.84	7.17	14.44	2.30
ω_3	59.92	63.84	6.54	64.27	7.26
ω_4		77.27		76.88	
ω_5	87.94	91.28	3.85	85.76	2.43

Table 4: Fixture 1 with Tipmass Natural Frequencies (Hz)

	Experiment	FEA	%error	TMM	%error
ω_1	2.88	3.16	9.72	2.72	5.56
ω_2	10.24	11.37	11.04	11.03	8.01
ω_3	30.80	35.03	13.73	31.09	0.94
ω_4	52.01	55.72	7.15	51.63	0.71
ω_5	71.42	77.89	9.05	77.64	8.71

Table 5: Fixture 2 Natural Frequencies (Hz)

	Experiment	FEA	%error	TMM	%error
ω_1	6.19	6.15	0.65	5.96	3.71
ω_2	8.18	8.20	0.24	8.03	1.83
ω_3	18.74	19.39	3.47	18.16	3.09
ω_4	19.25	19.76	2.65	19.28	0.16
ω_5	37.75	41.78	10.68	37.90	0.40

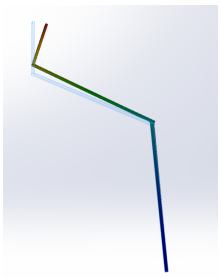
Table 6: Fixture 2 With 2 Mass Loads Natural Frequencies (Hz)

	Experiment	FEA	%error	TMM	%error
ω_1	5.84	6.09	4.28	5.50	5.82
ω_2	7.86	8.126	3.43	7.643	2.80
ω_3	17.47	17.18	1.65	16.78	3.94
ω_4	18.62	18.63	0.11	19.09	2.58
ω_5	33.58	35.80	6.61	33.60	0.06

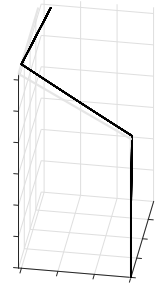
7 show the natural frequencies for Fixture 3, both FEA and TMM predicted the natural frequencies accurately. The FEA and TMM predicted the results with relatively the same accuracy.

Table 7: Fixture 3 Nature Frequencies (Hz)

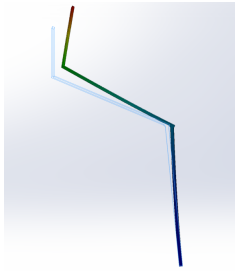
	Experiment	FEA	%error	TMM	%error
ω_1	2.89	2.67	7.61	2.66	7.96
ω_2	3.39	3.51	3.54	3.28	3.24
ω_3	6.63	6.67	0.60	6.58	0.75
ω_4	13.41	14.10	5.15	13.34	0.52
ω_5	18.87	18.34	2.81	18.35	2.75



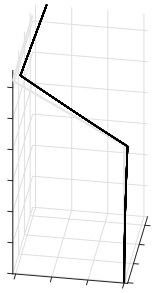
(a) Mode 1 FEA



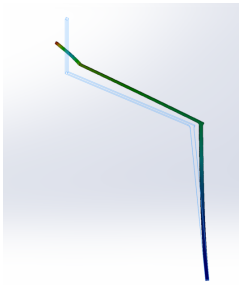
(b) Mode 1 TMM



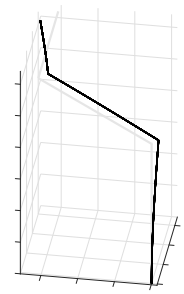
(c) Mode 2 FEA



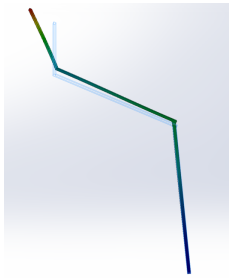
(d) Mode 2 TMM



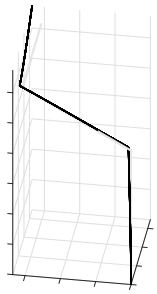
(e) Mode 3 FEA



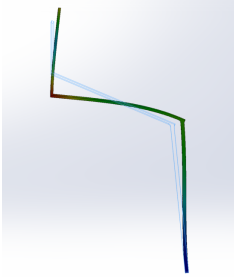
(f) Mode 3 TMM



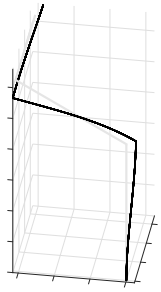
(g) Mode 4 Fea



(h) Mode 4 TMM



(i) Mode 5 FEA



(j) Mode 5 TMM

Figure 17: Fixture 2 Mode Shapes

As indicated by the consistent results comparing frequencies, the mode shapes obtained with the TMM also match very well with the ones obtained from FEA. Figure 17 shows the mode shapes of Fixture 2 from both FEA and the TMM. The minor differences are due to scaling.

The models used in FEA were designed to match the actual structure very closely. The models included material for the welds and fasteners. When doing the TMM analysis there is some ambiguity in determining where the beams actually start, and what length values are. One can potentially select those variables to match the experimental data even closer, but in these experiments the lengths of the beams were measured along the center of each beam. In addition, the tip-masses were not located precisely at the tip. The added mass for welds was approximated using CAD software.

3.3 Flexible Serial Arm Dynamics

In this section the dynamical model for a serial flexible robot arm will be derived using mode shapes that are defined for the whole structure. The mode shapes can be conveniently found with the method described in the previous section, however, other methods such as FEA could be used as well. The derivation follows similar procedures as found in [7, 152]. However, the major difference is that the mode shapes are defined for the entire structure, and the dynamics of the tip attachments, inertia due to motor shafts, and torsional effects are taken into account. The following approximations are used for this derivation: 1) The vibration in the links can be considered linear (no shear effects); 2) the joint position must be controlled.

3.3.1 Flexible Arm Kinematics

Before the equations of motion can be derived the relationships between the joint angles and the mass-elements of the serial arm must be found. A point on link i can be related to the coordinate system fixed to link $i-1$ by a transfer matrix A_i. Any point on the serial arm described in the local coordinate system i can be related to the inertial frame 0 with the transformation:

$$ {}^0r_i = r_i = T_i{}^i r_i, \tag{63} $$

$$ T_i = \prod_{j=1}^{i} A_j \tag{64} $$

$$ T_0 = \mathbb{I}, $$

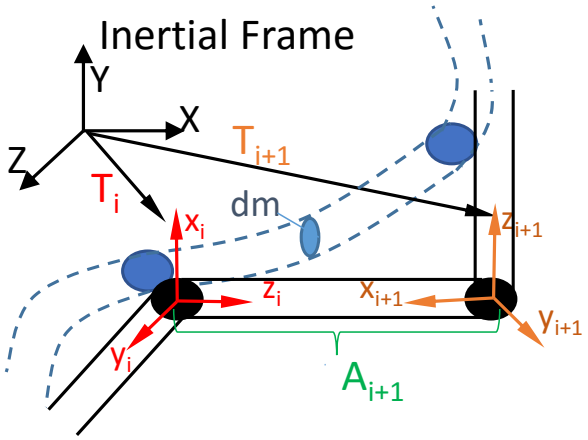


Figure 18: Frame Transformations

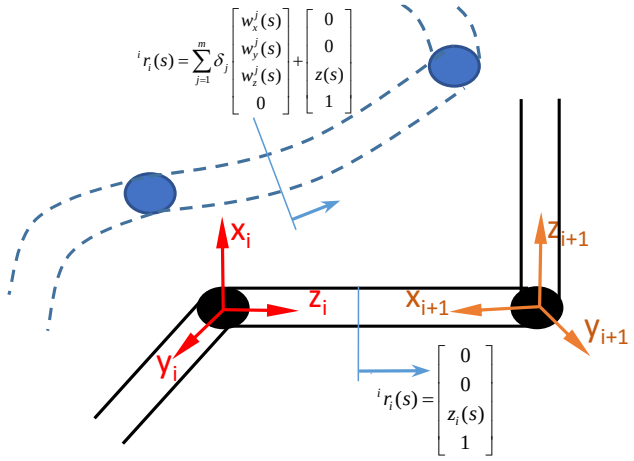


Figure 19: Coordinates on Link i

where ${}^{i}r_i$ is the position vector of the rigid arm in the local i coordinate frame. When the leading superscript is 0 (${}^{0}()$) it is omitted in the derivations for simplicity. A_i is the modified homogeneous matrix that ensures the neutral axis always matches the z_i direction of the neutral axis of a the link i. An algorithm on how to calculate A_i will be presented in a later Section. Figure 18 shows the relationships between the coordinate frames. Matrix T_i is always in the form of $\begin{bmatrix} R_{i_{3\times3}} & p_{i_{3\times1}} \\ 0_{1\times3} & 1 \end{bmatrix}$, where R_i is the rotation of frame i with respect to the inertial frame and p_i is the location of the origin of the i^{th} frame with respect to the inertial origin.

Point s on link i with respect to the frame i can be represented by a finite sum of separable mode shapes which are products of a time varying amplitudes and kinematically admissible mode shapes:

$$ {}^{i}r_i(s) = \sum_{j=1}^{m} \delta_j \begin{bmatrix} w^i_{j_x}(s_z) \\ w^i_{j_y}(s_z) \\ w^i_{j_z}(s_z) \\ 0 \end{bmatrix} + \begin{bmatrix} s_x \\ s_y \\ s_z \\ 1 \end{bmatrix}, \tag{65} $$

where δ_j is the mode shape amplitude for the j^{th} mode; $w^i_{j_y}$, $w^i_{j_z}$ are the deflections of mode j on link i in the x_i, y_i and z_i directions; and m is the number of mode to be considered. These parameters are illustrated in Figure 19

3.3.2 Algorithms for finding A_i and T_i

An algorithm for finding the transformation matrices A_i and T_i, that were used throughout this chapter, is presented in this section. The "for loop" cycles through the number of links and finds the appropriate transfer matrices based on the DH parameters. The reason the standard homogeneous transfer matrix generally used in serial robot arms is not used, is so that the flexible co-ordinates can be consistent, the local link frames must always have the z axis pointed in the direction of the normal axis in the link. This was illustrated in Figures 10 and 11. Note that this algorithm is very similar to the one used for the TMM analysis.

function FIND A_i T_i

$A_i = \mathbb{I}_{3\times3}$

$T_i = \mathbb{I}_{3\times3}$

$R_{tot} = \mathbb{I}_{3\times3}$

for link=1 to NumberOfLinks **do**

if dh.a(link)==0 **then**

$R_{tot} = R_{tot} \cdot R_x(dh.\alpha(link))^T$

else

$R_{tot} = R_{tot} \cdot R_y(\frac{\pi}{2})^T$

end if

$R_{Tot} = R_{Tot} \cdot R_z(link \cdot \frac{\pi}{2})^T$

if (link>1) **then**

$$A_i = \begin{bmatrix} & 0 \\ R_{Tot} & 0 \\ & linkLength_{i-1} \\ 0_{1\times3} & 1 \end{bmatrix}$$

$T_i = T_{i-1} A_i$

else

$$A_i = \begin{bmatrix} R_{Tot} & 0_{3\times1} \\ 0_{1\times3} & 1 \end{bmatrix}$$

$T_i = A_i$

end if

$R_{Tot} = \mathbb{I}_{3\times3}$

$R_{Tot} = R_{Tot} \cdot R_z(link \cdot \left(-\frac{\pi}{2}\right))^T$

if dh.a(link)!=0 **then**

$R_{Tot} = R_{Tot} \cdot R_z(dh.\alpha(link))^T$

$R_{Tot} = R_{Tot} \cdot R_y(-\frac{\pi}{2})^T$

end if

$R_{Tot} = R_{Tot} \cdot R_z(dh.\theta(link+1))^T$

end for

return A_i T_i

end function

All of the rotation matrices are of standard dimension (3×3). Note that the A_1 and T_1 do not contain the joint variable q_1. This is because the robot might have links before the first actuator. For derivatives $\frac{\partial A_i}{\partial q_i}$ the joint angle vector is defined as $q = [0, q_1, q_2, ..., q_n]^T$. Since the derivative $\frac{\partial A_1}{\partial q_0} = 0$, A_1 and T_1 do not contain any joint variables q, but they are required for generality of the equations and to find the absolute positions of each point of the robot arm.

3.3.3 Kinetic energy

Next the expression for kinetic energy for the serial flexible arm will be derived. Kinetic energy of link i can be expressed as

$$KE_i = KE_{i_{link}} + KE_{i_{mot}} + KE_{i_{tor}}, \tag{66}$$

where $KE_{i_{link}}$ is the kinetic energy that is from motion of of the links ; $KE_{i_{mot}}$ is the kinetic energy that is associated with the shaft of the motor; $KE_{i_{tor}}$ is the kinetic energy associated with only the torsion of the link. The kinetic energy of a rotational joint is expressed as:

$$KE_{mot_i} = J_{i_{mot}} \dot{q}_i^2, \tag{67}$$

where $J_{i_{mot}}$ is the polar moment of inertia of the shaft of the motor after gearing, and $\dot{q}_i$ is the rotational speed of the output shaft after the gearbox. Analogous expressions can be derived for prismatic joints. The translational kinetic energy for a link i is:

$$KE_{i_{link}} = \frac{1}{2} \int_{B_i} Tr\left(\frac{dr_i}{dt} \frac{dr_i^T}{dt} \right) dm \tag{68}$$

where $Tr()$ represents the trace operator, dm is the differential mass element, and the integration is taken over the entire link B_i, that includes tip attachments. Often in derivation of kinetic energy, the order of the vectors is reversed and therefore the trace operator is not needed. However, it is essential for this derivation, as will become evident in the following steps. Similarly the torsional kinetic energy for a link is:

$$KE_{i_{tor}} = \frac{1}{2} \int_{l_i} \left(\frac{d\Theta_z^{i\,T}}{dt} \frac{d\Theta_z^i}{dt} \right) dz, \tag{69}$$

where J_i is the polar moment of inertia of link i, $\Theta_z^i = \sum_{j=1}^{m} \delta_j \theta_z^i$ is the torsion of link i. Note that this energy term only takes into account the local torsional energy, not the deflection caused by torsion of other parts of the arm. The other kinetic energy due to torsion is already included in (68) because the mode shapes are defined over the whole structure.

The vector velocity of r_i can be calculated in the following form:

$$\frac{dr_i}{dt} = \dot{T}_i^i r_i + T_i^i \dot{r}_i \tag{70}$$

Therefore (68) becomes:

$$\frac{1}{2}\int_{B_i} Tr(\dot{T}_i^i r_i^i r_i^{iT} \dot{T}_i^T + 2\dot{T}_i^i r_i^i \dot{r}_i^{iT} T_i^T + T_i^i \dot{r}_i^i \dot{r}_i^{iT} T_i^T) dm \tag{71}$$

Hence the total kinetic energy is the summation of kinetic energies over all the links:

$$KE = \sum_{i=1}^{n} KE_i = \sum_{i=1}^{n} \left(Tr(\dot{T}_i B_{3i} \dot{T}_i^T + 2\dot{T}_i B_{2i} T_i^T + T_i B_{1i} T_i^T) + To_i + J_{i_{mot}} \dot{q}_i^{\,2} \right) \tag{72}$$

where,

$$To_i = \frac{1}{2}\sum_{j=1}^{m}\sum_{k=1}^{m} \dot{\delta}_j \dot{\delta}_k \left(\int_{l_i} \rho_i J_i (\theta_{j_z}^i)^T \theta_{k_z}^i dz + J_{t_i} \theta_{j_z}^i(l_i) \theta_{k_z}^i(l_i) \right) = \frac{1}{2}\sum_{j=1}^{m}\sum_{k=1}^{m} \dot{\delta}_j \dot{\delta}_k P_{ijk} \tag{73}$$

$$B_{1i} = \sum_{j=1}^{m}\sum_{k=1}^{m} \dot{\delta}_j \dot{\delta}_k (C_{ikj} + C_{ijk}^*) \tag{74}$$

$$C_{ijk} = \frac{1}{2}\int_{B_{l_i}} [w_{j_x}^i, w_{j_y}^i, w_{j_z}^i, 0]^T [w_{k_x}^i, w_{k_y}^i, w_{k_y}^i, 0] dm \tag{75}$$

$$\begin{aligned} C_{ijk}^* = \frac{1}{2} m_i [w_{j_x}^i(l_i), w_{j_y}^i(l_i), w_{j_z}^i(l_i), 0]^T [w_{k_x}^i(l_i), w_{k_y}^i(l_i), w_{k_y}^i(l_i), 0] \\ + \frac{1}{2} I_{t_{x_i}} [w_{j_x}^{i\,\prime}(l_i), 0, 0, 0]^T [w_{k_x}^{i\,\prime}(l_i), 0, 0, 0] \\ + \frac{1}{2} I_{t_{y_i}} [0, w_{j_y}^{i\,\prime}(l_i), 0, 0]^T [0, w_{k_y}^{i\,\prime}(l_i), 0, 0] \end{aligned} \tag{76}$$

$$\begin{aligned} B_{2i} = \sum_{j=1}^{m} \dot{\delta}_j (C_{ij} + C_{ij}^*) + \sum_{j=1}^{m}\sum_{k=1}^{m} \dot{\delta}_j \delta_k (C_{ijk} + C_{ijk}^*) = \\ = \sum_{j=1}^{m} \dot{\delta}_j \left(C_{ij} + C_{ij}^* + \sum_{k=1}^{m} \delta_k \left(C_{ijk} + C_{ijk}^* \right) \right) = \sum_{j=1}^{m} \dot{\delta}_j D_{ij} \end{aligned} \tag{77}$$

$$C_{ij} = \frac{1}{2}\int_{B_{l_i}} [x_i, y_i, z_i, 1]^T [w_{j_x}^i, w_{j_y}^i, w_{j_z}^i, 0] dm \tag{78}$$

$$C_{ij}^* = \frac{1}{2} m_i [0, 0, l_i, 1]^T [w_{j_x}^i(l_i), w_{j_y}^i(l_i), w_{j_z}^i(l_i), 0] \tag{79}$$

$$\begin{aligned} B_{3i} = (C_i + C_i^* + \sum_{j=1}^{m} \delta_j ([C_{ij} + C_{ij}^T] + [C_{ij}^* + C^{*T}{}_{ij}]) \\ + \sum_{k=1}^{m}\sum_{j=1}^{m} \delta_k \delta_j (C_{ijk} + C_{ijk}^*) = F_i \end{aligned} \tag{80}$$

$$C_i = \frac{1}{2}\int_{B_{l_i}} [x_i, y_i, z_i, 1]^T [x_i, y_i, z_i, 1] dm \tag{81}$$

$$C_i^* = \frac{1}{2}\begin{bmatrix} I_{t_{x_i}} & 0 & 0 & 0 \\ 0 & I_{t_{y_i}} & 0 & 0 \\ 0 & 0 & m_i l_i^2 & m_i l_i \\ 0 & 0 & m_i l_i & m_i \end{bmatrix} \tag{82}$$

In the formulas above x_i, y_i and z_i represent all the points on the links in the local x y and z coordinates; l_i is the length of link i; B_i represents the entire link, while B_{l_i} represents the link without the tip attachments; ρ_i is the density of link i; $I_{t_{x_i}}$ and $I_{t_{x_i}}$ are the mass moments of inertias for the tip attachments on link i in the x and y directions, respectively; J_{t_i} is the moment of inertia of the tip attachment in the z_i direction; m_i is the mass of the tip-attachments on link i. The C^* terms are analogous to C terms, except they are calculated at the end of the link. C_i^* is written out explicitly to show the inertias of the rigid attachment. The differential mass dm can be expressed as $dm = \rho_i dV = \rho_i dxdydz$. The calculations are simplified when the link is symmetric about the local z axis in the local x and y directions. In such cases, all products of integrals with x_i and y_i are 0; except with themselves, i.e $x_i \cdot x_i^T$.

It is well known that kinetic energy can be expressed by:

$$KE = \frac{1}{2}\dot{X}^T M \dot{X}, \tag{83}$$

where $\dot{X}_{ij} = [\dot{q}^T \dot{\delta}^T]^T$ is the velocity vector including all generalized velocities. In order to equate (72) to (83), $\dot{T}$ must be derived by starting with

$$T_i = \prod_{n=1}^{i} A_n(q_n(t)), \tag{84}$$

where $A_n \in \mathbb{R}^{4\times4}$ and $q_n \in \mathbb{R}^1$. Therefore,

$$\dot{T}_i = \sum_{h=1}^{i} \breve{T}_{h-1} U_h {}^h\tilde{T}_i \dot{q}_h, \tag{85}$$

where,

$$U_h = \frac{\partial A_h}{\partial q_h}, \tag{86}$$

$$\breve{T}_{h-1} = \prod_{j=1}^{h-1} A_j \tag{87}$$

$$ {}^{h}\tilde{T}_i = \prod_{j=h+1}^{i} A_j \tag{88} $$

The kinetic energy for the system can, therefore, be represented as:

$$ \begin{aligned} KE = \frac{1}{2}\sum_{i=1}^{n}\Bigg\{ Tr\Bigg(\sum_{h=1}^{i}\left(\breve{T}_{h-1}U_h\,{}^{h}\tilde{T}_i\right)\dot{q}_h F_i \left(\sum_{\alpha=1}^{i}\left(\breve{T}_{\alpha-1}U_\alpha\,{}^{\alpha}\tilde{T}_i\right)\right)^{T}\dot{q}_\alpha \Bigg) + \\ Tr\Bigg(2\sum_{h=1}^{i}\breve{T}_{h-1}U_h\,{}^{h}\tilde{T}_i\dot{q}_h\sum_{j=1}^{m}\dot{\delta}_j D_{ij}T_i^T\Bigg) + \\ Tr\Bigg(T_i\sum_{j=1}^{m}\sum_{k=1}^{m}\dot{\delta}_j\dot{\delta}_k(C_{ikj}+C^*_{ijk})T_i^T\Bigg) + \sum_{j=1}^{m}\sum_{k=1}^{m}\dot{\delta}_j\dot{\delta}_k P_{ijk} + J_{i_{mot}}q_i^2\Bigg\} \end{aligned} \tag{89} $$

The trace and sum operations are exchanged and terms are collected. The inertia coefficients in (89) can be divided into 3 groups: the joint angles $\dot{q}_\alpha\dot{q}_h$, joint angle and mode shape amplitude $\dot{q}_h\dot{\delta}_j$, and mode shape amplitudes $\dot{\delta}_j\dot{\delta}_k$.

Separating similar terms in (89) enables the joint angles $\dot{q}_\alpha\dot{q}_h$ extracted as:

$$ \frac{1}{2}\sum_{i=1}^{n}\sum_{h=1}^{i}\sum_{\alpha=1}^{i}\left\{ Tr[\breve{T}_{h-1}U_h^h\tilde{T}_i F_i\left(\breve{T}_{\alpha-1}U_\alpha\,{}^{\alpha}\tilde{T}_i\right)^T] + J_{i_{mot}}\eta\right\}\dot{q}_\alpha\dot{q}_h, \tag{90} $$

where,

$$ \eta = \begin{cases} 1, & \text{if } \alpha = h \\ 0 & \text{otherwise} \end{cases} $$

The rigid-flexible coupling terms $\dot{q}_h\dot{\delta}_j$ from (89) appear as:

$$ \sum_{i=1}^{n}\sum_{h=1}^{i}\sum_{j=1}^{m} Tr\left(\breve{T}_{h-1}U_h\,{}^{h}\tilde{T}_i D_{ij}T_i^T\right)\dot{q}_h\dot{\delta}_j \tag{91} $$

All terms contain terms $\dot{\delta}_j\dot{\delta}_k$ and can be expressed as:

$$ \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{m}\left(Tr\left(T_i(C_{ikj}+C^*_{ijk})T_i^T\right) + P_{ijk}\right)\dot{\delta}_j\dot{\delta}_k \tag{92} $$

We can further simplify (92) by using the fact that $Tr(A) = Tr(A^T)$ and T is of the form $\begin{bmatrix} R_{i_{3\times3}} & p_{i_{3\times1}} \\ 0_{1\times3} & 1 \end{bmatrix}$ and $(C_{ikj}+C^*_{ijk})$ is of the form $\begin{bmatrix} A_{3\times3} & 0_{3\times1} \\ 0_{1\times3} & 0 \end{bmatrix}$. We can write out the following property remembering that $R^{-1} = R^T$:

$$ Tr(TBT^T) = Tr\begin{bmatrix} RAR^T & 0 \\ 0 & 0 \end{bmatrix} = Tr(A) \tag{93} $$

Therefore, (92) becomes:

$$\frac{1}{2}\sum_{j=1}^{m}\sum_{k=1}^{m}\sum_{i=1}^{n}\Bigg(\int_{l_i}(\rho_i A_i {w_j^i}^T w_k^i + \rho_{_i} J_{_i} {\theta_{j_z}^i}^T \theta_{k_z}^i)dz + m_i w_j^i(l_i) w_k^i(l_i) + J_{t_i}\theta_j^i(l_i)\theta_k^i(l_i) + I_{t_{x_i}} w_{j_y}^{'i}(l_i) w_{k_y}^{'i}(l_i) + I_{t_{y_i}} w_{j_x}^{'i}(l_i) w_{k_x}^{'i}(l_i)\Bigg)\dot{\delta}_j\dot{\delta}_k = \frac{1}{2}\sum_{j=1}^{m}\sum_{k=1}^{m}\sum_{i=1}^{n} N_{ijk}\dot{\delta}_j\dot{\delta}_k, \tag{94}$$

where,

$$w_j^i = \begin{bmatrix} w_{j_x}^i(z) \\ w_{j_y}^i(z) \\ w_{j_z}^i(z) \end{bmatrix} \tag{95}$$

This is the same form for the the kinetic energy due to flexible states as derived in Appendix B using more traditional modeling methods.

3.3.4 Potential Energy

The system can have potential energy from two sources: gravity and link deformation. In general, joint elasticity should be added to the sources for potential energy but is outside the main scope of this work.

Gravitational potential energy can be calculated by:

$$dPE_{g_i} = -g^T T_i^{\,i} r_i dm, \tag{96}$$

where g is the gravity vector in the local link coordinates $g = [g_x, g_y, g_z, 0]^T$. The total gravitational potential energy is:

$$PE_g = -g^T \sum_{i=1}^{n} T_i(M_i h_{mi} + M_{t_i} h_{m_t i} + \sum_{k=1}^{m} \delta_k \epsilon_k^i), \tag{97}$$

where M_i is the total mass of the link, $h_{mi} = [0, 0, h_{zi}, 1]^T$ a vecotr to the center of the gravity from the origin of frame i, M_{t_i} is mass at the end of link i (either the mass of a joint or the mass of the end-effector), $h_{m_t i} = [0, 0, l_i, 1]^T$ a vector to the center of the gravity from the origin of frame i and $\epsilon_k^i = \int_{B_i}[w_{k_x}^i, w_{k_y}^i, w_{k_z}^i, 0]^T dm$ which is the deflection from the undeformed line on link i

The potential energy due to link deformations can be given by:

$$PE_e = \sum_{i=1}^{n}\left\{\frac{1}{2}\int_{l_i} E_i I_{yy_i}\Big(\frac{\partial^2 W_x^i}{\partial z^2}\Big)^2 dz + \frac{1}{2}\int_{l_i} E I_{xx_i}\Big(\frac{\partial^2 W_y^i}{\partial z^2}\Big)^2 dz + \frac{1}{2}\int_{l_i} G_i J_i\Big(\frac{\partial \Theta^i}{\partial z}\Big)^2 dz\right\} \tag{98}$$

Taking into account that $W_x^i = \sum_{j=1}^m \delta_j w_{j_x}^i$, $W_y^i = \sum_{j=1}^m \delta_j w_{j_y}^i$ and $\Theta^i = \sum_{j=1}^m \delta_j \theta_{j_z}^i$ equation (98) can be further broken down to:

$$I_{yy_i}(w_{j_y}^i{}'')^T w_{k_y}^i{}'' + E_i$$

$$PE_e = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{m}\left(\int_{l_i}(E_i I_{yy_i}(w_{j_y}^i{}'')^T w_{k_y}^i{}'' + E_i I_{xx_i}(w_{j_y}^i{}'')^T w_{k_y}^i{}'' + G_i J_i(\theta_{z_j}^i{}')^T \theta_{z_k}^i{}' dz\right)\delta_j\delta_k, \tag{99}$$

where, $()' \equiv \frac{\partial}{\partial z}()$, I_{xx_i}, I_{yy_i} are the second moments of area in the x_i and y_i directions, respectively; E_i is the Young's modulus of link i and G_i is the shear modulus of link i. Equations (97) and (99) are summed to form the total potential energy for the system:

$$PE = PE_g + PE_e \tag{100}$$

3.3.5 Equations of Motion

Using the energy expressions derived in the previous section, the equation of motion are derived using Lagrange's Equation:

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{x}_i}\right) - \frac{\partial KE}{\partial x_i} + \frac{\partial PE}{\partial x_i} = Q_i, \tag{101}$$

Equations (90)-(92) are used to find the inertia matrix of the system. In the case of linear vibrations it can be assumed that the effect of deflection variables on the intertia matrix is negligible ($\delta = 0$). The inertia matrix can be partitioned in the following way:

$$M = \begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & M_{\delta\delta}(q) \end{pmatrix}, \tag{102}$$

where the elements of M_{qq} are defined as:

$$M_{qq_{h\alpha}} = \sum_{i=1}^{n} TR[\breve{T}_{h-1} U_h^h \tilde{T}_i F_i \left(\breve{T}_{\alpha-1} U_\alpha{}^\alpha \tilde{T}_i\right)^T] + J_{i_{mot}}\eta, \tag{103}$$

Similarly the elements of $M_{q\delta}$ are:

$$M_{q\delta_{hj}} = \sum_{i=1}^{n} TR\left(\breve{T}_{h-1} U_h^h \tilde{T}_i D_{ij} T_i^T\right), \tag{104}$$

and, finally the elements of $M_{\delta\delta}$

$$M_{\delta\delta_{kj}} = \sum_{i=1}^{n}\sum_{i=1}^{n} N_{ikj} \tag{105}$$

The dimensions of the matrices are as follows:

$$h, \alpha = 1, 2...n$$

$$k, j = 1, 2...m$$

Per [135] the matrix that contains the Coriolis and centrifugal terms can be defined to be $C(X, \dot{X})\dot{X}$ where $C(X, \dot{X})$ is the matrix of Christoffel symbols of the first kind and can be calculated as:

$$C(X, \dot{X})_{ij} = \frac{1}{2} \sum_{k=1}^{n+m} \left(\frac{\partial M_{ij}}{\partial X_k} + \frac{\partial M_{ik}}{\partial X_j} - \frac{\partial M_{kj}}{\partial X_i} \right) \dot{X}_k \tag{106}$$

where $i = 1, 2...(n+m)$. Note that although δ is assumed to be negligible, its derivatives can have a big influence and therefore must be included in the derivation.

Taking derivatives of the potential energy (100) yields the gravity terms and the stiffness matrix. The gravity terms can be represented as:

$$g_{q_j} = -g^T \sum_{i=j+1}^{n} \frac{\partial T_i}{\partial q_j} (M_i h_{mi} + M_{t_i} h_{m_t i} + \sum_{k=1}^{m} \delta_k \epsilon_k^i), \tag{107}$$

where $j = 1, 2...n$. The summation index starts from j+1 because q_1 does not appear until transfer matrix A_2 in $T_i = A_1...A_n$

$$g_{\delta_j} = -g^T \sum_{i=1}^{n} T_i \epsilon_j^i, \tag{108}$$

where $j = 1, 2...m$. The stiffness matrix K_{ff} can be found in the following way:

$$K_{\delta\delta_{jk}} = \sum_{i=1}^{n} \left(\int_{l_i} (E_i I_{yy_i} (w_{j_y}^i{}'')^T w_{k_y}^i{}'' + E_i I_{xx_i} (w_{j_y}^i{}'')^T w_{k_y}^i{}'' + G_i J_i (\theta_{z_j}^i{}')^T \theta_{z_k}^i{}' dz \right) \tag{109}$$

The only stiffness in the system is due to the link deformations. The generalized force comes from virtual work done by the actuator forces:

$$\delta W_i = \tau_i \delta q_i \tag{110}$$

Finally, using equations (102), (106), (107), 109, and (110) yields the full equation of motion for the system:

$$M(X)\ddot{X} + C(X, \dot{X})\dot{X} + KX + G(X) = Q, \tag{111}$$

In general, $Q = \begin{pmatrix} \Gamma \\ 0 \end{pmatrix}$ and $\Gamma = \tau$, where τ is the vector of joint torques. The expanded equations of motion for a n-link flexible serial manipulator with m significant modes are as follows:

$$\begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & M_{\delta\delta}(q) \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} C_r(q,\dot{q},\dot{\delta}) \\ C_f(q,\dot{q},\dot{\delta}) \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} g_r(q) \\ g_f(q) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & K(q) \end{pmatrix} \begin{pmatrix} q \\ \delta \end{pmatrix} = \begin{pmatrix} \Gamma \\ 0 \end{pmatrix}, \quad (112)$$

where, M is the $(n+m)$ x $(n+m)$ generalized mass matrix, h is the $(n+m)$ vector matrix containing the Coriolis and centrifugal terms and the terms accounting for interaction of joint variables and their rates and flexible variables and their rates, C is the $(n+N)$ vector of gravitational terms, K is the $(m \times m)$ flexural stiffness matrix, Γ is the n-vector of torques applied by the motors.

Note that the differential equations governing the flexible and rigid states are fully coupled. In addition, the only input to the system Γ goes directly into the equation governing the rigid states q, while the flexible states δ are excited only through the coupling terms. Modal damping can easily be added to the equation.

Note that the flexible sub-system is only valid for a specific configuration of the manipulator, therefore the model needs to be constantly recomputed to get the correct response of the system.

3.3.6 Model Continuity

Because the model derived in the previous section is dependent on the system mode-shapes (that by definition can be of an arbitrary magnitude) that vary based on the joint configuration of arm, care must be taken to ensure that the model is continuous over time. In general, the eigenvalue problem for the flexible states can be given in the form:

$$\left(K_{\delta\delta}(q) - \omega^2 M_{\delta\delta}(q)\right) \phi(q) = 0 \quad (113)$$

This problem was examined by Bruls [20] where it was shown that if the modes are normalized with the conditions:

$$M_{\delta\delta}(q) = I \quad (114)$$

then the model is continuously smooth in the configuration space. Relationship (114)is always solvable because the modes are orthogonal to one another by definition when using exact system-wide modes. This will also lead to:

$$K_{\delta\delta} = \Omega^2(q), \tag{115}$$

where $\Omega(q)$ is the diagonal matrix of the modal natural frequencies. The structure of the model also matches the one found in [20].

For a selected eigenvalue ω_i and the associated mode ϕ_i, (113) can be differentiated with respect to a joint coordinate q_j to show the continuous nature of the modes and natural frequencies.

$$\left(\frac{\partial K}{\partial q_j} - \omega_i^2 \frac{\partial M}{\partial q_j} - \frac{\omega_i^2}{\partial q_j}\right)\phi_i + (K - \omega_i^2 M)\frac{\phi_i}{\partial q_j} = 0 \tag{116}$$

Pre-multiplying (116) with ϕ_i^T eliminates the second terms since it is in the kernel of $(K - \omega_i^2 M)$, and gives the term $\frac{\omega_i^2}{\partial q_j}$, while taking into account that $M = I$:

$$\frac{\omega_i^2}{\partial q_j} = \phi_i^T \left(\frac{\partial K}{\partial q_j} - \omega_i^2 \frac{\partial M}{\partial q_j}\right)\phi_i \tag{117}$$

Because the right side of (117) is well defined, it can be concluded that ω_i exhibits smooth variations with respect to the change of q_j. ϕ_i in general can have have an arbitrary magnitude because it is the kernel of $(K - \omega_i^2 M)$. However, the normalization condition forbids it. Therefore it can be concluded that the proposed model is well-defined, as long as the necessary conditions are taken for the selection and normalization of the mode-shapes. A more thorough analysis of the smooth variation of modes can be found in [20]. Therefore, the equations of motion (112) can be rewritten as:

$$\begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & \mathbb{I} \end{pmatrix}\begin{pmatrix} \ddot{q} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} C_r(q,\dot{q},\dot{\delta}) \\ C_f(q,\dot{q},\dot{\delta}) \end{pmatrix}\begin{pmatrix} \dot{q} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} g_r(q) \\ g_f(q) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & K_{\delta\delta}(q) \end{pmatrix}\begin{pmatrix} q \\ \delta \end{pmatrix} = \begin{pmatrix} \mathbb{I} \\ 0 \end{pmatrix}\tau, \tag{118}$$

3.3.6.1 Changing mode order

The changing configuration of the serial structure could lead to conditions where variation of natural frequencies w_r and w_k cause their order to switch or can even lead to a

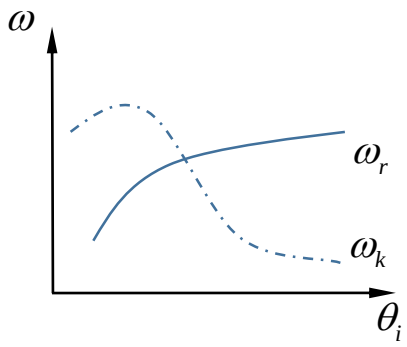


Figure 20: Crossing Natural Frequencies

double root as, is illustrated in Figure 20. The Modal Assurance Criterion (MAC) can be used to track the progression of these modes from one configuration to the next [104]. In general, the MAC value between mode 1 and mode 2 is a number between 0 and 1, where 0 indicates no correlation between the modes, and 1 is a perfect match between the modes. For one dimensional mode shapes ϕ_i, the MAC condition is:

$$MAC_{1,2} = \left(\frac{\phi_1 \phi_2^T}{\sqrt{\phi_1 \phi_1^T} \sqrt{\phi_2 \phi_2^T}} \right)^2 \tag{119}$$

For mode shapes defined in 3D space, over multiple links, the MAC condition can be altered to take the form:

$$MAC_{1,2} = \left(\frac{\sum_{i=0}^{n} \left(w_{1_x}^i {w_{2_x}^i}^T + w_{1_y}^i {w_{2_y}^i}^T + w_{1_z}^i {w_{2_z}^i}^T \right)}{\sqrt{\sum_{i=0}^{n} \left(w_{1_x}^i {w_{1_x}^i}^T + w_{1_y}^i {w_{1_y}^i}^T + w_{1_z}^i {w_{1_z}^i}^T \right)} \sqrt{\sum_{i=0}^{n} \left(w_{2_x}^i {w_{2_x}^i}^T + w_{2_y}^i {w_{2_y}^i}^T + w_{2_z}^i {w_{2_z}^i}^T \right)}} \right)^2 \tag{120}$$

The MAC value, however, does not tell us if the computed mode shape has the correct sign value.

A simple cantilever beam is used to illustrate this problem in Figure 21, where both ϕ and $\bar{\phi}$ satisfy the normalization and the MAC value criteria. Hence, for mode shapes to be continuous, they need to be re-calculated with the following expression:

$$Mode_1 = Mode_1 \times sign \left(\sum_{i=0}^{n} \left(w_{1_x}^i {w_{2_x}^i}^T + w_{1_y}^i {w_{2_y}^i}^T + w_{1_z}^i {w_{2_z}^i}^T \right) \right) \tag{121}$$

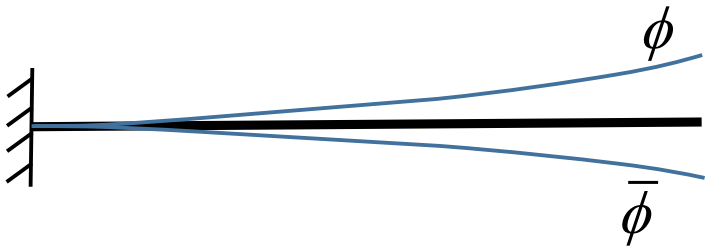


Figure 21: First Mode Shape for a Simple Cantilever Beam

This expression makes sure that the mode shapes are always on the same "side" (above or below in the case of Figure 21) of the links.

When the system exhibits a double root, any linear combination of the two mode shapes $\alpha\phi_r + \beta\phi_k$, where α and β are linear parameters, is a valid mode shape mathematically, as long as they are orthogonal to one another. Therefore, for computational simplicity, it is advised that the double root not be used to calculate the mode shapes, however, to preserve continuity, they can be approximated based on the ones in the direct vicinity ($\theta + \Delta\theta_i$, where $\Delta\theta$ is sufficiently small) of the double root.

3.3.6.2 Strategies for Computation

Recalculating the system natural frequencies and mode shapes for the full system equations of motion (118) during execution is computationally taxing. However, one can take advantage of the fact that all of the modes can be computed *a-priori* based on the joint configuration space, if all of the end-effector loads are known. Two strategies for generating the terms in (118) include: 1) spline fit, where all of the terms in the matrices are fit to a polynomial based on the joint angles q; and 2) piecewise fit where the model is discretized in the configurations space. An illustration of the piecewise fit method is shown in Figure 22 where the terms that are dependent on the mode shapes are chosen based on the current joint configuration (q_i and q_j in this case). The values Δq must be chosen small enough so that the model can still be considered to be varying smoothly.

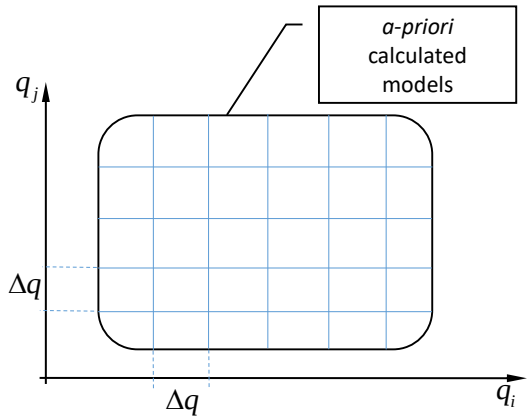


Figure 22: Piecewise Fit for Computing the Equations of Motion

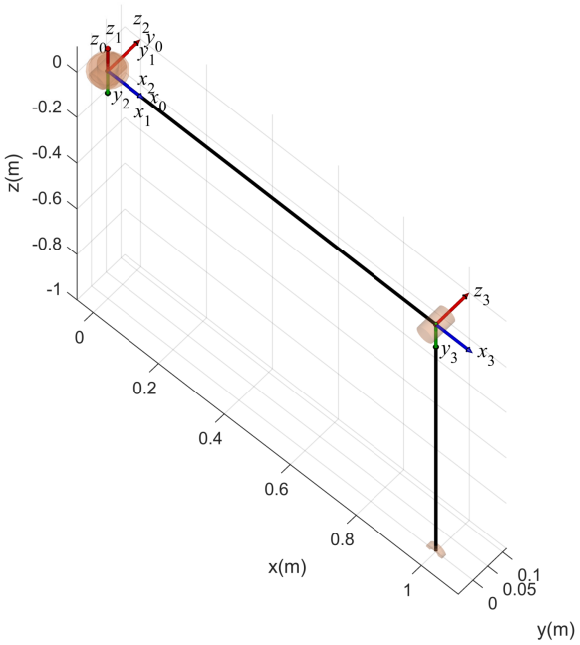


Figure 23: 3 Joint 2 Link Robot With Coordinate Frames in the DH Convention

3.3.7 Case Study

In order to illustrate the effectiveness of this model, simulations were carried out, and were then compared to results obtained with commercial FEA software (Comsol in this case.) The robot arm used in the simulations had 2 flexible links and 3 joints. Figure 23 shows the robot arm in the configuration where all joint angles q are 0. All of the parameters for the robot arm are shown in Table 8, where $dh.x$ are the DH parameters presented in

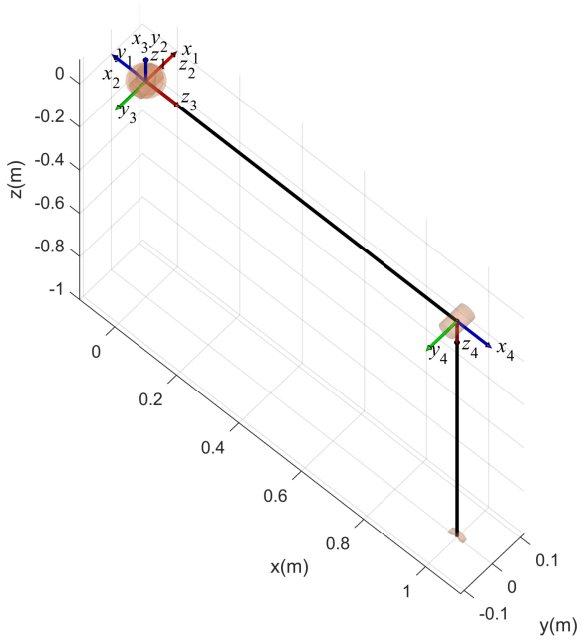


Figure 24: 3 Joint 2 Link Robot Arm With C-ordinate Frames in the Local Link Convention

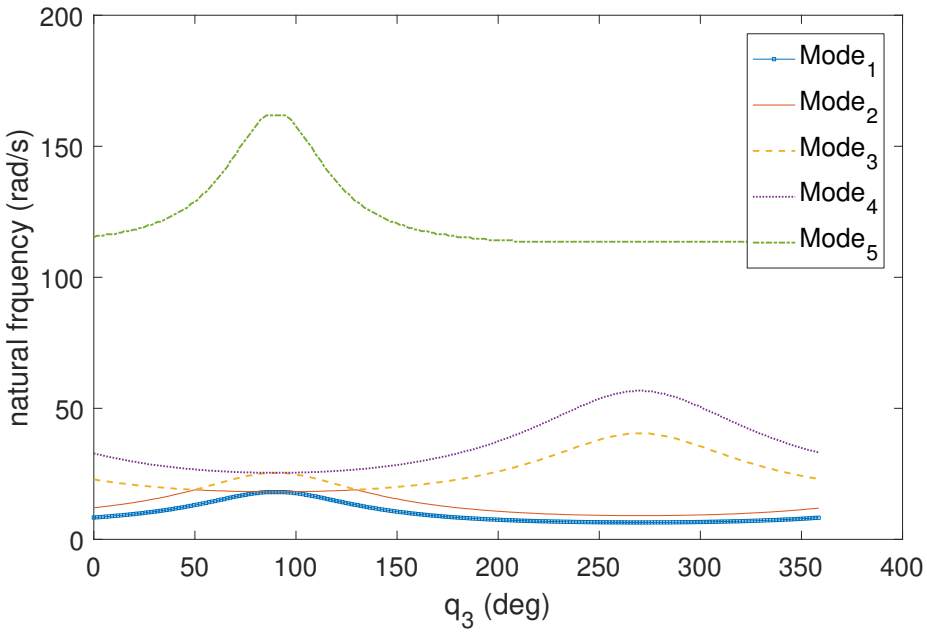


Figure 25: Varying Natural Frequencies of The Robot Arm Due to Change in q_3

Section 3.1, a_i is the link thickness in the local y_i direction, and b_i is the thickness in the local x_i direction, ρ is the density of the links, G is the shear modulus, E is the Young's modulus, m_t is the mass at the end of the links, and the I terms are the moments of inertia of the mass at the end of each link. The local link coordinate frames are shown in Figure

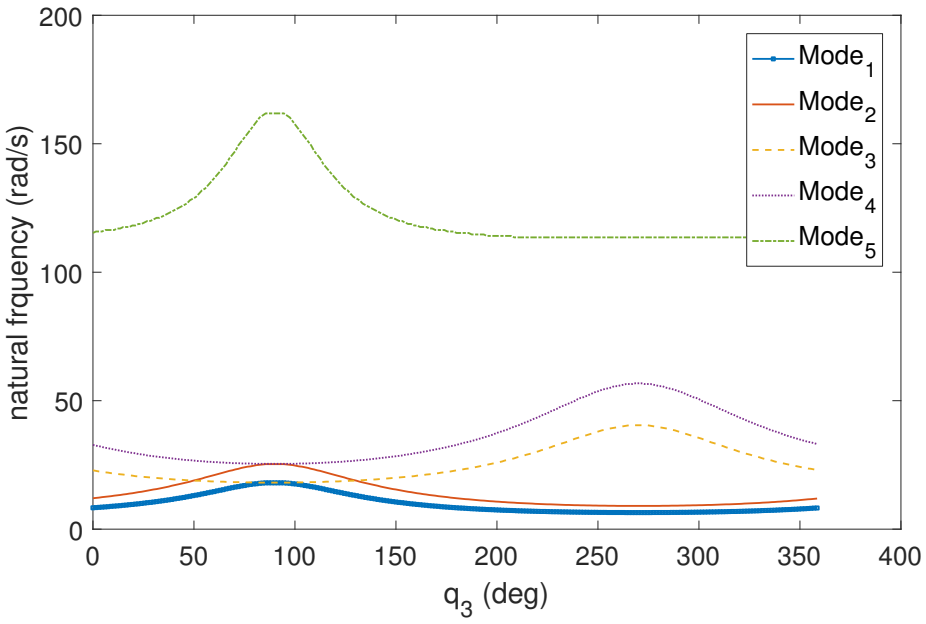


Figure 26: Varying Natural Frequencies of The Robot Arm Changing Continuously

24. Note that two fictitious links of zero length have been used to make the desired robot configuration obtainable with the DH parameters. Figure 25 shows the natural frequencies with respect to the elbow angle q_3 calculated every 1.5°. There is a flat part in the graph at 90° where link 2 coincides with link 1 and there are numerous double roots that are approximated based on the modes next to 90°. Notice that there are large variation in the structural frequencies based on q_3. It can be observed that in numerous configurations the modes cross each other in the mode order, however, it is not intuitively clear which line the modes actually follow. Figure 26 shows the modes when the modified MAC value algorithm has been used to track the modes, as was described in Section 3.3.6.1. As can bee seen in the figure, the frequencies change smoothly with respect to q_3.

3.3.7.1 Simulations

The simulations were carried out such that the joint angles q followed a prescribed trajectory, i.e, perfect tracking of the joint trajectories. The torques that would have been needed to carry out such a motion was calculated from the first row of (118). The flexible variables were calculated from the second row of (118). All of the moves performed in this

Table 8: Robot Parameters

i	0	1	2	3	4
$dh.\theta$ (rad)	-	q_1	q_2	q_3	0
$dh.d$ (m)	-	0	0	0	1
$dh.a$ (m)	0	0	1	0	-
$dh.\alpha$ (rad)	0	$-\frac{\pi}{2}$	0	$-\frac{\pi}{2}$	-
a (m)	-	0	0	0.005	0.005
b (m)	-	0	0	0.007	0.007
ρ kg/m^3	-	0	0	2700	2700
G (GPa)	-	0	0	26.9	26.9
E (GPa)	-	0	0	69.0	69.0
m_t (kg)	-	0	0	0	0
I_{x_t} (kgm^2)	-	0	0	0	0
I_{y_t} (kgm^2)	-	0	0	0	0
I_{z_t} (kgm^2)	-	0	0	0	0

Table 9: Joint Trajectories for Verification

	q_{1_0}	q_{1_f}	q_{2_0}	q_{2_f}	q_{3_0}	q_{3_f}
Move 1	0	-1	0	-1	0	-1
Move 2	0	0	0	-1	0	-1
Move 3	0	-π	0	0	0	0

study used bang-bang acceleration profiles which leads to a "S-curve" in the positions as illustrated in Figure 27. The positive acceleration lasts for 1 second followed by a negative acceleration for the same time duration. Five significant modes were calculated in all of the cases, although in most cases two or three modes would have been enough because the amplitudes for the higher modes were negligible. Gravity was neglected in all of the simulations. The three moves shown in Table 9 describe the joint trajectories used for simulations. q_{i_0} represents the initial angle of joint i in rad and q_{i_f} represents the final angle of joint i for the move.

During move 1, when all joints move from 0 to -1 rad with a 2 second move time. Results from simulating this move are shown in 28. The δ variables shown are the flexible variables used in (118). The first and second modes had the biggest effect on the structure. Figure 29 shows the location of the end-effector compared to results obtained from FEA program

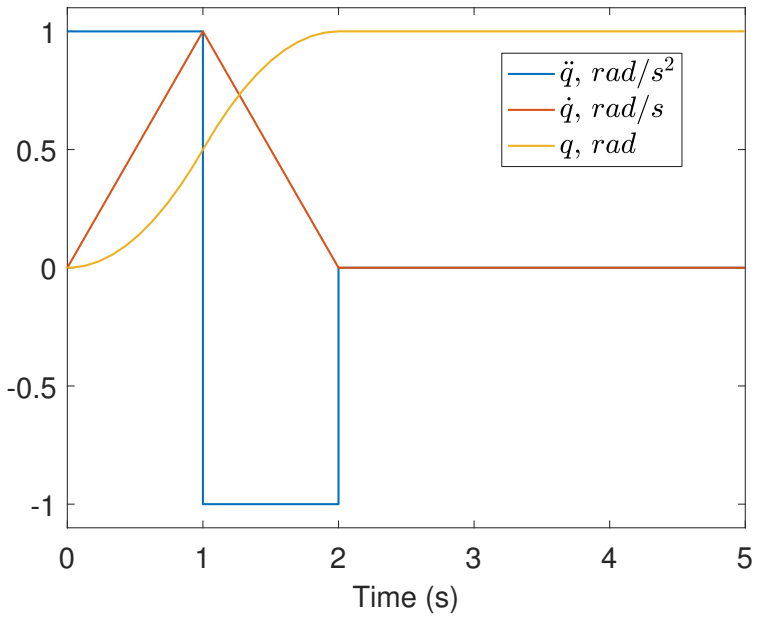


Figure 27: "S-curve" Motion with a 2 Second Move Time

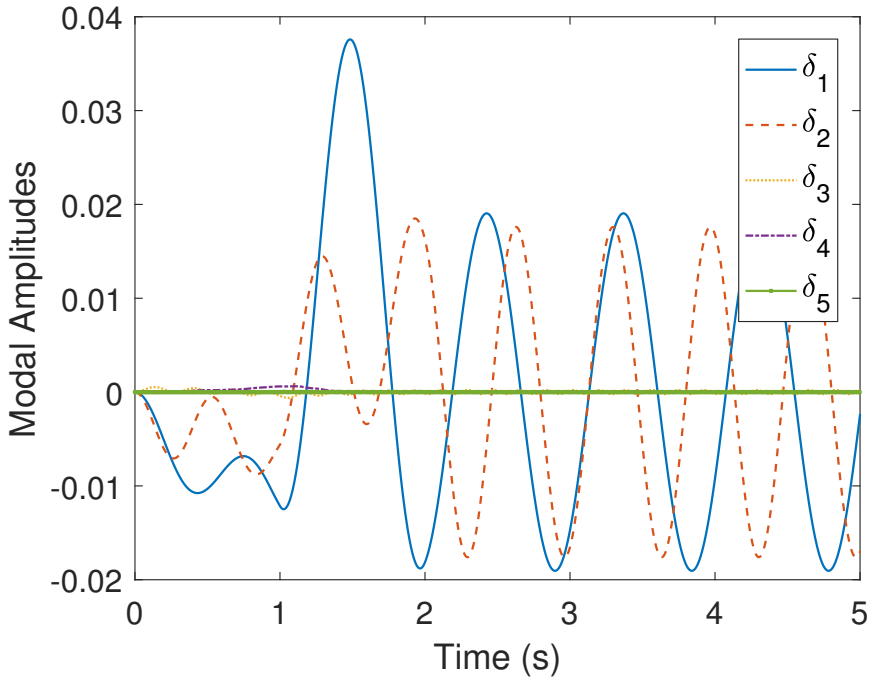


Figure 28: Flexible Variables for Move 1

Comsol. It is clear that the proposed model produces results that are almost identical to the FEA software. Note that because all of the joints are moving the flexible subsystem exhibits variable mode shapes. The torques that are required to produce that move based on the

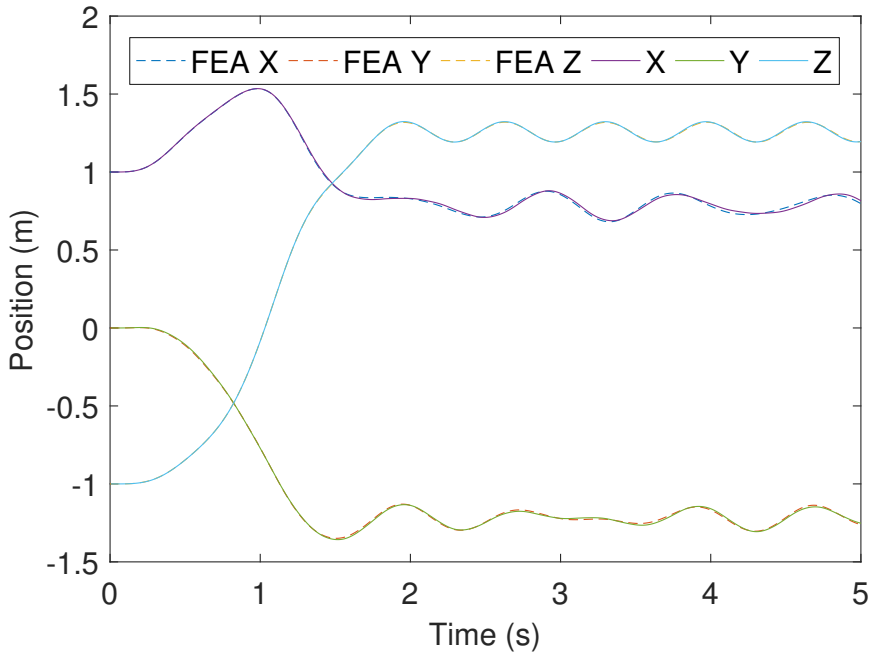


Figure 29: Tip Displacement Compared to FEA for Move 1

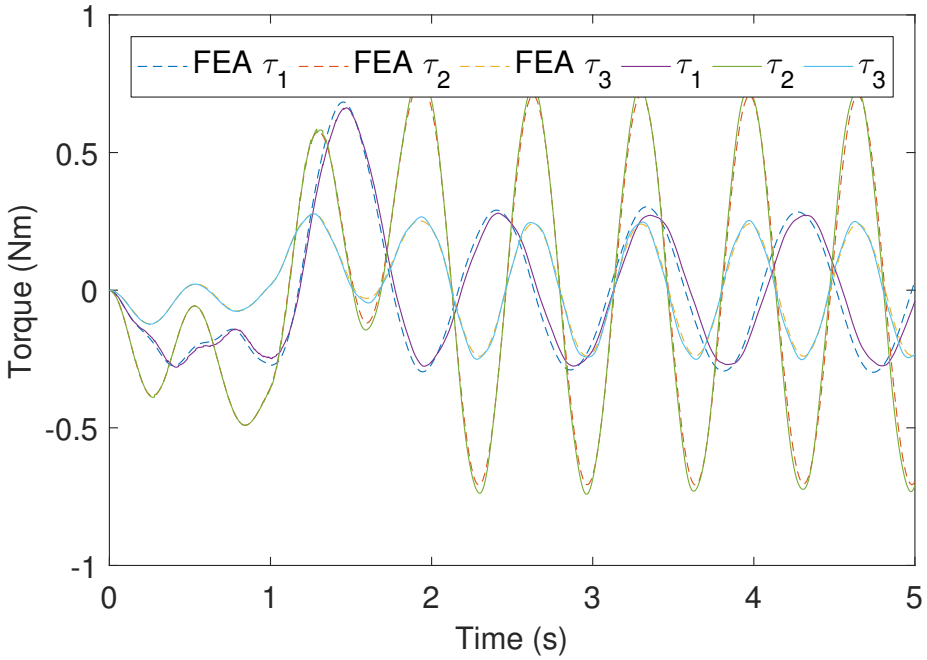


Figure 30: Torques compared to FEA for Move 1

FEA model and the proposed model are shown in Figure 30. There are minor differences in phase, but the magnitude of the torques remains very close throughout the move.

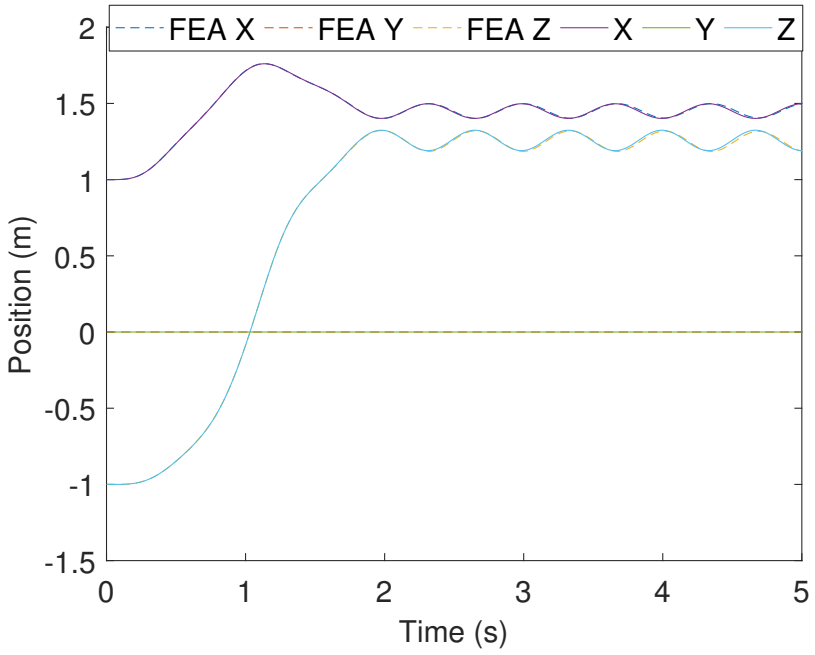


Figure 31: Tip Displacement Compared to FEA for Move 2

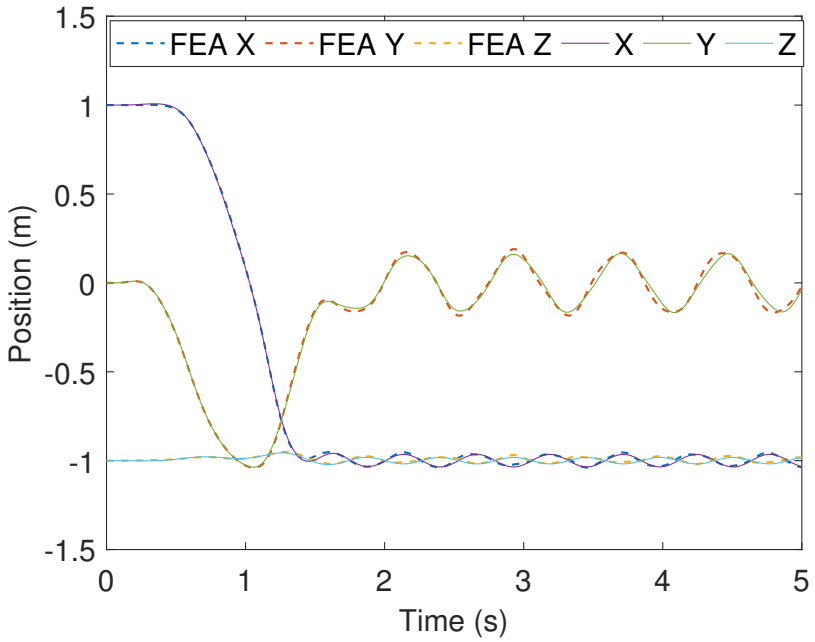


Figure 32: Tip Displacement Compared to FEA for Move 3

Move 2 is defined by moving joints 2 and 3 from 0 to π in 2 seconds, while keeping joint 1 locked. This move corresponds to a purely planar trajectory; however because joint 3 is moving the mode shapes of the system change with the system configuration. Figure 31

shows the results obtained from the proposed model compared to the FEA model. Similarly to move 1 the displacements are almost indistinguishable when comparing the two modeling methods. The Z co-oridinate is 0 since the move is purely planar.

Move 3 is defined by having joint 1 move from 0 to π in 2 seconds, while keeping joints 2 and 3 locked. This move corresponds to the configuration shown in Figure 23 being rotated around the Z axis. Figure 32 shows the tip displacement based on the proposed model and the FEA simulation. The response from the TMM based model is very close to the FEA generated one During this movement the terms in (118) that depend on mode shapes do not need to be recalculated since the mode shapes of the system do not change with the motion of q_1.

These results are remarkable because the responses from the proposed model and the commercial FEA simulation are almost identical, yet the basis for calculation each of them are vastly different. The Comsol simulations took an average of 4 minutes to compute at 10 ms sampling rate. While the TMM based model that was implemented in Simulink, completed the same simulations in less that 2 seconds with a 1 ms sampling rate. This implies that the proposed model is suitable for real time implementations without losing fidelity, unlike the current FEA models.

3.3.8 Comparison with other AMM modeling techniques

The major benefit of modeling serial-elastic actuators with the methods described in the previous section is that the number of state variables will be less than with previous works, for example [57, 87, 134, 152]. The benefit comes from the fact that we a given mode shape and its modal amplitude for the entire n-link structure and, therefore, the number of control variables is $n-1+m$, where n is the number of links, and m is the number of significant modes. When using traditional approaches the number of states is $n-1+m\cdot n\cdot d$, where n is the number of links, m is the number of significant modes per link and d is the number of relevant compliant directions. For example, a 4 link robot with 2 significant modes, 2 flexural directions of interest, and one torsional deflection, then the number of state variables based on the TMM based approach is $3+2=5$. Previous methods would

typically have $3 + 2 \cdot 4 \cdot 3 = 27$ state variables.

The downside of this proposed method is that the entire model needs to be recalculated for every robot configuration. Fortunately, this can be done off-line. Depending on the boundary conditions used by the traditional methods the model may or may not have to be recalculated. In previously reported experiments, researchers have used one set of mode shapes for the entire task-space, while claiming that the results were good enough to damp out the oscillations [40, 90, 136].

An illustration of why the TMM approach is in some ways superior to the traditional methods can be achieved by comparing the mode shapes and the natural frequencies they produce. Consider a two link serial arm with an angle of $2.75rad$ between the links is. The lengths of the links are 1 and 1.5 m. There is a tip mass of 1 kg added to the end of the second link. The flexural rigidity used is 10.7813 Nm^2. Only 1D flexure is considered. The natural frequencies obtained from TMM are compared to FEA and the other boundary conditions used by some researchers in the past. The derivation on how to obtain natural frequencies from the "traditional" approach to modeling flexible robot arms is shown in Appendix C. The results are shown in Table 10. It can be seen that there are serious shortcomings when the full system modes are not taken into account. “C" in the table means clamped, “P" pinned, “F" free and “M" mass. “C-M" for example refers to the use of clamped boundary conditions at the start of the link and mass at the distal end. To reiterate, this example was

Table 10: Natural Frequencies for Different BCs

BCs	ω_1	ω_2
TMM	1.3750	16.1550
FEA	1.3610	15.808
C-F	2.5920	14.6689
P-P	32.0045	72.0160
P-F	11.3444	38.0352
C-C	72.4	199.5
C-P	49.9330	112.2779
C-M	2.5884	14.6355

the simplest case where flexure was only in one direction and no torsinal effects were taken

into consideration. It can be seen that it makes more sense to obtain the mode shapes and natural frequencies with the TMM, rather than with the methods used in the past for serial flexible robot arms.

3.3.9 Chapter Summary

A systematic approach to finding natural frequencies and mode-shapes for n-link spatial serial structures was presented. The model was validated using Finite Element Analysis and experiments. There was good agreement between the predictions and the experimental measurements. This method could be used for generating more efficient models for diverse applications such as flexible robotics, and characterizing vibrations in solar panel arrays, and lattice crane structures.

A new low-order dynamical model based on varying mode-shapes for serial flexible robot arms was presented. The model was verified with commercial FEA software, that produced results very similar to the proposed method. The main benefits of this model are the systematic approach for derivation and the computation speed compared to other high fidelity models. In the following chapters this model will be used for estimation and control of flexible serial robot arms.

Chapter IV

DEVELOPMENT OF A FLEXIBLE SERIAL ARM TESTBED

The literature regarding flexible robots is filled with modeling techniques and controller approaches that have never been validated on hardware, as was discussed in Chapter 2. In order to verify the claimed contributions of dissertation, real-world experiments are in order. A new multi-link flexible robot test bed was designed and built following inspiration from robots described in Section 2.4. The following goals were adhered to in the design of the flexible manipulator arm:

- Motion capability in all spatial coordinates
- Fabrication using "standard" manufacturing equipment
- Reconfigurability of the links on the arm
- Standard communication protocols with all electrical components
- Large rotations that illustrate changing modes and natural frequencies of the system
- The lowest natural frequencies of the flexible arm should be significantly below the control bandwidth of the motors

The need to use standard manufacturing equipment comes from the fact that having parts done professionally adds significant time between design iterations in addition to high costs. Since this test bed is to be used for research, standard communication protocols are required so that the user does restrict themselves to a specific software or hardware ecosystem, thus forbidding to use of products from a variety of vendors.

4.1 Physical Design

The robot testbed is shown in Figure 33 in a configuration with the links pointing roughly downward. The robot is mounted on overhead horizontal beams with an aluminum bracket.

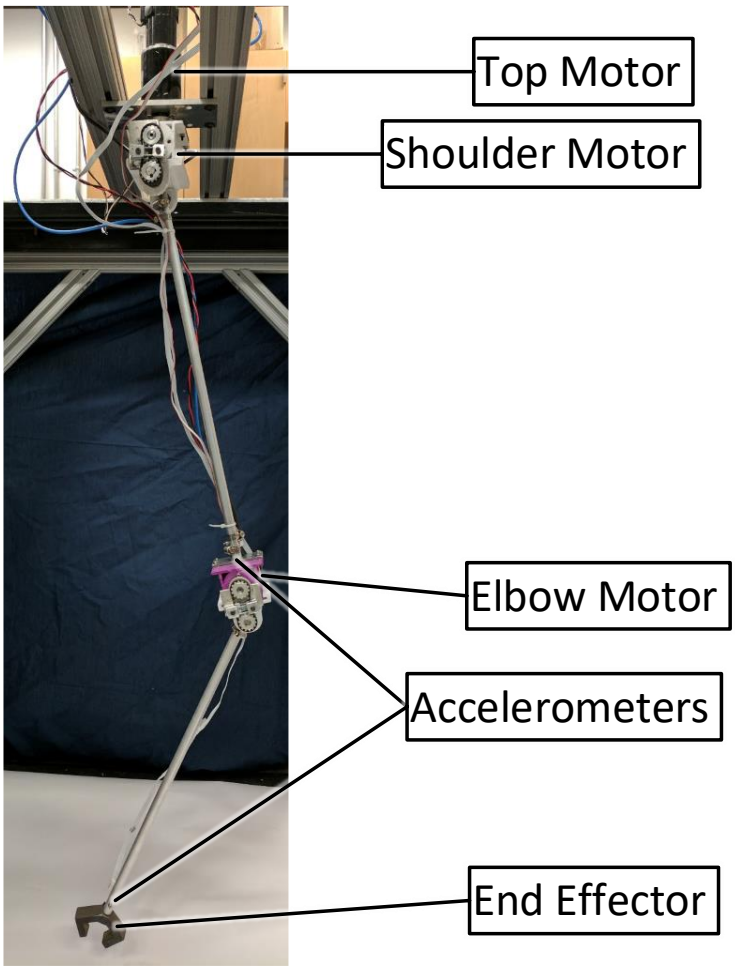


Figure 33: 2-Link Flexible Serial Arm Test Bed

The end-effector on the robot is made of steel. A close up of the elbow joint can be seen in Figure 34; the design of the shoulder joint is analogous. The belt on the joints is of type GT2 to handle torque well and provide sufficient positional accuracy. The shafts have keys between them and the pullies and link couplers to avoid slipping. Belt tighteners were added to avoid backlash in the system. The joint housings were manufactured using 3D printing with PLA material.

The motors chosen for the testbed are from Harmonic Drive. Models RH-14D, RH-11D, and RH-8D are used for the top motor, shoulder motor, and elbow motor respectively. These motors were chosen for their high positioning precision, suitable rated speed, and torque characteristics. The most important specifications of the motors are listed in Table

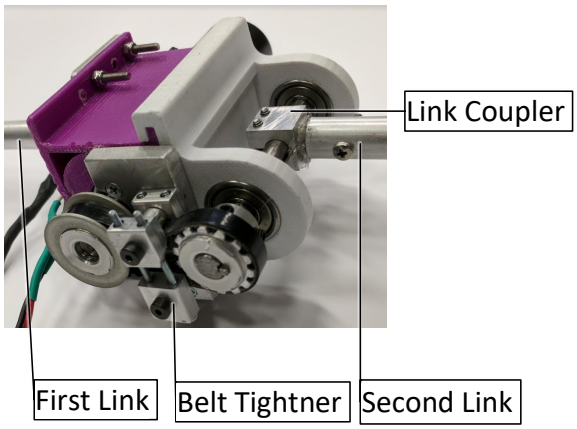


Figure 34: Elbow Joint

Table 11: Motor Parameters

	Top Motor	Shoulder Motor	Elbow Motor
Gear Reduction	1:50	1:50	1:50
Mass(kg)	0.77	0.5	0.3
Max Momentary Torque (Nm)	14	4.9	2.7
Rated Torque (Nm)	3.2	2.2	1.4
Max Speed (r/min)	100	100	100
Rated Speed (r/min)	60	60	60
Rated Voltage (V)	24	24	24
Encoder Resolution* ($count/r$)	1000	1000	1000

*before gearbox

11. All of the characteristics are shown in Appendix D.

The links are comprised of very thin walled tubes made out of 3003 Aluminum. They have 12.7 mm outer diameter and a wall thickness of 0.41 mm. The length of Link 1 is 0.47 m and Link 2 is 0.48 m. Note that the length of the joints and end effector are not included in lengths of the links, therefore, the "link-length" in the DH parameters are longer than the values given here.

Figure 35 depicts how the coordinates on the robot are defined. Joint q_1 allows for rotation about the vertical axis. q_1 is not physically limited (in control software, it is limited to 0... – 180°). Joints q_2 and q_3 are limited between 0...180° and 0... – 180°, respectively, due to the design of the joints. The dashed lines show the robot configuration in the limit

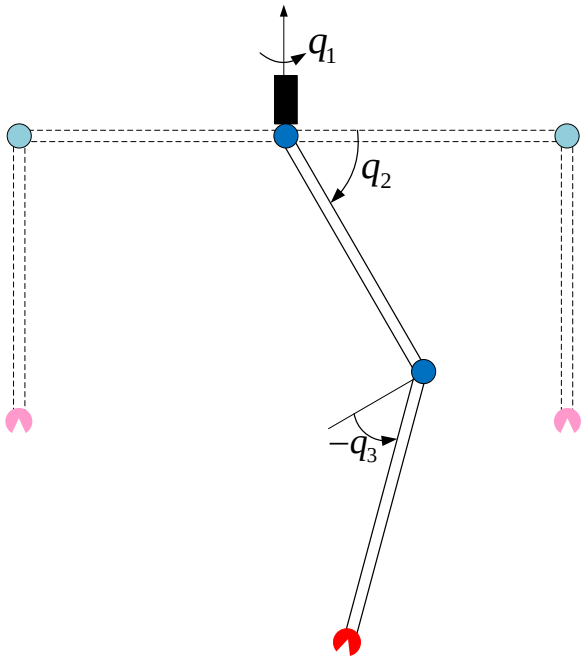


Figure 35: Joint Definitions (Dashed Lines Show Limiting Configurations)

conditions. For the right dashed arm, $q_2 = 0$ and $q_3 = 0$. For the left dashed arm $q_2 = 180°$ and $q_3 = -180°$. The reason for negative magnitudes for the coordinates comes from the DH convention. The parameters of the the robot used in the Dynamical Model presented in Chapter 3 are shown in Table 12.

4.1.1 Alternative Design

The use of belt drives on this test bed was not the first choice, but rather a practical solution that achieves good performance. Figure 36 shows a design where the motion is connected to the link directly. This design requires one coupler that is fixed to the frame of the motor and a rotating coupler that is connected to the output shaft of the motor and to the body of the motor through a slim bearing. The benefits of this design include no additional backlash, friction, flexure due to belts. In addition, the mass of the joint would be concentrated at the axis of the joint.

Numerous attempts were made to achieve the direct-drive design; however suitable motors could not be obtained. This design is difficult to manufacture because all tolerances

Table 12: Robot Parameters

i	0	1	2	3	4
$dh.\theta$ (rad)	-	q_1	q_2	q_3	0
$dh.d$ (m)	-	0	0	0	0.48
$dh.a$ (m)	0	0.53	0	0	-
$dh.\alpha$ (rad)	0	$-\frac{\pi}{2}$	0	$-\frac{\pi}{2}$	-
id (m)	-	0	0	0.00127	0.00127
od (m)	-	0	0	0.0127	0.0127
ρ kg/m^3	-	0	0	2700	2700
G (GPa)	-	0	0	26.9	26.9
E (GPa)	-	0	0	69.0	69.0
m_t (kg)	-	0	0	0.53	0.15
I_{x_t} (kgm^2)	-	0	1120e-6	517e-6	25.2e-6
I_{y_t} (kgm^2)	-	0	0	72e-6	78.5e-6
I_{z_t} (kgm^2)	-	0	0	520e-6	56.700e-6
J_{mot} (kgm^2)	-	21.6e-3	11.0e-3	3.7e-3	-

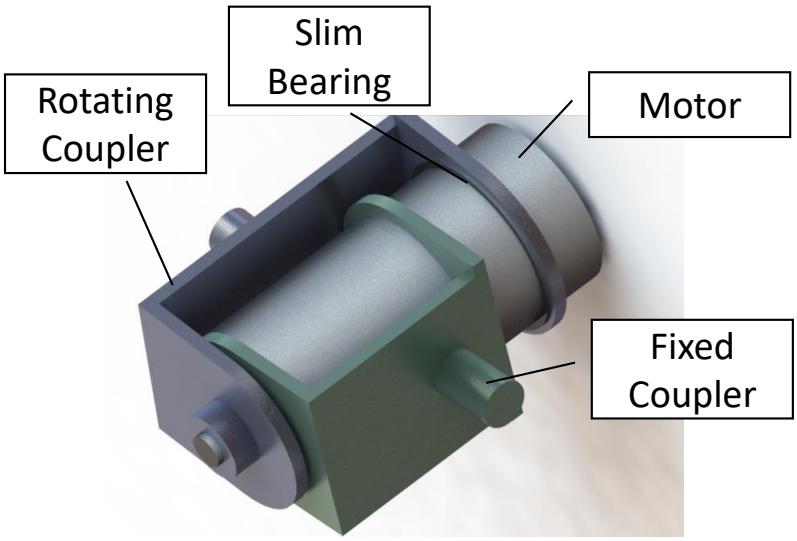


Figure 36: Alternative Joint Design

need to be low to mitigate wobble in the links.

4.2 Sensor Selection

While the joint angles and velocities are readily measurable from encoders and tachometers, the flexible states, δ, are not directly measurable and an observer must be used. The model for the observer is dependent on what type of sensors are used. The following sections describe the pros and cons for common sensors used in flexure sensing.

4.2.1 Vision

Direct measurement of link flexure is possible with optical sensing, such as with cameras. The location of a point for a beam when measured with a camera can be represented as:

$$x = w(x,t) = \sum_{j=1}^{m} \delta_j(t) w_j(x), \tag{122}$$

where m is the number of significant modes, δ_j is the amplitude of the jth mode shape and $w_j(x)$ is the mode shape evaluated at the measured point. While flexure with vision based sensors seems straight forward, there are several cons in the following list:

- Cons
  - For fast measurements, equipment is very expensive
  - Accurate 3D vision systems require numerous cameras
  - Lighting in the room can affect measurements
  - Markers on the structures can be occluded from the cameras' point of view
- Pros
  - Flexure estimation from vision data is straight forward
  - Measurements can be very accurate

After considering these design trade offs, it was concluded that a vision system, is not a practical sensor solution for sensing flexure, outside of a laboratory environment. Although the work done in this dissertation is done in a such an environment, the goal is for this work to be used in a variety of applications.

4.2.2 Strain Gages

Strain gages are by far the most common sensor used in structural vibrations. They work on a simple principle where the electrical resistance varies in proportion to the amount of strain in the device. Bonded metallic strain gages that consist of very fine wire or metallic foil arranged in a grid pattern are most widely used. It is known that in bending strain on

a beam can be represented as:

$$\epsilon = -a\frac{d^2W}{dx^2}, \tag{123}$$

where, ϵ is the measured strain, a is the distance from the neutral axis of the beam to the surface where the strain gage lies and $\frac{d^2W}{dx^2}$ is the curvature of the beam, at the location of the strain gage. Furthermore, using the modal representation, the curvature of the beam can be represented as:

$$\frac{d^2W}{dx^2} = \sum_{1}^{m} \delta_j(t) w_j''(x), \tag{124}$$

where m is the number of significant modes, δ_j is the jth modal amplitude and $w_j''(x)$ is the second spatial derivative of the mode shape.

Regardless of the type of strain gage, typical pros and cons include:

- Cons
  - Signal conditioning hardware is expensive
  - Temperature affects readings
  - Bonding agent adds local rigidity to beam
  - Signals are susceptible to electro magnetic interference (EMI)
  - Strain beyond elastic region induces permanent damage
- Pros
  - Gages themselves are low cost (\$10+)
  - No feed-through from rigid motion
  - The measurements are proportional to the modal amplitudes and not their derivatives
  - Accurate measurements

Although strain gages have been proven to be an effective and accurate sensor for numerous structural vibrations applications, the list of cons is long. In particular, expensive signal conditioning is always required and environmental conditions need to be constantly taken into account.

4.2.3 Accelerometers

A good alternative to vision and strain gages are accelerometers that can often measure 3 directions in one small sensor. Micro-electromechanical systems (MEMs) accelerometer technology has advanced significantly in recent years and are included in most small mobile devices, such as smartphones, tablets, smart-watches, etc. MEMs accelerometers work by measuring a capacitance change due to the motion of a small proof mass connected to the outside chassis through springs. Voltage measurement due to the change in the proof mass location is proportional to the acceleration. In addition, modern MEMs accelerometers often have on board processors that do signal processing.

For a simple beam, the acceleration of a point can be described as:

$$\ddot{x} = \ddot{w}(x,t) = \sum_{j=1}^{m} \ddot{\delta}_j(t) w(x), \tag{125}$$

where m is the number of significant modes, $\ddot{\delta}$, is the modal acceleration and $w(x)$ is the mode shape at the location where the acceleration is measured.

A list of common pros and cons for accelerometers includes:

- Cons
  - Direct feed-through from the input to the structure
  - Drift in low frequencies
- Pros
  - MEMS accelerometers are very low cost
  - Sensors are durable to shock and environment changes
  - MEMS accelerometers do not generally require extra signal conditioning
  - Digital communication protocols can be used to shield data from EMI

Accelerometers were chosen to be used on this test bed, as it is a sensor, that is easy to incorporate into a control system and its performance does not suffer from environmental effects. Additionally, accelerometers are much easier to physically mount on a surface compared to strain gages.

Figure 37: Multifunction Input/Output device PCIe-6363

Figure 38: 12A8 Motor Driver

4.3 System Architecture

The controller for the system is a standard desktop PC running a custom version of Real-Time Linux by National Instruments. The processor of the computer is a Intel i7-2600K and it has 4 GB of RAM. The multifunction input/output device inserted into a PCI-e slot on the computer's motherboard is a PCIe-6363, shown in Figure 37[1] by National Instruments. The card has 32 Analog Inputs, 4 Analog Outputs, 48 multifunction digital inputs/outputs, including 4 counters.

The 12A8 motor drivers, shown in Figure 38[2], from Advanced-Motion-Control were used. These drives can operate in either velocity or torque mode. A PWM signal is sent to the motors. The control input and the current monitoring output use analog signals to

[1]Image from `http://www.ni.com/en-us/support/model.pcie-6363.html`
[2]Image from `https://www.a-m-c.com/products/?page=product&cid=root&id=12A8`

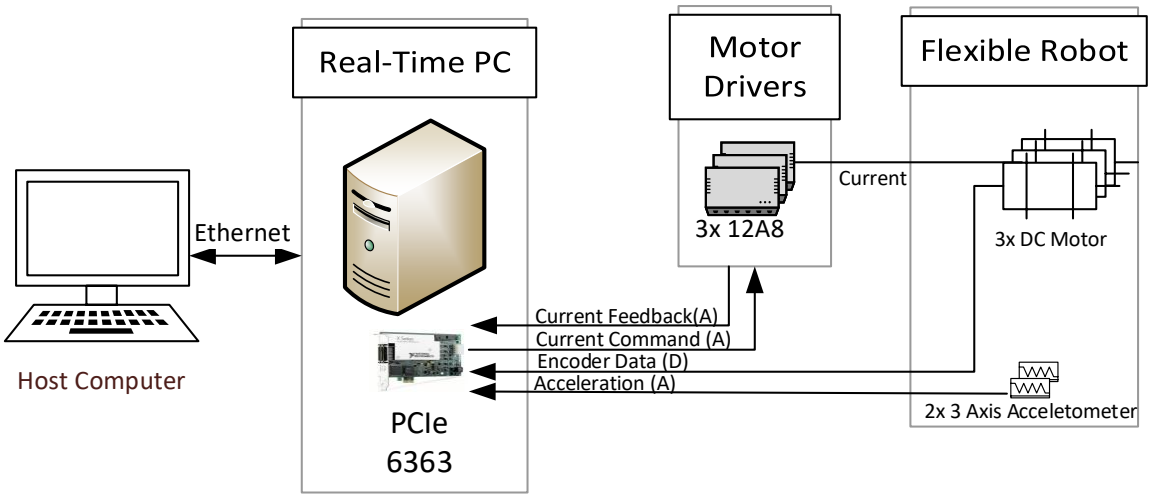


Figure 39: Diagram of the System

communicate. The maximum and continuous current limits of the drives are beyond what the selected motors can draw.

The full system schematic can be seen in Figure 39. In the diagram (A) represents an analog signal and (D) a digital signal.

4.4 System Natural Frequencies

The first four modeled natural frequencies of the test bed are shown in Figure 40. It can be seen that in the workspace of q_3 (180-360°), there is significant changes in modes 3 and 4 and slight change in the frequencies of modes 1 and 2. Figure 41 shows the first four mode shapes at configuration $q_3 = 270°$. Note that mode 1, shown in Figure 41a, and mode 4, shown in Figure 41d, are very similar to the first two mode shapes of a beam if flexure is only considered in the x_3 and x_4 direction. Similarly, mode 2, shown in Figure 41b, and mode 3, shown in Figure 41c, are the first two mode shapes for a beam in the y_3 and y_4 directions.

Figure 42 shows the mode shapes when the angle q_3 is set to 350°. The reason why the natural frequencies of mode 1, whos shape is seen in Figure 42a, and mode 2, whos shape is seen in Figure 42b, increase slightly is because the "moment arm" to the base of the tip

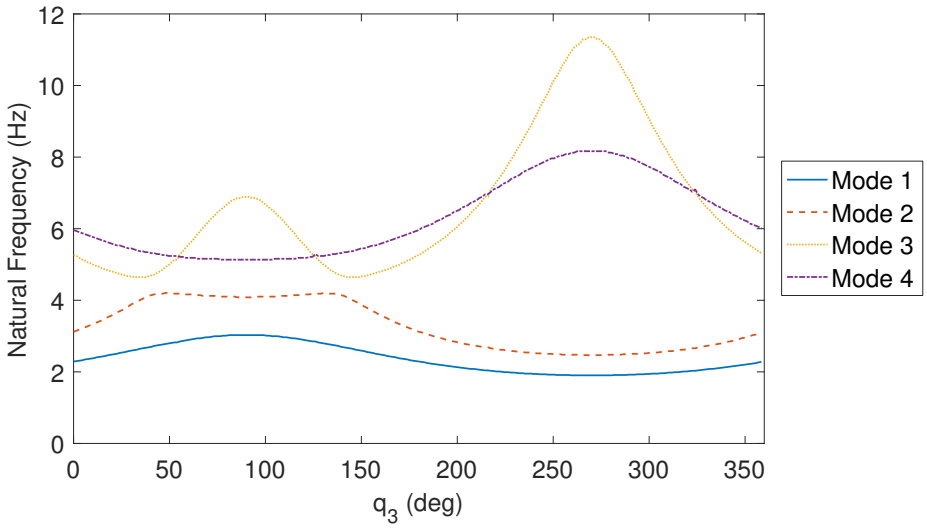
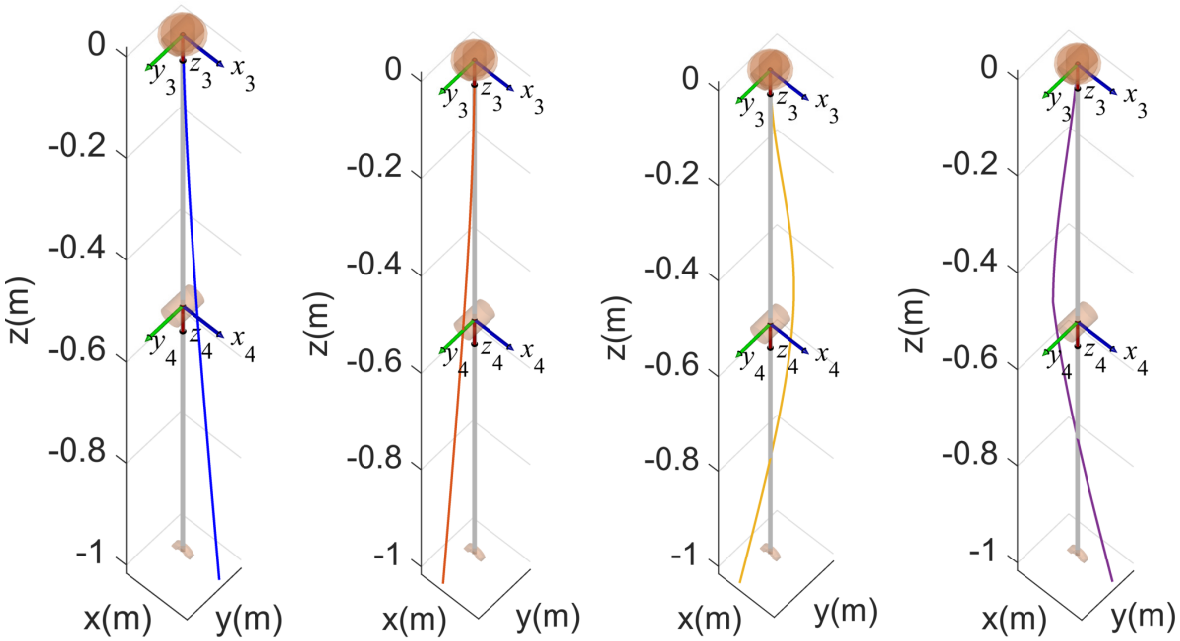


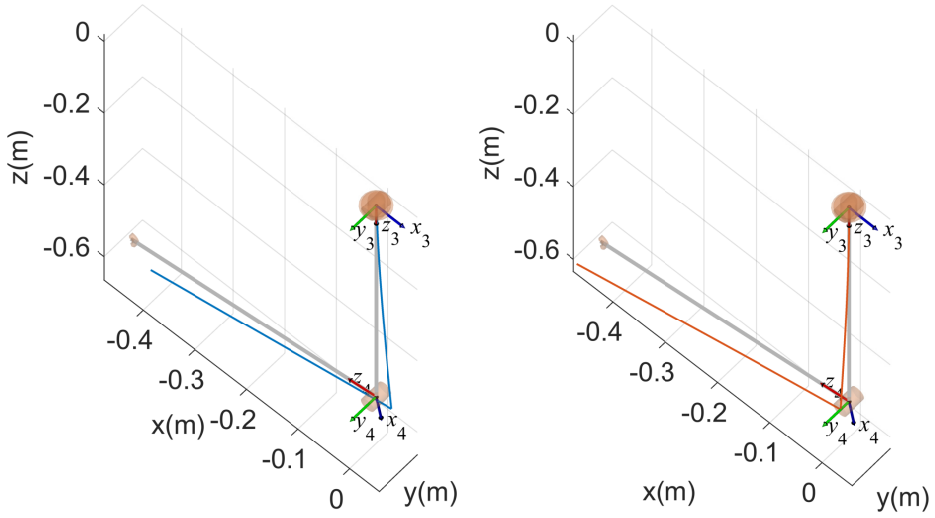
Figure 40: Modeled Natural Frequencies of the Test Bed



(a) Mode 1 (b) Mode 2 (c) Mode 3 (d) Mode 4

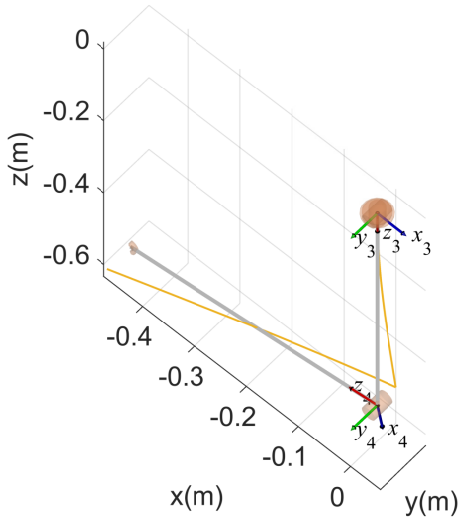
Figure 41: First Four Modes at $q_3 = 270°$

mass and link 2 is lower in configuration $q_3 = 350°$ compared to $q_3 = 270°$. The physical reason why the natural frequency form mode 3 drops from 11.6 Hz at $q_3 = 270°$ to 5.6 Hz at $q_3 = 350°$ is evident by comparing Figures 41c and 42c. In Figure 42c, the robot effectively has much more "torsional" inertia.

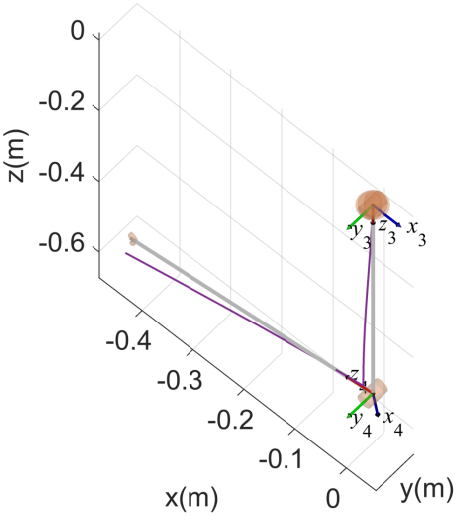


(a) Mode 1

(b) Mode 2



(c) Mode 3



(d) Mode 4

Figure 42: First Four Modes at $q_3 = 350°$

4.5 Test Bed Verification

If this test bed is to be confidently used for verification, its characteristics need to align with the model derived in Chapter 3. First, the natural frequencies in the robots work space are evaluated. Second, the response to motion is compared to the responses obtained from the model.

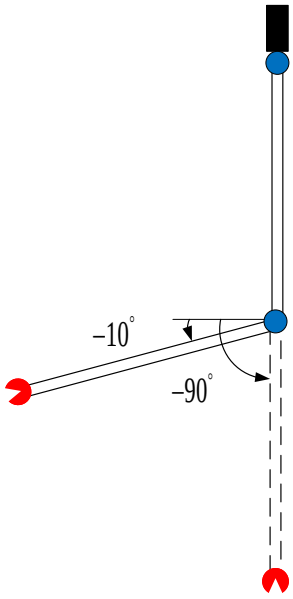


Figure 43: Limiting Configurations for Impulse Tests

4.5.1 Natural Frequency Tests

Impulse tests were conducted on the test bed by varying angle q_3 by 10° increments from 350°(=-10°) to 270°(=-90°). Figure 43 shows the test bed at those limiting angles. Only a half of the work space for q_3 is tested since the other half is symmetric. The three joints of the robot were controlled by P controllers with gains of 120, 100, 90 Nm/rad for joints q_1, q_2, and q_3, respectively. These values were added into the model as joint stiffnesses. Data was gathered from two three-axis accelerometers at a data acquisition rate of 5000 Hz. Data points were gathered from the peaks of the Fast Fourier Transform (FFT) graphs calculated for each acceleration signal. No windowing nor averaging was used in the calculations. For example, Figure 44 shows the FFT of the accelerometer data at the end effector in the link x direction at q_3 = 300°. Four distinct peaks can be found at 2, 2.4, 8, and 9 Hz.

Figure 45 shows the results obtained from all of the impulse tests, marked by an asterisk, versus the modeled values. It can be seen that the experimentally obtained frequencies match the modeled values very well. These results further validate the model presented in Section 3.2 in the sense that the model is also valid for structures with hollow cross-sections.

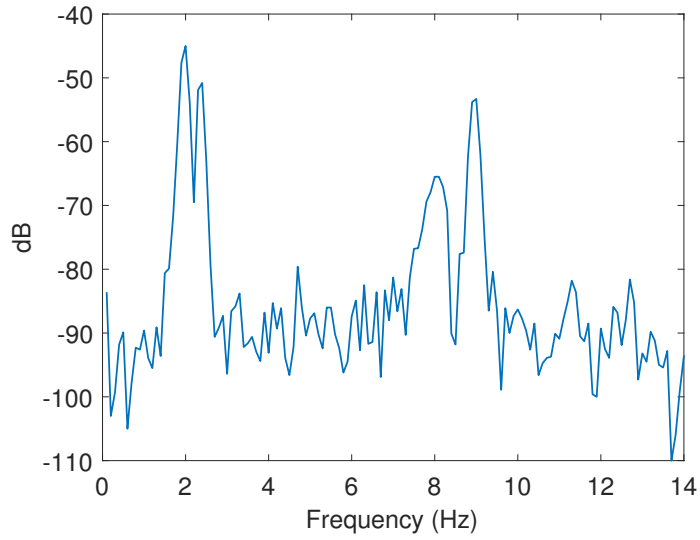


Figure 44: FFT of Accelerometer Signals on Link 2 in the Link x Direction at $q_3 = 300°$

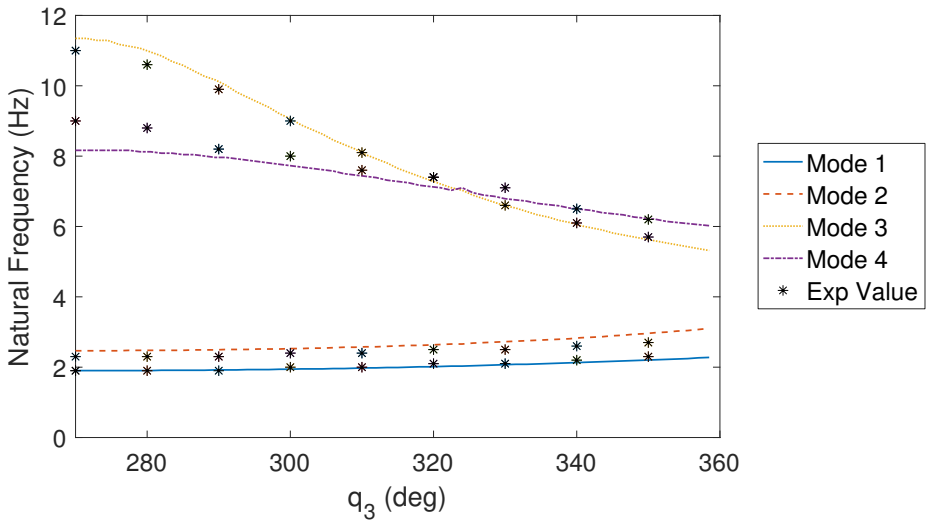


Figure 45: Experimental Data From Impulse Tests at Various Configurations

4.5.2 Motion Tests

In the previous section, the test bed was validated for only static configurations. While such tests are important, they are not sufficient to validate the test bed for all relevant motions. Results from 3 motions are presented in this section to illustrate the capabilities of the testbed. The control law used in all of the tests was a standard feed-forward torque type controller commonly used with serial robot arms:

$$\tau = M_{qq}(q)\ddot{q}_d + K_d\dot{e} + K_p e + g(q), \tag{126}$$

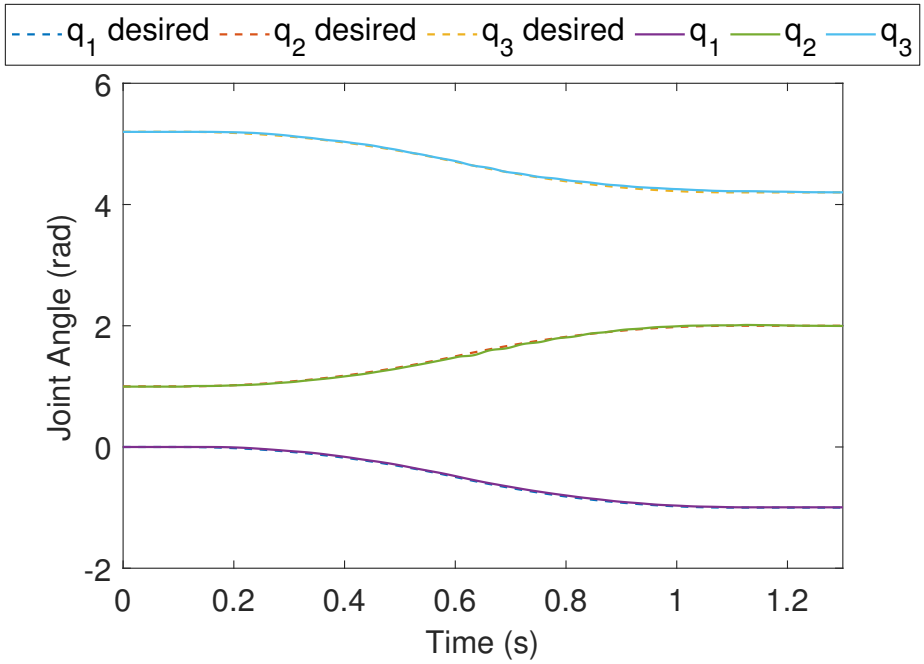


Figure 46: Desired and Measured Joint Angles For Move 1

where M_{qq} is the upper left inertia sub matrix and $g(q)$ is the gravity matrix from Section 3.3; $e = q_d - q$ is the error between the desired joint angle and the actual joint angle; K_d is the derivative gain, and K_p is the proportional gain. $K_d = diag(20, 20, 10)$ and $K_p = diag(100, 90, 60)$ were used. The effects of gravity were subtracted from the accelerometer reading, and a high-pass filter with a cut-off frequency of 0.2 Hz was used to filter out the errors due to sensor alignment.

Move 1 is performed by giving all joints are given a bang-bang acceleration that results in a change of 1 rad for each joint. The duration of the motion is 1 s. Joint q_1 moves from 0 to -1 rad, joint q_2 moves from 1 to 2 rad, and joint q_3 moves from 5.2 to 4.2 rad. Figure 46 shows the desired joint values versus the measured angles during the transient stage. It can be seen that the motor controller is able to track the motion relatively well, with minor differences between the signals.

Figure 47 shows the acceleration signals gathered from the accelerometer on the tip of the robot arm versus the accelerations obtained from simulating the motion using the recorded joint position data. The accelerations are in reference to the inertial frame and

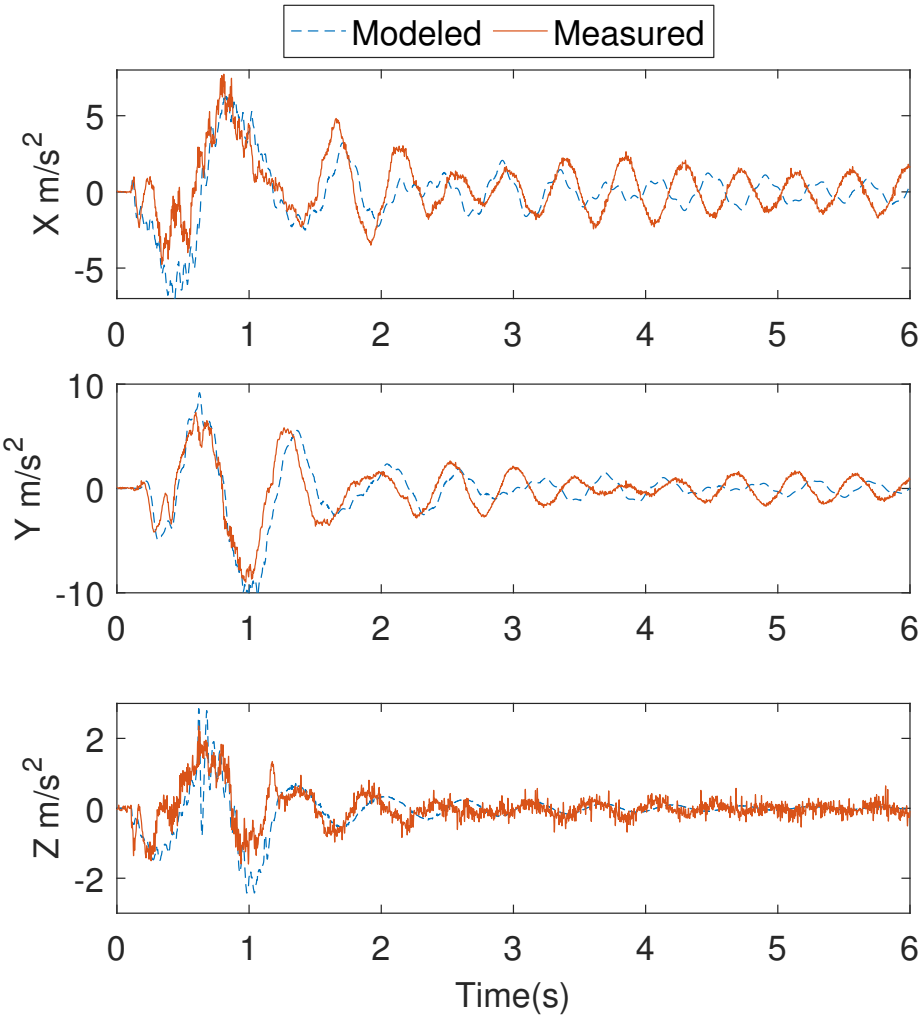


Figure 47: Accelerometer Reading vs Modeled Acceleration For Move 1

will be discussed in detail in Section 5.4.2. It can be seen that the simulation results match the experimental results relatively well. In all directions, the magnitude is very similar. However, there are slight differences in phase. The reasons for disparities include imperfect clamping of the links, non-linear spring effects in the belt and, motor shafts slipping in the pullies.

Move 2 is a planar move where joint $q1$ remains stationary, joint q_2 moves from 1 to 2 rad, and joint q_3 moves from 6.28 to 5.28 rad. Figure 48 shows the accelerometer data obtained during and after the motion versus accelerations from simulating the motion using the recorded joint motion. It can be seen that data in the X direction matches relatively well

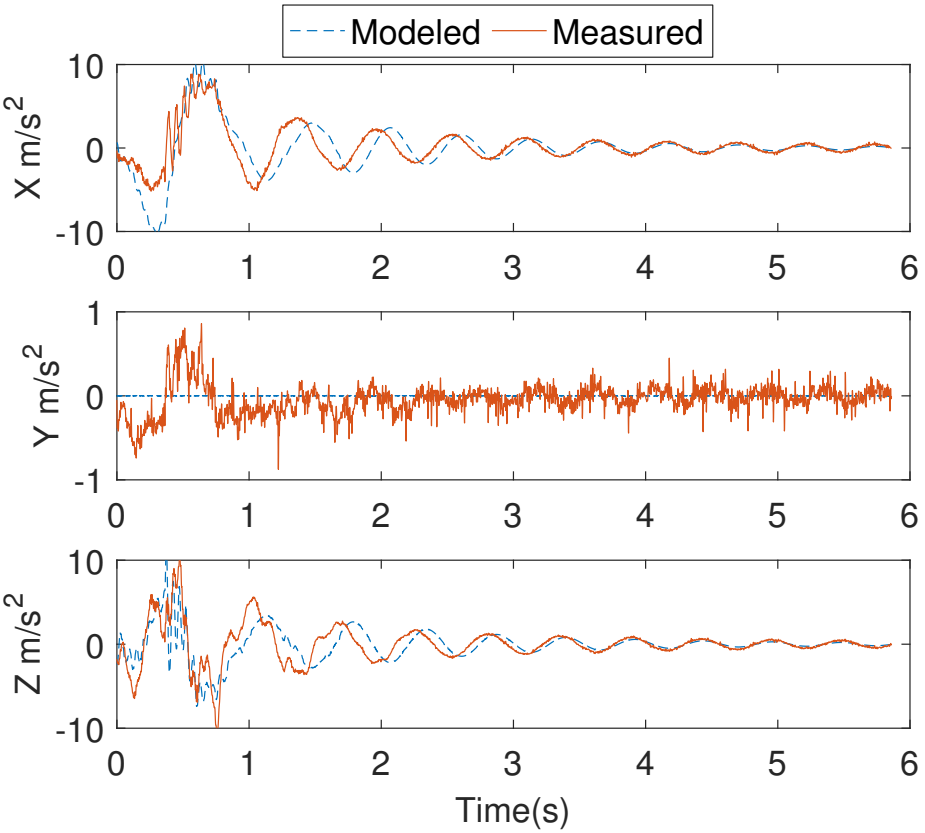


Figure 48: Accelerometer Reading vs Modeled Acceleration For Move 2

in magnitude, but has a slight delay in phase. Since this move is purely planar, the simulated Y direction acceleration is 0 throughout the motion. In the experiment, the accelerometer recorded non zero signals that had a relatively small amplitude. Y axis measurements also illustrate the high-pass filter working as the signal trends towards zero mean after 1 second of motion. Acceleration in the Z direction matches the simulated one with the same characteristics as the X axis data.

During move 3, joints q_2 and q_3 are held stationary and joint q_1 moves from 0 to -1 *rad*. Figure 49 shows the accelerometer signals from the the experiment versus the modeled values. It can be seen that the results do not match as well compared to moves 1 and 2. In the simulation, an extra mode was excited that was not consistent with the experiment. The magnitude of the vibration was similar between the simulation and experiment. High stress on the components might have caused improper clamping conditions to occur.

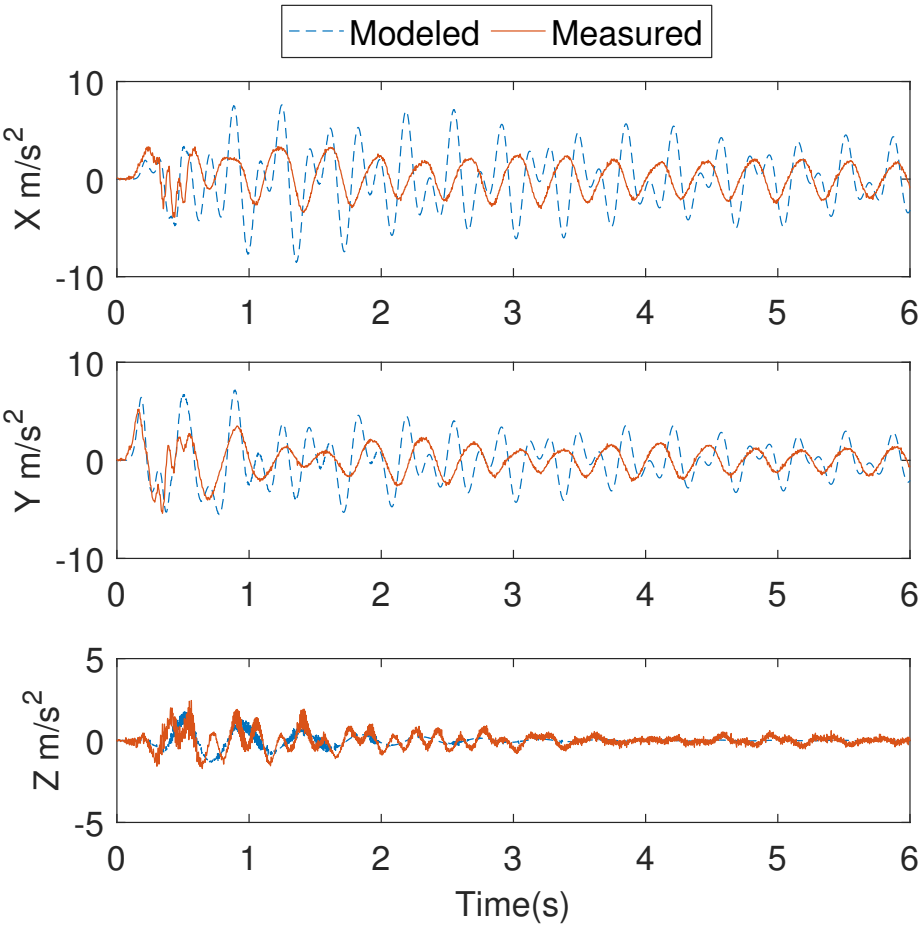


Figure 49: Accelerometer Reading vs Modeled Acceleration For Move 3

The test bed does not match the model presented in the previous Chapter perfectly. However, the experiments show that the data obtained from the test-bed does show similar characteristics compared to the model can be confidently used for hardware validation of estimation and control.

Appendix F contains additional figures showing all joint motions for all moves.

4.6 Chapter Summary

In this chapter the development of a 3 joint, 2 flexible link test bed was described. The test-bed illustrates how the natural frequencies of the system are configuration dependent and match the theory presented in Chapter 3. The data obtained during motions of the manipulator did not match the model presented in the previous chapter perfectly; however, it was close enough to use for estimation and control. In the following chapters, the test bed

will be used to test estimation and controller algorithms.

Chapter V

ESTIMATION FOR FLEXIBLE MANIPULATOR ARMS

Even though open-loop controllers can theoretically move a robot without any significant oscillations, they cannot handle disturbances caused by the environment or in the presence of significant model error. It is, therefore, vital that the flexible states are known for a closed-loop controller to ensure that oscillations are damped effectively. This chapter presents an estimator for the flexible states based on Kalman filtering techniques. Acceleration and strain signals are used for reconstructing the states.

5.1 Measuring Flexible State Variables

Taking advantage of the fact that deflection can be separated into a summation of time dependent variables (modal amplitudes) multiplied by space dependent variable (mode shapes), and assuming that the mode shapes of the entire structure are known in advance, the estimation problem comes down to finding the time-dependent variables.

Assume that at point z_s link deflection $w(z_s, t)$ is measurable. The deflection is therefore described as:

$$w(z_s, t) = \sum_{i=1}^{\infty} \psi_i(z_s)\delta_i(t), \tag{127}$$

where δ_i is the i^{th} modal amplitude and $\psi_i(z_s)$ is the modal deflection at point z_s. The significance of higher modes in the response becomes insignificant and a truncated model of n finite modes is appropriate. The flexure at point z_s becomes

$$w(z_s, t) \approx [\psi_1(z_s)\psi_2(z_s)...\psi_n(z_s)] \begin{bmatrix} \delta_1(t) \\ \delta_2(t) \\ \vdots \\ \delta_n(t) \end{bmatrix} \tag{128}$$

If p ($p > n$) measurements at locations z_i ($i = 1, 2, .., p$) are available, then the modal

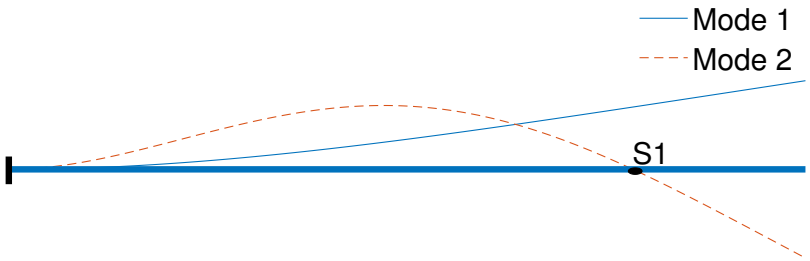


Figure 50: First and Second Mode of a Fixed-Free Beam

amplitudes can be approximated in the least squares sense:

$$\begin{bmatrix} \delta_1(t) \\ \delta_2(t) \\ \vdots \\ \delta_n(t) \end{bmatrix} \approx \left[\Psi^T \Psi\right]^{-1} \Psi^T \begin{bmatrix} w(z_1,t) \\ w(z_2,t) \\ \vdots \\ w(z_p,t) \end{bmatrix}, \tag{129}$$

where

$$\Psi = \begin{bmatrix} \psi_1(z_1) & \psi_2(z_1) & \dots & \psi_n(z_1) \\ \vdots & \vdots & \vdots & \vdots \\ \psi_1(z_p) & \psi_2(z_p) & \dots & \psi_n(z_p) \end{bmatrix} \tag{130}$$

This approach can be extended to 3D flexure estimation. Therefore using the least squares estimation for flexible coordinates requires at least one sensor per unknown state variable. Generally the state variables also contain the derivatives of the modal amplitudes $\dot{\delta}$ and $2n$ sensors are required. The modal velocities can be calculated by taking derivatives of the modal amplitudes, however, this approach is not advisable if significant noise is expected to be in the measurements. One additional requirement is that the sensors cannot lie on nodal points since they provide no information about specific modes. If the measurements were to be taken close to the nodal points they would induce high gains for the corresponding coordinates. This could lead to noisy or even unstable estimations. An example of this situation is illustrated in Figure 50 where the first two modes for a fixed-free beam are shown. A sensor on point $S1$ will not have any information about mode 2 since it is the nodal point for that mode. Hence the least square estimation is not a good strategy for finding flexible state variables.

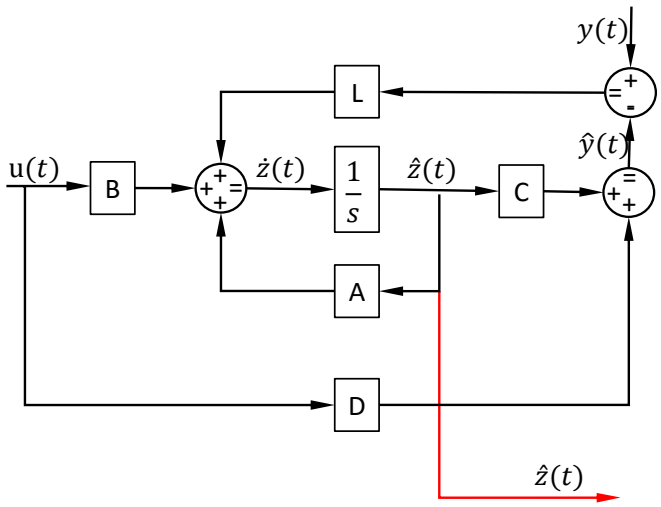


Figure 51: Observer Schematic

To estimate the full state of the system with a limited set of sensors, a state observer must be utilized. An observer uses a model of the system and the available sensing data to estimate the states that are not directly measurable.

An illustration of a state observer for a linear system is shown in Figure 51. The observer is a dynamic system that has a loop gain L that drives the predicted state estimates $\hat{z}$ to the true values z by acting on the error between the measured outputs and the predicted outputs $(y(t) - \hat{y}(t))$. The convergence properties and robustness to measurement noise are altered by the gain L. Care must be taken when choosing the controller gain, as a high gain introduces noise and a low gain might not make the system converge fast enough. For a single link manipulator an observer like this could be utilized, however, multi link manipulators are non-linear and therefore nonlinear observers must be used.

5.2 Non-linear Observation

Although the formulation for a non-linear observer is much harder than for linear systems, analogous methodologies apply. Consider a non-linear system given by

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= h(x(t), u(t)), \end{aligned} \tag{131}$$

where x represents state variables, y is the output of the system, u is the input to the system, and f and h are nonlinear vector functions. Similarly to linear observers the a non-linear observer can be defined by:

$$\begin{aligned} \dot{\hat{x}} &= f(\hat{x}(t), u(t)) + K(y(t) - \hat{y}(t)) \\ y(t) &= h(\hat{x}(t), u(t)), \end{aligned} \tag{132}$$

where $\hat{()}$ represents the estimated states, the observer gain $K = K(x, u)$ is chosen that the observation error $e(t) = x(t) \backsimeq \hat{x}(t)$ tends to zero. The observation error dynamics are therefore given by:

$$\dot{e} = \dot{x} - \dot{\hat{x}} = f(x, u) - f(\hat{x}, u) - K(g(x, u) - h(\hat{x}, u)) \tag{133}$$

Substituting $\hat{x} = x - e$ into 133 we obtain

$$\dot{e} = f(x, u) - f(x - e, u) - K(h(x, u) - h(x - e, u)) = \chi(x, u, e, K) \tag{134}$$

at steady state ($\dot{e} = 0$) it can be seen that $e = 0$ is a solution for equation (134). The observer gain K must therefore be chosen such that the error dynamics are asymptotically stable (force the error at steady state to zero). The stability of the observer can be examined by the first method of Lyapunov. The Jacobian for the error equation (134) is given by

$$A = \frac{\partial f(x - e, u)}{\partial e} - K(x, u)\frac{\partial g(x - e, u)}{\partial (x - e)} \tag{135}$$

By the first stability method of Lyapunov, the Jacobian matrix A must have negative real parts for all its eigenvalues for $x \epsilon X$ and $u \epsilon U$, where X and U are sets of admissible state and control variables.

5.2.1 Observability

A system is said to be observable if the current state can be reconstructed from observing its input-output behavior. Without the loss of generally consider an autonomous system (136).

$$\begin{aligned} \dot{x} &= f(x) \\ y &= h(x) \end{aligned} \tag{136}$$

To determine the initial condition x_0 from measurement $y(t)$, $0 \leq t \leq T$ derivatives of y must be taken:

$$\begin{aligned} y &= h(x) \\ \dot{y} = h_x(x)\dot{x} &= h_x(x)f(x) = h^{(1)}(x) \\ \ddot{y} = h_x^{(1)}(x)\dot{x} &= h^{(2)}(x), \\ &\vdots \\ y^{(m)} &= h^{(m)}(x) \end{aligned} \tag{137}$$

where the subscript x represents the partial derivative with respect to x $\frac{\partial}{\partial x}$. If system (137) is solvable then it implies observability. Alternatively system (137) can be written in terms of Lie derivatives

$$h_x(x)f(x) = L_f h(x), \tag{138}$$

where $L_f h(x)$ is the Lie derivative of h along the vector field f. The higher order derivatives in the Lie formulation can be given by

$$L_f^j h(x) = \frac{\partial L_f^{j-1} h(x)}{\partial x} f(x) \tag{139}$$

Therefore system (137) can be written as:

$$O(x) = \begin{bmatrix} \frac{\partial L_f^0 h(x)}{\partial x} \\ \vdots \\ \frac{\partial L_f^{n-1} h(x)}{\partial x} \end{bmatrix} \tag{140}$$

System 136 is therefore observable if $O(x)$ has full rank n. A more rigorous explanation of nonlinear observability can be found in [66].

For linear systems

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{141}$$

(140) can be written in the form:

$$O = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{142}$$

For linear time varying systems

$$\begin{aligned} \dot{x} &= A(t)x + B(t)u \\ y &= C(t) \end{aligned} \tag{143}$$

(140) is represented by

$$O = \begin{bmatrix} C \\ \frac{\partial C}{\partial t} + CA \\ \vdots \end{bmatrix} \tag{144}$$

5.3 Flexible Manipulator Arm Kinematics

Before an observer can be designed the kinematics of a flexible arm must be discussed. A point on link i can be related to the coordinate system fixed to link $i-1$ by a transfer matrix A_i. Therefore every point s on link i can be represented in the inertial frame 0 by the relation

$${}^0r_i(s) = {}^0T_i{}^ir_i(s) \tag{145}$$

where

$$\begin{aligned} {}^jT_i &= \prod_{m=j+1}^{i} A_m(q(t)) \\ {}^0T_0 &= \mathbb{I}, \end{aligned} \tag{146}$$

where ${}^ir_i$ is the position vector of the rigid arm in the local i coordinate frame. A_i is the transfer matrix that makes sure that the neutral axis always matches the z_i direction of the neutral axis of the link i. Refer to Chapter 3 for the full nomenclature. Figure 52 shows the relationships between the coordinate frames. Matrix $T_i = {}^0T_i$ is always in the form of $\begin{bmatrix} R_{i_{3\times3}} & p_{i_{3\times1}} \\ 0_{1\times3} & 1 \end{bmatrix}$, where R_i is the rotation of frame i with respect to the inertial frame and p_i is the position vector of the origin of the i-th frame with respect to the inertial origin

Furthermore point s on link i with respect to the frame i can be represented by a finite sum of separable mode shapes which are products of time varying amplitudes and

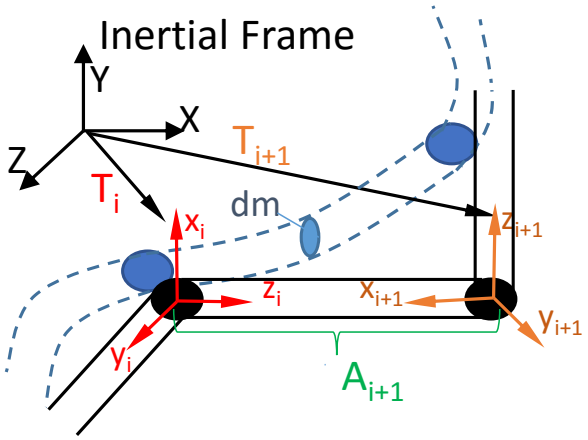


Figure 52: Frame Transformations

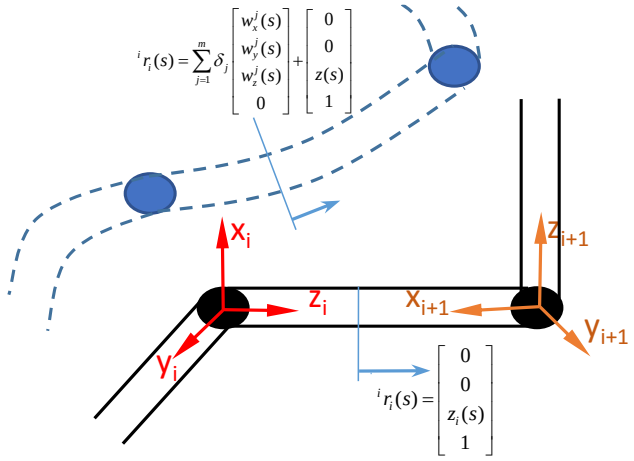


Figure 53: Coordinates on Link i

kinematically admissible mode shapes

$$ {}^{i}r_i(s) = \sum_{j=1}^{m} \delta_j \begin{bmatrix} w^i_{j_x}(s) \\ w^i_{j_y}(s) \\ w^i_{j_z}(s) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ s \\ 1 \end{bmatrix}, \tag{147} $$

where $\delta_j(t)$ is the mode shape amplitude for the j^{th} mode; $w^i_{j_x}$, $w^i_{j_y}$, $w^i_{j_z}$ are the deflections of mode j on link i in the x_i, y_i and z_i directions; m is the number of modes considered. This is illustrated in Figure 53.

Therefore the acceleration of a point s on link i can be represented in frame 0 by taking

two time derivatives of (145) yielding

$$
{}^0\ddot{r}_i(s) = \ddot{T}_i(q,\dot{q},\ddot{q})^i r_i(s) + 2\dot{T}_i(q,\dot{q})^i \dot{r}_i(s) + T(q)^i \ddot{r}_i(s) =
$$

$$
= \ddot{T}_i(q,\dot{q},\ddot{q}) \left(\sum_{j=1}^{m} \delta_j \begin{bmatrix} w_{j_x}^i(s) \\ w_{j_y}^i(s) \\ w_{j_z}^i(s) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ s \\ 1 \end{bmatrix} \right) + 2\dot{T}_i(q,\dot{q}) \left(\sum_{j=1}^{m} \dot{\delta}_j \begin{bmatrix} w_{j_x}^i(s) \\ w_{j_y}^i(s) \\ w_{j_z}^i(s) \\ 0 \end{bmatrix} \right) + T_i(q) \left(\sum_{j=1}^{m} \ddot{\delta}_j \begin{bmatrix} w_{j_x}^i(s) \\ w_{j_y}^i(s) \\ w_{j_z}^i(s) \\ 0 \end{bmatrix} \right) \tag{148}
$$

The $\dot{T}$ and $\ddot{T}$ terms can be found by differentiating (146)

$$
\frac{dA_i}{dt} = \frac{\partial A_i}{\partial q_i}\dot{q}_i = U_i \dot{q}_i, \tag{149}
$$

$$
{}^j\dot{T}_i = \sum_{h=j+1}^{i} {}^jT_{h-1} U_h {}^hT_i \dot{q}_h, \tag{150}
$$

$$
\begin{aligned}
{}^0\ddot{T}_i = \sum_{h=1}^{i} ({}^0\dot{T}_{h-1} U_h {}^hT_i \dot{q}_h + {}^0T_{h-1}(\dot{U}_h {}^hT_i \dot{q}_h + U_h({}^h\dot{T}_i \dot{q}_h + {}^hT_i \ddot{q}_h))) = \\
\sum_{h=1}^{i} (\sum_{k=1}^{h-1} ({}^0T_{k-1} U_k {}^kT_{h-1} \dot{q}_k) U_h {}^hT_i \dot{q}_h + {}^0T_{h-1}(V_h {}^hT_i \dot{q}_h^2 + \\
+ U_h \sum_{k=h+1}^{i} ({}^hT_{k-1} U_k {}^kT_i \dot{q}_k)\dot{q}_h + {}^hT_i \ddot{q}_h))),
\end{aligned} \tag{151}
$$

where

$$
\dot{U}_i = \frac{\partial U_i}{\partial q_i}\dot{q}_i = V_i \dot{q}_i \tag{152}
$$

${}^0\ddot{r}_i(s)$ represents the acceleration in the inertial frame 0. However, an accelerometer on link i measures acceleration in reference to the local frame i. Therefore, the following transformation must be used to relate the acceleration from the accelerometers to the inertial frame:

$$
{}^0\ddot{r}_i(s) = {}^0T_i \ddot{r}_l, \tag{153}
$$

where ${}^0T_i$ is the homogeneous transfer matrix from the inertial frame to frame i, and $\ddot{r}_l = [\ddot{x}_l, \ddot{y}_l, \ddot{z}_l, 0]^T$ is the vector of accelerations in the local link frame i. Additionally, if the accelerometers used are MEMs type then gravity must also be subtracted from (148).

5.4 Estimation of Flexible States

To estimate states of a system

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x) \end{aligned} \tag{154}$$

the plant $f()$ and the measurements $h()$ must be defined before estimator algorithms can be presented.

5.4.1 Model of the Plant for Flexible State Estimation

As described in Chapter 3 the dynamics of a flexible robot can be represented by the following equation:

$$\begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & \mathbb{I} \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} C_{qq}(q,\dot{q},\dot{\delta}) & C_{q\delta}(q,\dot{q},\dot{\delta}) \\ C_{\delta q}(q,\dot{q},\dot{\delta}) & C_{\delta\delta}(q,\dot{q},\dot{\delta}) \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} g_r \\ g_f \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & K \end{pmatrix} \begin{pmatrix} q \\ \delta \end{pmatrix} = \begin{pmatrix} \tau \\ 0 \end{pmatrix}, \tag{155}$$

where $(q^T, \delta^T)^T$ is the n-vector of generalized joint variables q , and m- vector of flexible deformation variables δ, M is the $(n+m) \times (n+m)$ generalized mass matrix, $\mathbb{I}$ is the $m \times m$ identity matrix, C is the $n+m$ vector matrix containing the Coriolis and centrifugal terms and the terms accounting for interaction of joint variables and their rates and flexible variables and their rates, g is the $n+m$ vector of gravitational terms, K is the $m \times m$ flexural stiffness matrix, τ is the vector of input torques. In state space form of system (155) can be represented as:

$$\dot{x} = f(x,\tau), \tag{156}$$

where x is the $2(n+m)$ vector joint positions q, joint velocities $\dot{q}$, modal displacements δ, and modal velocities $\dot{\delta}$. Generally the joint position, q and velocities, $\dot{q}$, can be directly measured from encoders and tachometers. Therefore it is not necessary to construct an observer that includes those terms since computational complexity increases with a larger number of states. The flexible state variables δ and $\dot{\delta}$ cannot be directly measured and observers must be used. Therefore a reduced order observer should be constructed.

From equation (155) the bottom row can be written out to give the differential equation

governing the flexible coordinates:

$$\ddot{\delta} = -C_{\delta\delta}(q,\dot{\delta})\dot{\delta} - K(q)\delta - M_{q\delta}(q)\ddot{q} - C_{\delta q}(q,\dot{\delta})\dot{q} - g_{\delta}(q) \tag{157}$$

This equation can be converted to the following state space form:

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} \dot{\delta} \\ -K(q)\delta - C_{\delta\delta}(q,\dot{\delta})\dot{\delta} - C_{\delta q}(q,\dot{\delta})\dot{q} - u \end{bmatrix} \tag{158}$$

where $u = -M_{\delta q}(q)\ddot{q} - g_{\delta}(q)$. u includes all the terms that do not have the flexible states in them directly and are therefore considered as a known input to the system. Equation (158) is non-linear but can be linearized around the operating point:

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & \mathbb{I} \\ -K(q) & -\tilde{C}_{\delta\delta}(q,\dot{\delta}) - \tilde{C}_{\delta q}(q,\dot{\delta})\dot{q} \end{bmatrix}_{|\dot{\delta}=\dot{\delta}_o} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbb{I} \end{bmatrix} u, \tag{159}$$

where $\dot{\delta}_o$ is the operating point at which (159) is linearized about;

$$\tilde{C}_{\delta\delta}(q,\dot{\delta}) = \frac{\partial C_{\delta\delta}(q,\dot{\delta})}{\partial \dot{\delta}}, \tag{160}$$

and

$$\tilde{C}_{\delta q}(q,\dot{\delta}) = \frac{\partial C_{\delta q}(q,\dot{\delta})}{\partial \dot{\delta}} \tag{161}$$

Note that (159) is not linear but is linear time-varying since the terms depend on joint angles q and $\dot{q}$. A short form of equation (159) can be represented by:

$$\dot{\Delta} = F(q,\dot{q})\Delta + Bu(q,\dot{q}), \tag{162}$$

where $\Delta = \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix}$

5.4.2 The Measurement Model for Acceleration Feedback

Equation (148) yields the acceleration that a 3 three axis accelerometer at point p on link i would measure. Since ${}^0\ddot{r}_i(p)$ includes a term that is purely from the rigid motion serial

arm, the adjusted acceleration ${}^0\tilde{\ddot{r}}_i(p)$ that excludes that term is defined:

$$
\begin{aligned}
{}^0\tilde{\ddot{r}}_i(p) &= {}^0\ddot{r}_i(p) - \ddot{T}_i(q,\dot{q},\ddot{q}) \begin{bmatrix} 0 \\ 0 \\ p \\ 1 \end{bmatrix} = \\
&= \ddot{T}_i(q,\dot{q},\ddot{q}) \left(\sum_{j=1}^{m} \delta_j \begin{bmatrix} w^i_{j_x}(p) \\ w^i_{j_y}(p) \\ w^i_{j_z}(p) \\ 0 \end{bmatrix} \right) + 2\dot{T}_i(q,\dot{q}) \left(\sum_{j=1}^{m} \dot{\delta}_j \begin{bmatrix} w^i_{j_x}(p) \\ w^i_{j_y}(p) \\ w^i_{j_z}(p) \\ 0 \end{bmatrix} \right) + T_i(q) \left(\sum_{j=1}^{m} \ddot{\delta}_j \begin{bmatrix} w^i_{j_x}(p) \\ w^i_{j_y}(p) \\ w^i_{j_z}(p) \\ 0 \end{bmatrix} \right)
\end{aligned}
\tag{163}
$$

Because the state vector of the system consists of only δ and $\dot{\delta}$, $\ddot{\delta}$ must be eliminated from the measurement equation (163). Therefore, substituting the bottom row of (158) into (163) the measurement from one 3 axis accelerometer can be written in terms of the state vector $\begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix}$:

$$
\begin{aligned}
{}^0\tilde{\ddot{r}}_i(p) = \left[\ddot{T}_i W_i + T_i W_i(-K) \quad 2\dot{T}_i W_i + T_i(-\tilde{C}_{\delta\delta} - \tilde{C}_{\delta q}\dot{q}) \right]_{\dot{\delta}=\dot{\delta}_o} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} \\
+ T_i W_i u
\end{aligned}
\tag{164}
$$

where

$$
W_i(p) = \begin{bmatrix} w^i_{1_x}(p) & w^i_{2_x}(p) & \dots & w^i_{m_x}(p) \\ w^i_{1_y}(p) & w^i_{2_y}(p) & \dots & w^i_{m_y}(p) \\ w^i_{1_z}(p) & w^i_{2_z}(p) & \dots & w^i_{m_z}(p) \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{165}
$$

W_i is the matrix containing the mode shapes at the location of the sensor.

Because ${}^0\ddot{r}_i(p) = \begin{bmatrix} {}^0\ddot{x}_i(p) \\ {}^0\ddot{y}_i(p) \\ {}^0\ddot{z}_i(p) \\ 0 \end{bmatrix}$, where $\ddot{x}$, $\ddot{y}$, $\ddot{z}$ are the accelerations in the fixed 0 frame, only the first three rows of equation (164) are useful. If the accelerometer has less than 3 measurement axis, only the directions that measurements must be kept. When additional

accelerometers are used (164) can easily be expanded to accommodate that. For example using one accelerometer on link i at point p_1 and the other accelerometer on link j at point p_2, extra lines are added to (164) as shown in:

$$\begin{bmatrix} {}^0\ddot{\tilde{r}}_i(p_1) \\ {}^0\ddot{\tilde{r}}_j(p_2) \end{bmatrix} = \begin{bmatrix} \ddot{T}_i W_i + T_i W_i(-K) & 2\dot{T}_i W_i + T_i(-\tilde{C}_{\delta\delta} - \tilde{C}_{\delta q}\dot{q}) \\ \ddot{T}_j W_j + T_j W_j(-K) & 2\dot{T}_j W_j + T_j(-\tilde{C}_{\delta\delta} - \tilde{C}_{\delta q}\dot{q}) \end{bmatrix}_{\dot{\delta}=\dot{\delta}_o} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} + \\ + \begin{bmatrix} T_i W_i u \\ T_j W_j u \end{bmatrix} \tag{166}$$

In compact form Equation (166) can be expressed as

$$y = H\Delta + Du \tag{167}$$

Equations (162) and (167) are therefore in the standard state space form that form the basis of the estimation algorithm.

The nonlinear system with N accelerometers is represented by:

$$\dot{x} = \begin{bmatrix} \dot{\delta} \\ -C_{\delta\delta}(q,\dot{q})\dot{\delta} - K(q)\delta - D(q)\dot{\delta} - M_{q\delta}(q)\ddot{q} - C_{\delta q}(q,\dot{q},\dot{\delta})\dot{q} - g_\delta(q) \end{bmatrix} = f(x,u) \tag{168}$$

$$y = \begin{bmatrix} \ddot{T}_1 W_1\delta + 2\dot{T}_1 W_1\dot{\delta} + T_1 W_1(-C_{\delta\delta}\dot{\delta} - K\delta - D\dot{\delta} - M_{q\delta}\ddot{q} - C_{\delta q}\dot{q} - g_\delta) + \ddot{T}_1 p_1 \\ \vdots \\ \ddot{T}_N W_N\delta + 2\dot{T}_N W_N\dot{\delta} + T_N W_N(-C_{\delta\delta}\dot{\delta} - K\delta - D\dot{\delta} - M_{q\delta}\ddot{q} - C_{\delta q}\dot{q} - g_\delta) + \ddot{T}_N p_N \end{bmatrix} = h(x,u) \tag{169}$$

5.4.3 The Measurement Model for Strain Feedback

Although the estimator used in this dissertation uses acceleration signal, the derivation for a measurement model, using the flexible arm dynamics given by (155), for strain gages is in order. This is important as strain gages are currently by far the most utilized sensor in vibration control as described in Section 2.2.1.

The strain of a point p on link i is represented by:

$$\epsilon^i(p) = \sum_{j=1}^{m} \delta_j \begin{bmatrix} a_x {w^i_{j_x}}''(p) \\ a_y {w^i_{j_y}}''(p), \end{bmatrix} \tag{170}$$

where $\epsilon^i(p)$ is the measured strain on link i at point p; a_x and a_y are the distances from the neutral axis of the beam in the x_i and y_i directions; ${w^i_{j_x}}''(p)$ and ${w^i_{j_y}}''(p)$ are the second spatial derivatives ($()'' \equiv \frac{\partial^2}{\partial z_i^2}$) of the j^{th} in the x and y axis respectively; and δ_j is the j^{th} modal amplitude. the strain in only the x and y configurations is considered as axial deformation for serial robots is insignificant.

Therefore, in the standard state-space formulation the measurement for one pair of strain gages represented by

$$y_1 = \begin{bmatrix} Ws_i(p) & 0_{2\times m} \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix}, \tag{171}$$

where,

$$Ws_i(p) = \begin{bmatrix} a_x \begin{bmatrix} {w^i_{1_x}}''(p) & \cdots & {w^i_{m_x}}''(p) \end{bmatrix} \\ a_y \begin{bmatrix} {w^i_{1_y}}''(p) & \cdots & {w^i_{m_y}}''(p) \end{bmatrix} \end{bmatrix} \tag{172}$$

is the matrix that holds all of the second derivatives of the mode shapes on location p on link i. If strain from only one direction is available the only one row in (171) is kept. Strain from multiple locations can be combined to the measurement vector by adding more rows:

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_r \end{bmatrix} = \begin{bmatrix} Ws_i(p_1) & 0_{2\times m} \\ \vdots & \vdots \\ Ws_j(p_r) & 0_{2\times m} \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} \tag{173}$$

Although it is obvious that the measurement model for stain given in (173) is much less computationally intensive than the one for accelerations (169), strain gages themselves might not be the best choice as the sensor for serial robot arms as was discussed in Section 4.2. The full state space formulation with strain gage measurements is given by:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} \dot{\delta} \\ -C_{\delta\delta}(q,\dot{q})\dot{\delta} - K(q)\delta - D(q)\dot{\delta} - M_{q\delta}(q)\ddot{q} - C_{\delta q}(q,\dot{q},\dot{\delta})\dot{q} - g_\delta(q) \end{bmatrix} = f(x,u) \\ y &= \begin{bmatrix} y_1 \\ \vdots \\ y_r \end{bmatrix} = \begin{bmatrix} Ws_i(p_1) & 0_{2\times m} \\ \vdots & \vdots \\ Ws_j(p_r) & 0_{2\times m} \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} = h(x) \end{aligned} \tag{174}$$

5.4.4 Extended Kalman Filter

The Kalman filter was originally derived for linear stochastic systems that can be represented as

$$\begin{aligned} \dot{x} &= Ax + Bu + w \\ y &= Cx + v \\ w &\ (0, Q) \\ v &\ (0, R), \end{aligned} \tag{175}$$

where w represents the model noise that is zero mean, Gaussian with a covariance of Q, v is the measurement noise that is zero mean, Gaussian and with a covariance of R. The most general formulation for a Kalman filter is given by [119]:

$$\begin{aligned} \hat{x}(0) &= E[(0)] \\ P(0) &= E[(x(0) - \hat{x}(0))(x(0) - \hat{x}(0))^T] \\ K &= PC^T R^{-1} \\ \dot{\hat{x}} &= A\hat{x} + Bu + K(y - C\hat{x}) \\ \dot{P} &= -PC^T R^{-1} CP + AP + PA^T + Q, \end{aligned} \tag{176}$$

where $\hat{x}$ is the estimation of state x, P is the estimation-error covariance, and K is the Kalman gain. Furthermore, $\hat{x}(0)$ is the initial guess for the estimator $E[]$ represents expected value, and $P(0)$ is the initial guess for the error covariance. By definition P must always be positive definite, real, symmetric. Numerically, however, this condition does not always hold and care must be taken to condition P. The equation for $\dot{P}$ is often referred to as the algebraic Riccati equation [119]. For a linear stochastic system the Kalman filter is an optimal filter with respect to virtually any criteria [92].

For a non-linear system model given by:

$$\begin{aligned} \dot{x} &= f(x, u, w, t) \\ y &= h(x, v, t) \\ w &\ (0, Q) \\ v &\ (0, R), \end{aligned} \tag{177}$$

where, $f()$ is the system equation and $h()$ is the measurement function, the Kalman filter can be applied to if the system can be linearized about a known trajectory (x_0, y_0, w_0, v_0) based on *a-priori* knowledge of the system behavior. The actual trajectories of the system will naturally differ from the nominal one due to disturbances and errors in the model. How the Extened Kalman Filter (EKF) differs from the linearized filter is that the nominal trajectory $x_0(t)$ is equal to the linearized Kalman filter estimate $\hat{x}(t)$. The linearized system around $\hat{x}$ system is given by:

$$\begin{aligned} A &= \frac{\partial f}{\partial x}_{|\hat{x}} \\ L &= \frac{\partial f}{\partial w}_{|\hat{x}} \\ H &= \frac{\partial h}{\partial x}_{|\hat{x}} \\ M &= \frac{\partial h}{\partial x}_{|\hat{x}} \\ \tilde{Q} &= LQL^T \\ \tilde{R} &= MRM^T \end{aligned} \tag{178}$$

The Extended Kalman Filter is realized by [119]:

$$\begin{aligned} \hat{x}_0 &= E[x(0)] \\ P(0) &= E[(x(0) - \hat{x}(0))(x(0) - \hat{x}(0))^T] \\ \dot{\hat{x}} &= f(\hat{x}, u, w_0, t) + K[y - h(\hat{x}, v_0, t)] \\ K &= PH^T\tilde{R}^{-1} \\ \dot{P} &= AP + PA^T + Q - PC\tilde{R}^{-1}CP \end{aligned} \tag{179}$$

Matrix P is the error covariance matrix which quantifies the uncertainty of the estimated states. If the nonlinearities are not too severe, P gives an idea on how accurate the estimates are. A large Q indicates less confidence in the accuracy of the model, and relatively more confidence for the measurements. In general, increasing Q increases the Kalman gain K which also changes the estimation $\hat{x}$ more aggressively. Similarly, a large R represents that the measurements are relatively less trustworthy and the gain K is lower, and therefore $\hat{x}$ changes less aggressively. Furthermore, P increases as Q and R increase, indicating lower

confidence for the state estimate $\hat{x}$. Although stability and convergence are readily available for the Kalamn filter, they are substantially much harder to find for nonlinear systems.

Although theoretically $P(t)$ should always be symmetric positive definite (because it is a covariance matrix), numerical errors due to the way processors handle numbers can make the matrix go non-symmetric or indefinite. Variable step ODE solvers will generally not have a problem keeping P properly conditioned, but problems can arise when fixed step solvers are used. In order to use and deploy continuous Kalman filters in real-time, fixed step solver are required. Typical recommendations [119] for improving the numerical stability include:

- Increase numerical precision of variables
- Use some form of Square root filtering
- Force P to be symmetric at each iteration $P = \frac{P+P^T}{2}$
- Initialize P so that no large changes in P happen
- Include fictitious process noise

5.4.4.1 Square Root Filtering

One of the numerous techniques used to make P more numerically stable is the square root filter. The idea is based on finding a matrix S such that:

$$P = SS^T, \tag{180}$$

where S is the matrix square root of P. Because P is symmetric positive definite an upper triangular matrix S satisfying $P = SS^T$ can always be found. Using the definition of S, it is substituted into the Riccati equation in (179):

$$\dot{S}S^T + S\dot{S}^T = ASS^T + SS^TA^T - SS^TC^tR^{-1}CSS^T + Q \tag{181}$$

Pre-multiplying (181) by S^{-1} and post-multiplying by S^{-T} yields:

$$S^{-1}\dot{P}S^{-T} = S^{-1}AS + S^TA^TS^{-T} - S^TC^TR^{-1}CS + S^{-1}QS^{-T} \tag{182}$$

Per [119] Mu, the upper triangular matrix of $S^{-1}\dot{P}S^{-T}$, allow finding $\dot{S}$:

$$\dot{S} = SM_u \tag{183}$$

Incorporating the square root filter into EKF is computationally more expensive than solving for the pure Riccati equation in (179), but is numerically more stable.

5.5 Case Study

The effectiveness of an EKF for flexible serial robot arms is presented in this section. Simulations are performed using both accelerometer feedback and strain feedback. The same two-link robot model that was used for simulation studies in Section 3.3.7 is used here, because it was validated using FEA analysis. The first three modes are used in the simulations since it was shown in the earlier section that higher modes have an insignificant effect on the behavior of the robot. The moves that were used in Section 3.3.7 are again summarized in Table 13.

Table 13: Joint Trajectories for Verification

	q_{1_0}	q_{1_f}	q_{2_0}	q_{2_f}	q_{3_0}	q_{3_f}
Move 1	0	-1	0	-1	0	-1
Move 2	0	0	0	-1	0	-1
Move 3	0	$-\pi$	0	0	0	0

The plant model does not change due to feedback type and is given by (158), and the linearized version is given in (159). The measurement noise characteristics can often be taken from sensor datasheets, the process model noise Q required by the Kalman filter, however, is almost always much harder to find [94]. Therefore Q was empirically chosen to be a good balance between being able to reduce the effect modeling errors and help recover the state with inaccurate measurements. The process noise covariance value of

$$Q = 5 \cdot 10^{-6}\mathbb{I} \tag{184}$$

was used for all simulation tests, where $\mathbb{I}$ is the 6×6 identity matrix. The dimension is due to three significant modes and their derivatives.

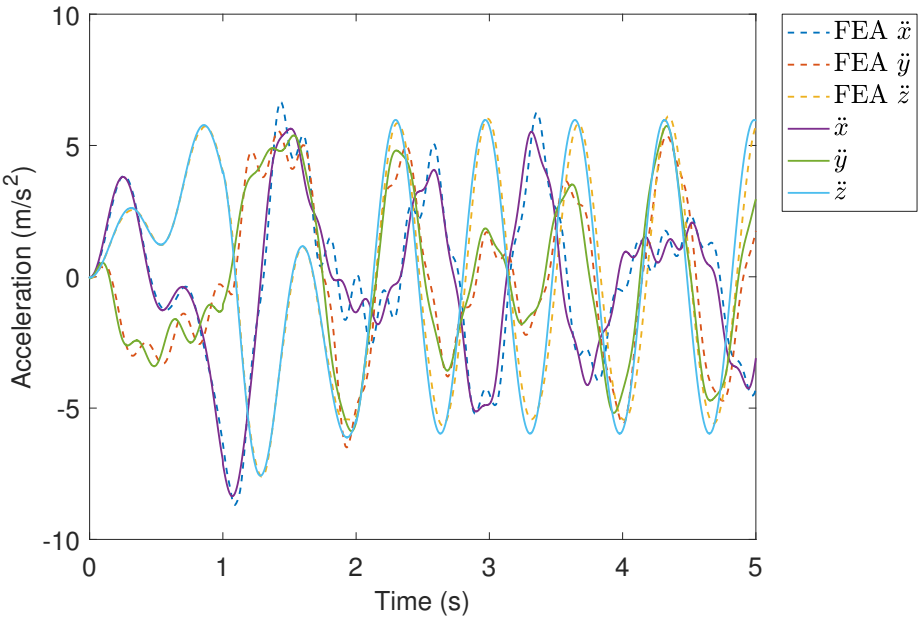


Figure 54: Comparison of Tip Acceleration Signals for Move 1 From The TMM Based Model and FEA Analysis

5.5.1 Acceleration Feedback

The nonlinear measurement equation is given in (169) and the linearized measurement equation in (168). The measurement comes from two three-axis accelerometers mounted at the end of each link. The noise characteristics were modeled after the sensors used on the testbed described in Chapter 4. Therefore the noise covariance matrix for two accelerometers used in EKF is

$$R = (250 \cdot 10^{-6} g)^2 \cdot \mathbb{I}, \tag{185}$$

where $g = 9.8\ m/s^2$, and $\mathbb{I}$ is the identity matrix of size $r \times r$, where r is the number of sensor signals. The band-limited white noise with the same covariance was added to the simulated accelerometer signals.

The accelerometer model, given in (148), was compared against the acceleration values obtained from FEA analysis performed with the commercial package COMSOL. Figure 54 shows the acceleration signals as if they were read by a three axis accelerometer at the tip during move 1. The dashed lines obtained from FEA match very closely to the ones obtained from (148). The signals in the x and z match very well, however there are small

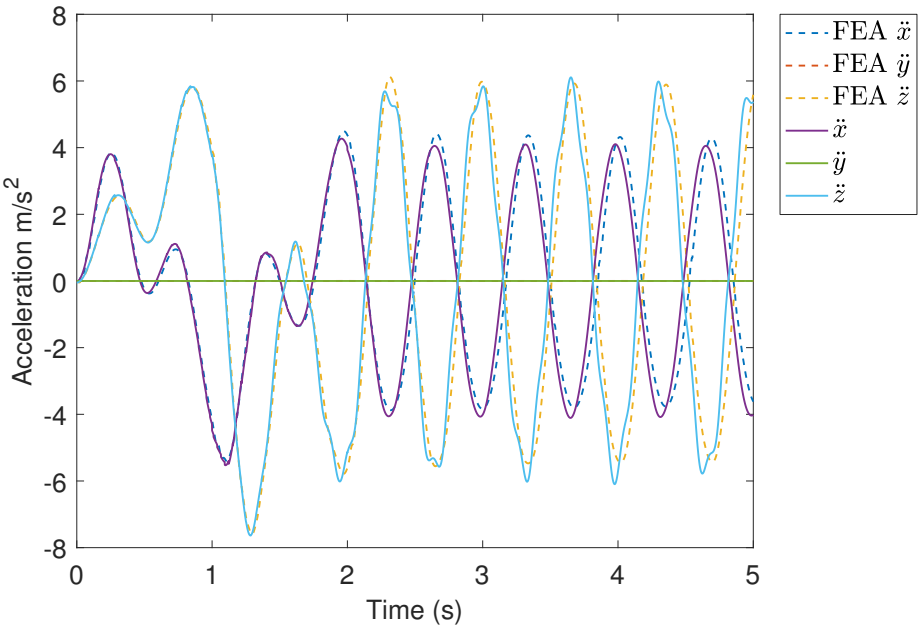


Figure 55: Comparison of Tip Acceleration Signals for Move 2 From The TMM Based Model and FEA Analysis

discrepancies between the accelerations in the y axis. It can also be seen that there is a small discrepancy between the frequencies for the two models, with the peaks of the signals from FEA lagging the ones obtained from the TMM based model (155).

The acceleration signals from move 2 are shown in Figure 55. The signals from the FEA simulation and the ones obtained from the model derived with TMM match up well. Note that since move 2 is purely planar there are no signals in the y axis. Again, it is clear that the peaks in the oscillations are slower in the FEA model.

Figure 56 shows the acceleration signals from move 3. The TMM based model diverges most during the motion where the torsional effects are the biggest. The response, however, still matches relatively well, with the magnitudes and the phase being very similar.

Taking into account that the acceleration signals matched well between the TMM based model and the FEA model, the acceleration signals from the TMM based model can be used evaluate the performance of the proposed EKF for flexible state reconstruction. Figure 57a shows the position of the tip from the TMM based model (solid lines) and the estimates

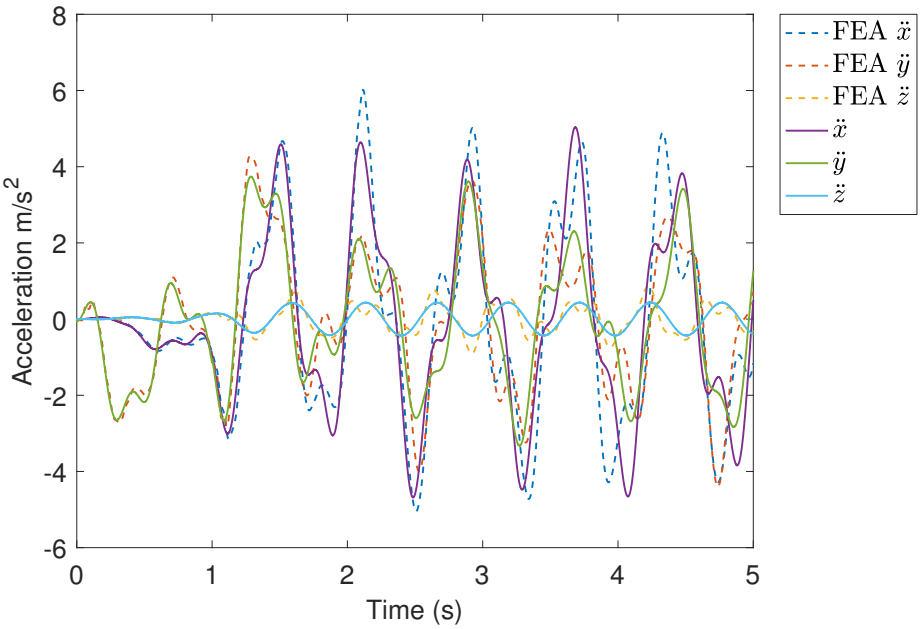
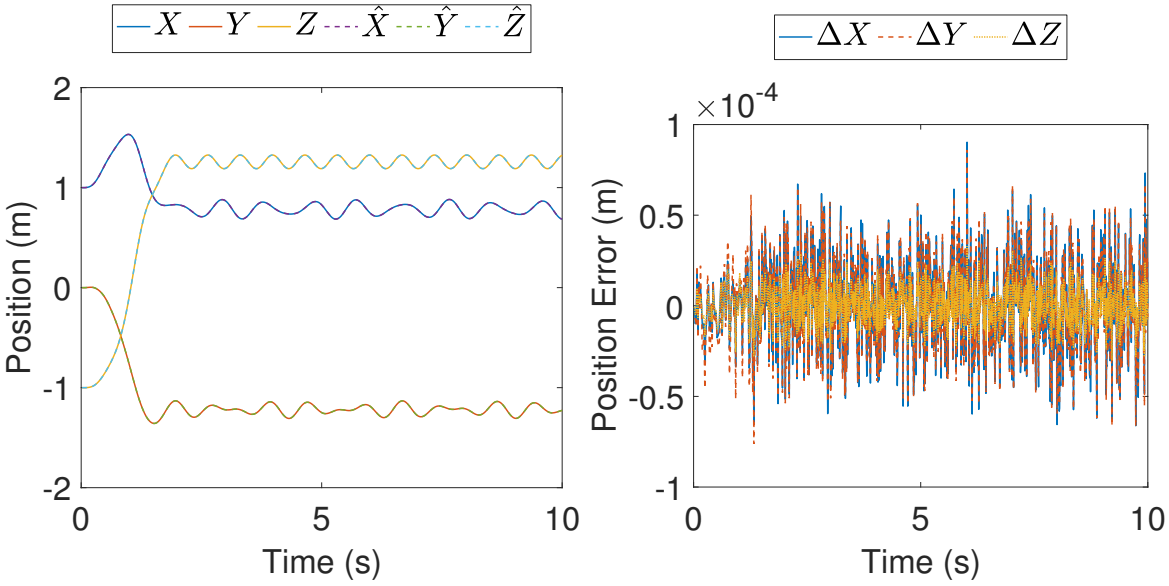


Figure 56: Comparison of Tip Acceleration Signals for Move 3 From The TMM Based Model and FEA Analysis

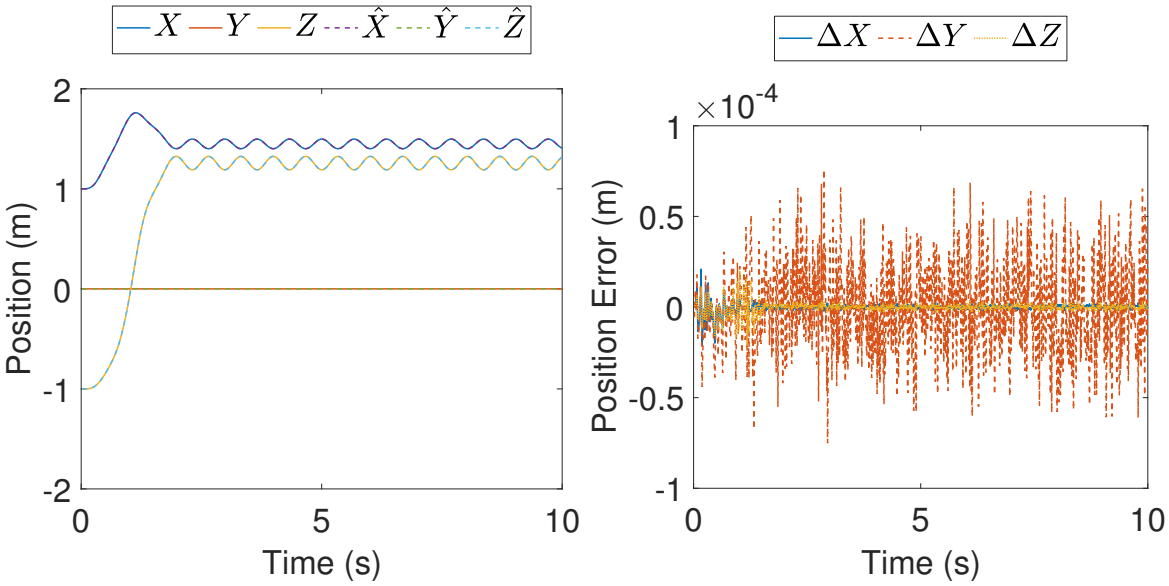


(a) Tip Position and Estimates for Move 1

(b) Tip Position Error for Move 1

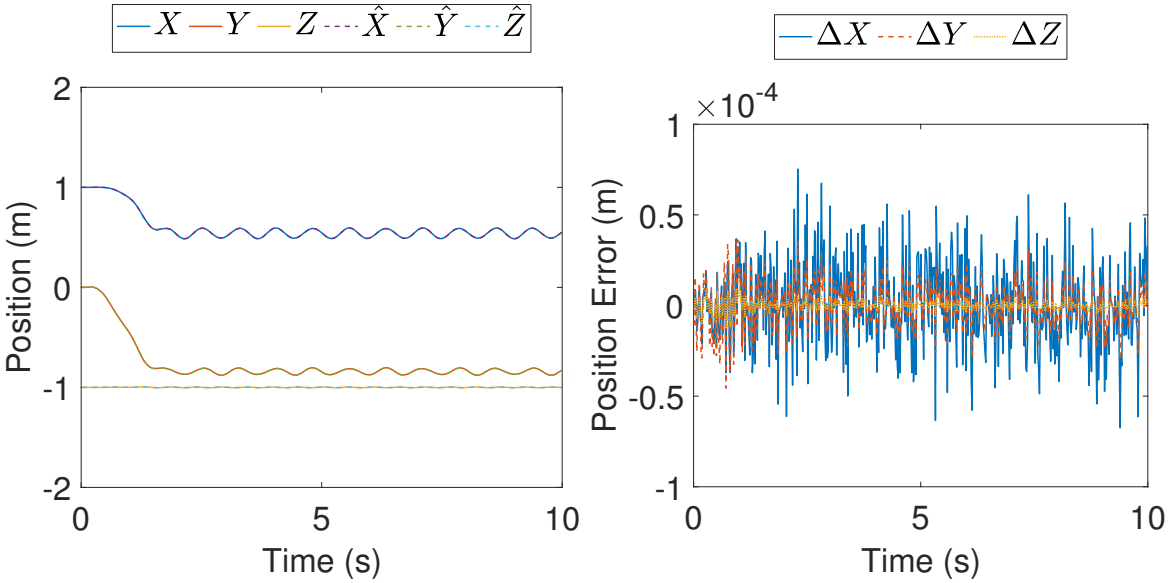
Figure 57: Move 1 Tip Position

(dashed lines) for Move 1. It is clear that the estimator was able to accurately track the modal amplitudes correctly. Figure 57b shows the error in the tip position for move 1. There is noisy *random walk* around 0 which is expected from a Kalman Filter.



(a) Tip Position and Estimates for Move 2 (b) Tip Position Error for Move 2

Figure 58: Move 2 Tip Position



(a) Tip Position and Estimates for Move 3 (b) Tip Position Error for Move 3

Figure 59: Move 3 Tip Position

Figure 58a shows the position of the tip for move 2. Again, the tip position from the estimated modal amplitudes match well with the modeled position. Figure 58b shows the same tip position error with the estimated modal amplitudes and its look resembles the

response in Figure 58b which is to be expected.

Figure 59a shows the position of the tip for move 3. Similarly to moves 1 and 2 there is very little difference between the estimated tip position and Figure 58b shows the same tip position error with the estimated modal amplitudes Figure 59b.

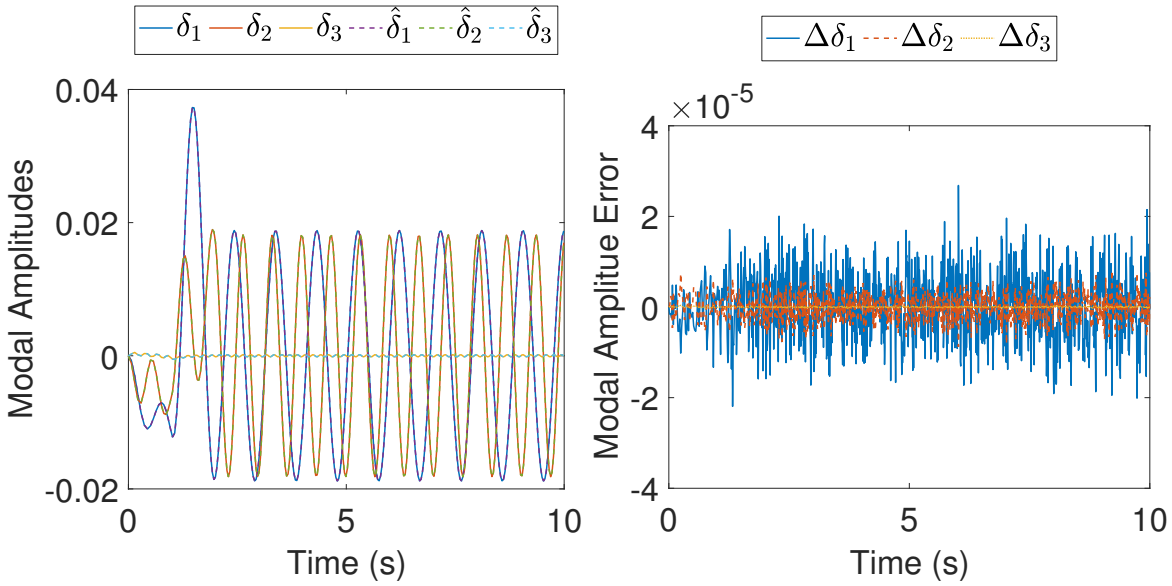
Extra figures for all moves showing details of modal amplitude errors and modal velocity errors can be found in Appendix G.1.

5.5.1.1 Robustness Analysis

If the initial state and the model dynamics are known by the estimator exactly, the sensor data is unbiased and the correct noise properties are known then the states are estimated correctly. Figure 60a shows the state variables δ and their estimates $\hat{\delta}$ for move 1. In Figure 60b, the difference between the estimated and modeled modal amplitudes, $\Delta\delta$, are shown. It can be seen that the error is three orders of magnitude smaller than the modeled values. Similarly Figure 61a shows the modal velocities for move 1. The error magnitude ($\Delta\dot{\delta} = \dot{\delta} - \hat{\dot{\delta}}$) is shown in Figure 61b. As with the modal amplitudes, the error is three orders of magnitude smaller than the modeled modal amplitudes. In this section various parameters in the model and sensor signal are varied to illustrate the robustness of the proposed EKF. All the simulations in this section are done with move 1 where all of joints move and flexures in all directions are excited.

Changing the initial guess to 0.1 for all states yielded results seen in Figure 62. Figure 62a show the modal amplitude error, and it can be seen that the estimator is able to quickly recover from the erroneous state estimate. The modal velocity error, seen in Figure 62b, took longer to converge to the correct value.

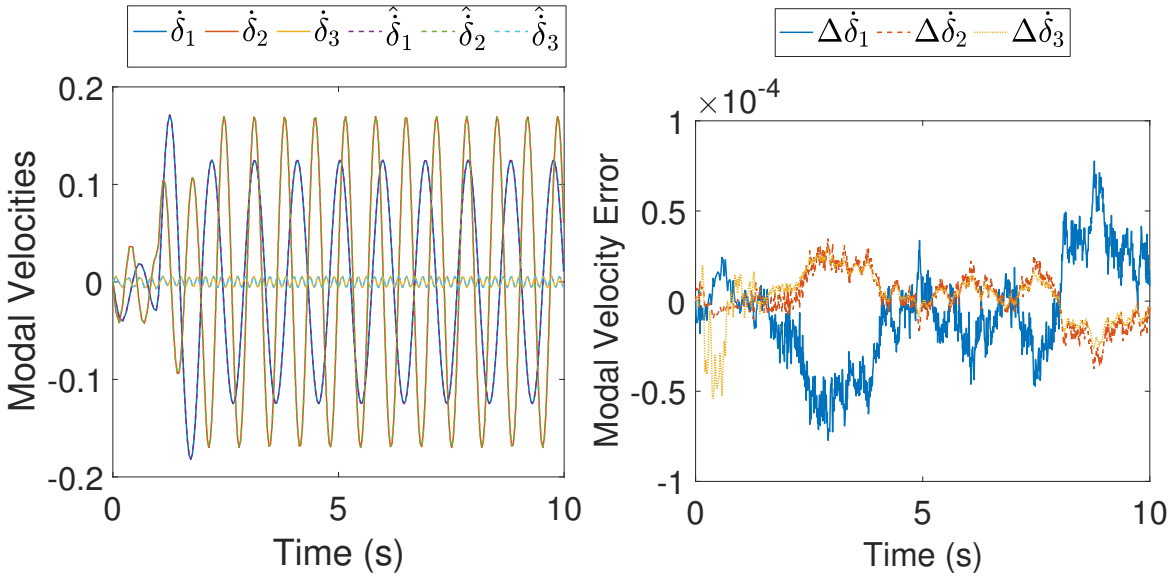
Adding a sensor off-set of 0.5 m/s^2 to all of the acceleration signals yielded results that can be seen in Figure 63. The modal amplitude errors seen in Figure 63a show that the error is still one order of magnitude lower than the modeled modal amplitudes seen in Figure 60a. The modal velocity errors, however, seen in Figure 63b, have an error magnitude about as large as the the modeled modal velocities as seen in Figure 61a and the states do not



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 60: Move 1 with Initial Guess Errors



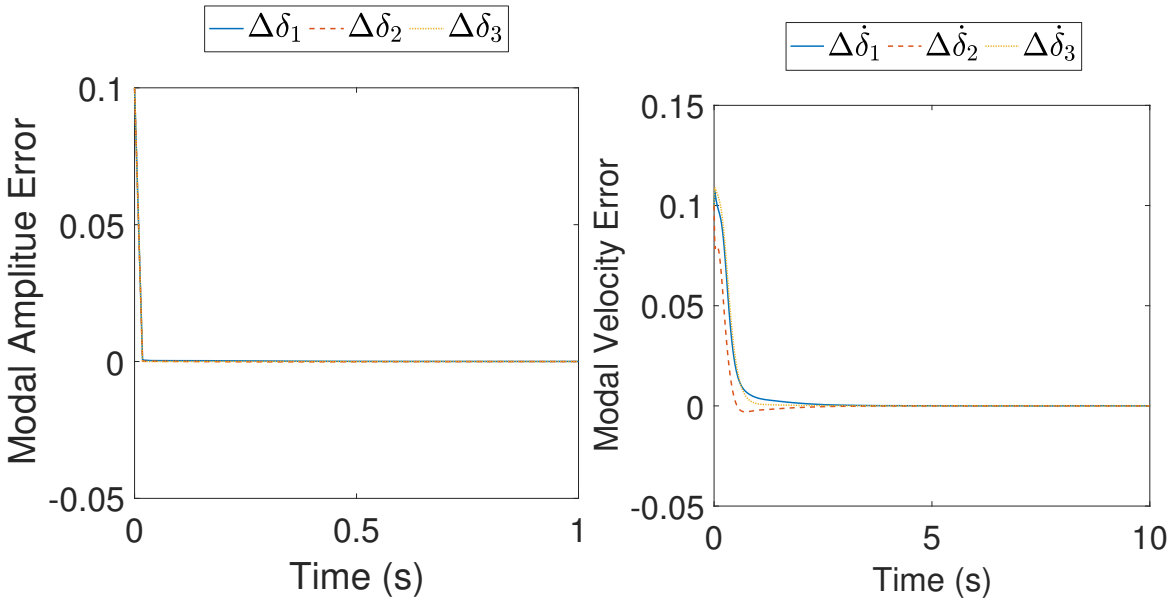
(a) Modal Velocities and Estimates

(b) Modal Velocity Error

Figure 61: Move 1 with Initial Guess Errors

converge to zero.

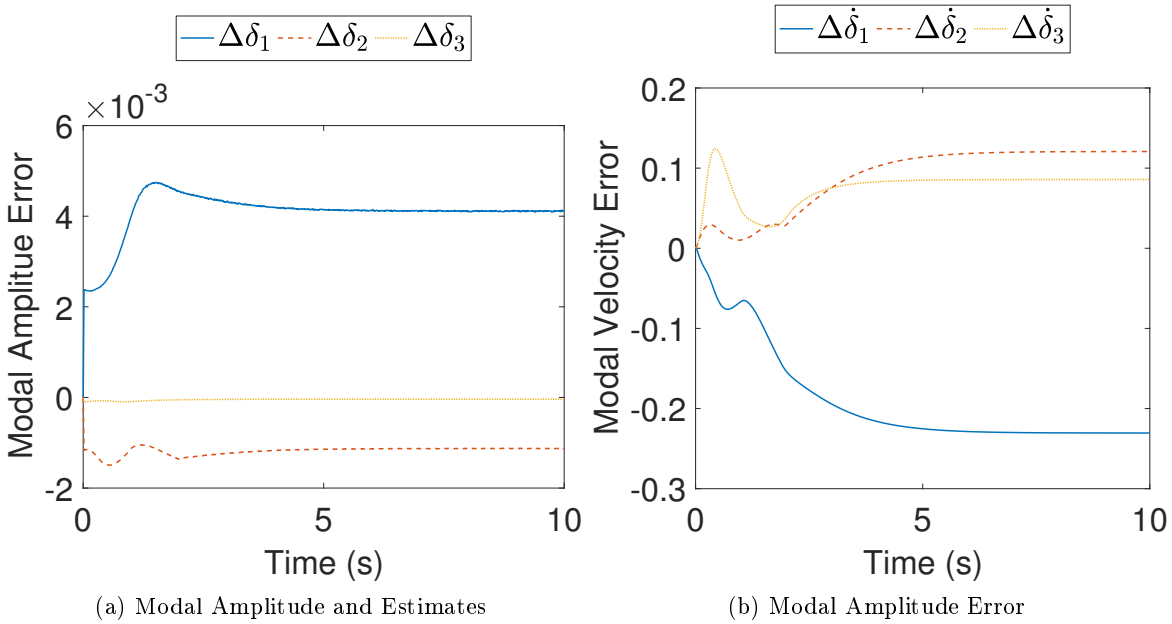
This problem can, however, be alleviated by adding a high-pass filter to the measurements. A high-pass filter with a cut-off frequency of one decade below of the lowest natural



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 62: State Errors to Wrong Initial Condition

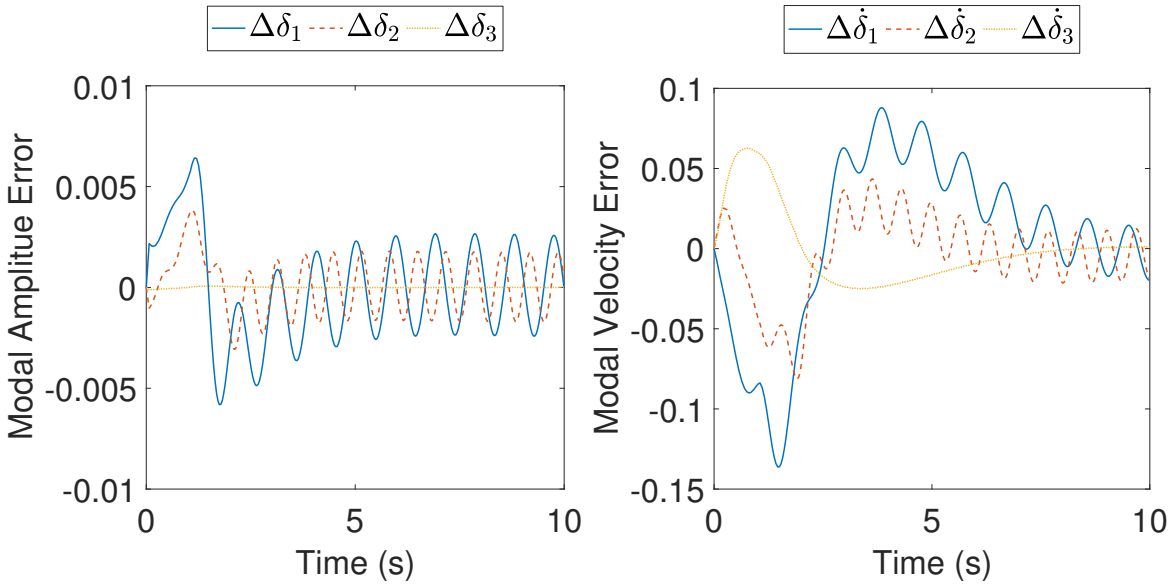


(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 63: State Errors to Sensor Off-Set

frequency in the system was used. Figure 64 shows the state errors when the high-pass filter was used. Figure 64a shows magnitude of the error has not decreased, however the error is now zero mean. Figure 64b shows an improvement over to Figure 63b as the modal velocity



(a) Modal Amplitude and Estimates

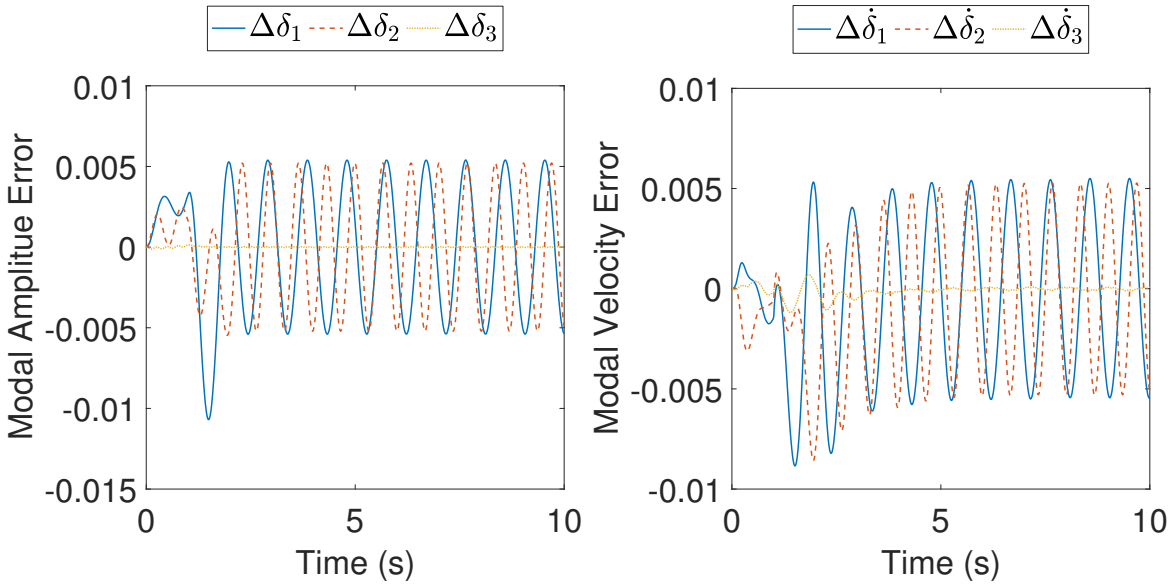
(b) Modal Amplitude Error

Figure 64: State Errors to Sensor Off-Set

error now has a smaller magnitude and the response is trending towards zero-mean. The modal amplitudes and the modal velocities have an oscillating error because the high-pass filter adds a phase shift into the signal. If the sensor offset is constant throughout, then the problem is solved by letting the filter eliminate the off-set before using them to make any measurements.

Figure 65 shows the modal amplitude and velocity errors when the system model in the estimator has a 40% model error in the stiffness Matrix. As can be seen in Figure 65a the error magnitude is about 4 times smaller than the nominal values shown in Figure 60a. Figure 65b shows that the modal velocities have an error magnitude that is about 20 times smaller than compared to Figure 61a. The error magnitude is oscillating since the frequencies in the estimator are differ from the model used to create the accelerometer signals.

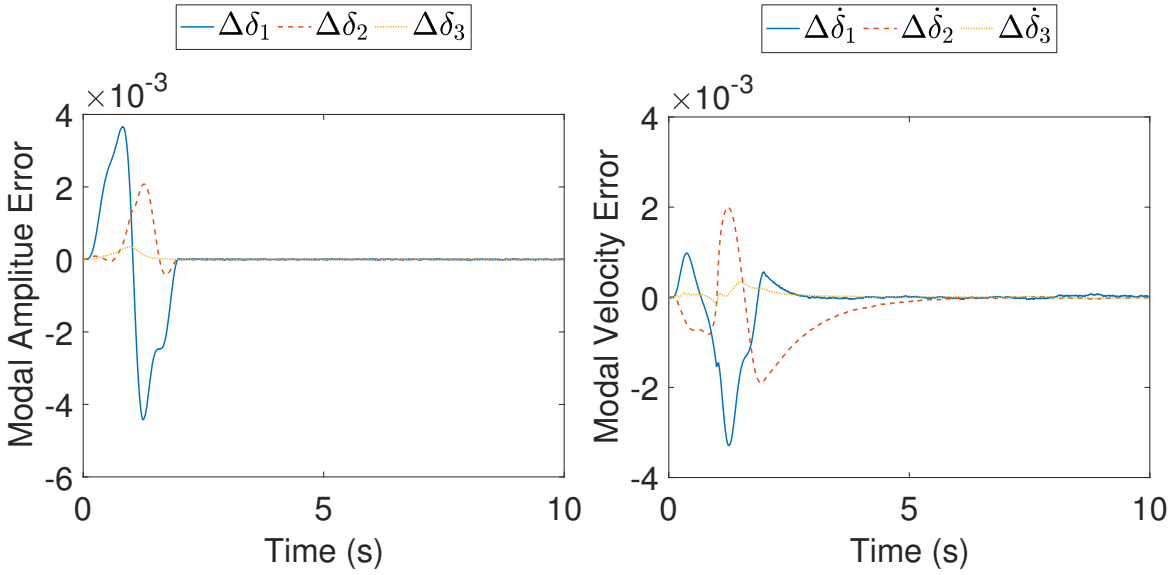
The state errors when the estimator model excludes Coriolis and centrifugal terms are shown in Figure 66. Figure 66a shows small modal amplitude errors during the motion and then quickly converging to the modeled states once the joint motion has stopped. Figure



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 65: State Errors to Increased Stiffness



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 66: State Errors to Exclusion of Coriolis and Centrifugal Terms

66b shows similar results as the modal velocities have small errors in the motion phase and after two seconds the states converge to the modeled states.

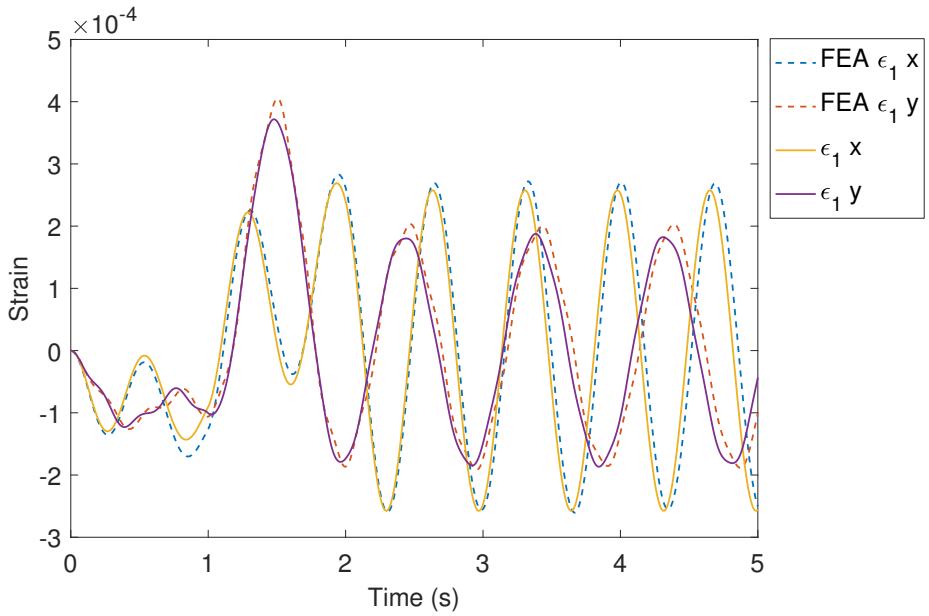


Figure 67: Comparison of Link 1 Strain Signals for Move 1 From The TMM Based Model and FEA Analysis

5.5.2 Strain Feedback

The measurement equation for strain feedback is given in (173). The noise covariance in this simulation study matches that of [60], where the authors used a strain gage and a high-speed analog input device to measure and control beam vibration in experiments. This choice was made to use realistic noise parameters. Therefore, the noise covariance matrix is defined as:

$$R = 9 \cdot 10^{-14}\mathbb{I}, \tag{186}$$

where $\mathbb{I}$ is the $r \times r$ identity matrix, and r is the number of sensor signals. Band-limited white noise with the same covariance was added to the simulated strain signals. Two strain gage pairs (one for x axis and the other for y) are placed on each link 5 cm from the start of the link.

The strain signals calculated in (173) were compared to the strain values that were calculated with COMSOL. Figure 67 shows the two strain signals from link 1 during move 1. The dashed lines obtained from FEA match very closely to the ones obtained from (173). The signals have a good match in terms of signal amplitude. It can also be seen that there

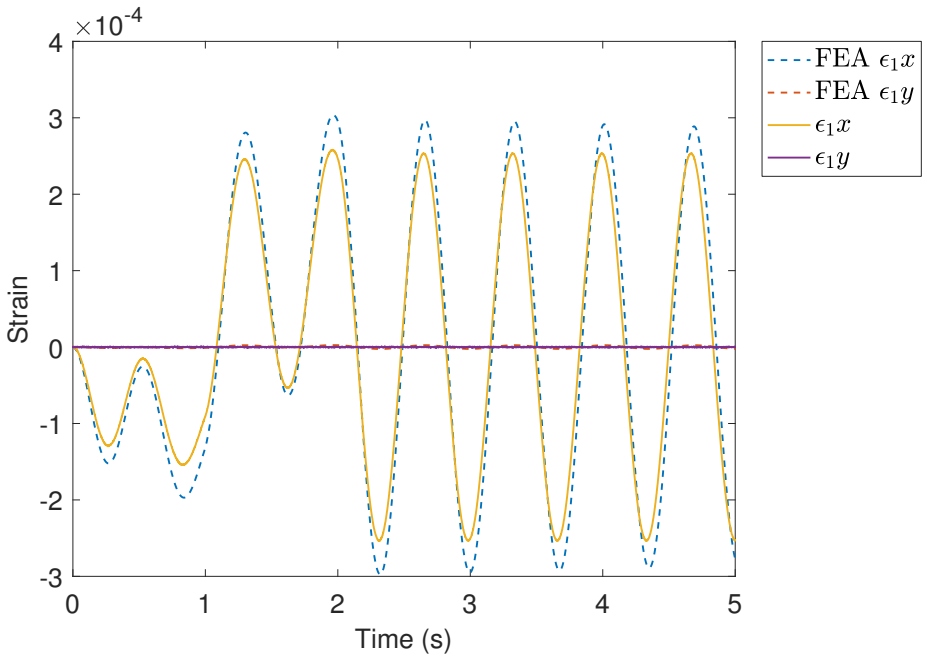


Figure 68: Comparison of Link 1 Strain Signals for Move 2 From The TMM Based Model and FEA Analysis

is a small discrepancy between the frequencies for the two models, with the peaks of the signals from FEA lagging the ones obtained from the TMM based model.

The strain measurements from move 2 are shown in Figure 68. The signals between the FEA simulation and the ones obtained from the model derived with TMM match up well. There is a small discrepancy between the magnitudes of the but the motion overall looks very similar. Note that since move 2 is purely planar there are no signals in the y axis based on the TMM model, there is, however a small amount of strain calculated in the FEA model.

Figure 69 shows the strain signals from move 3. The TMM based model diverges during move 3 due to unmodeled torsional effects. In the response, however, the signals still show a good trend and magnitude.

Taking moves 1-3 in consideration, it is evident that the strain signals from the TMM based model can be used evaluate the performance of the proposed EKF for flexible state reconstruction. Figure 70a shows the position of the tip from the TMM based model (solid

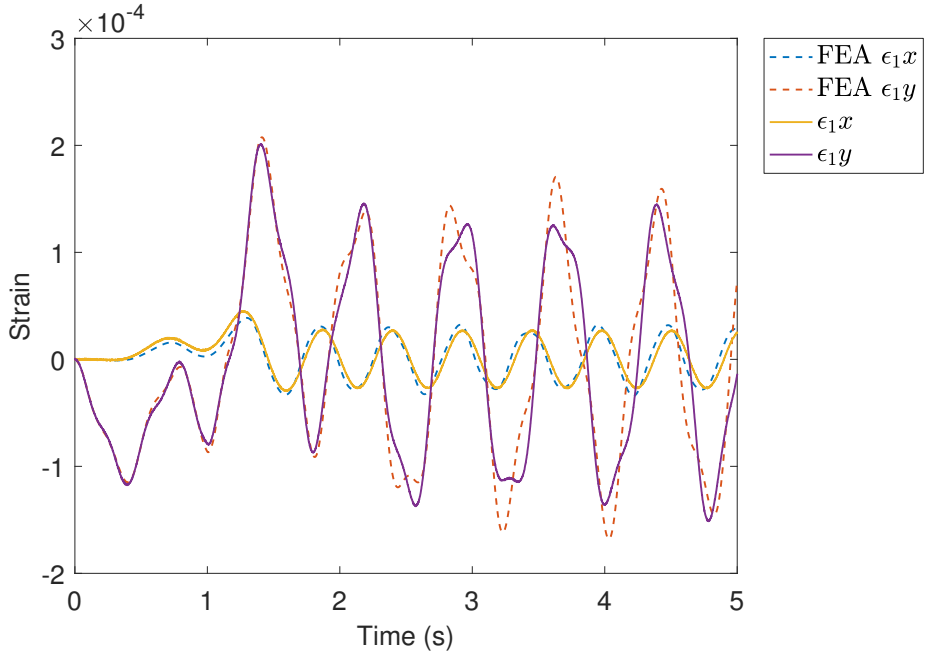
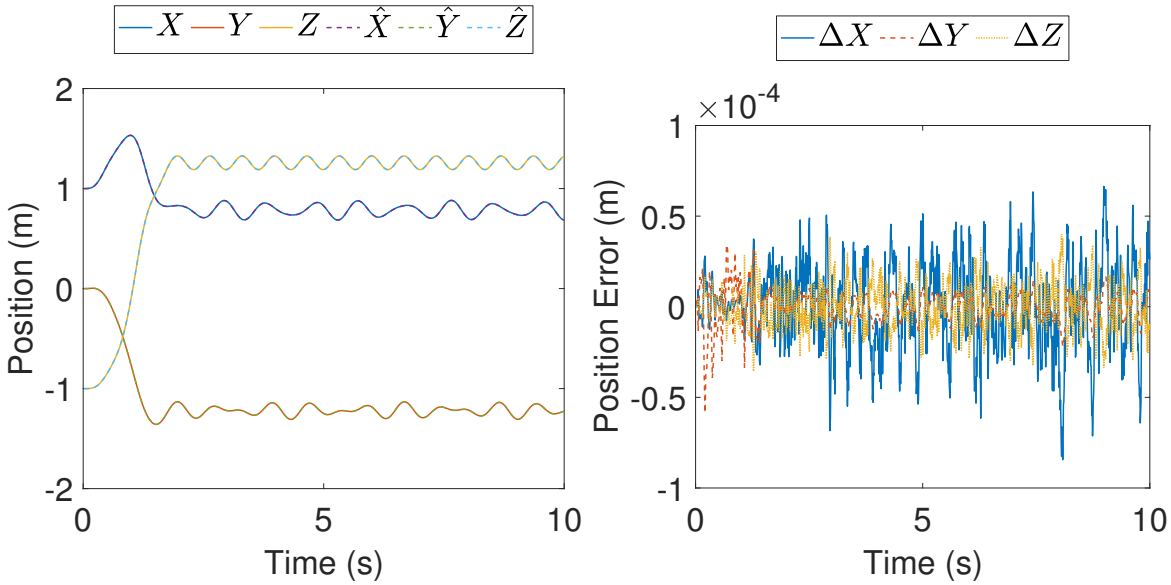


Figure 69: Comparison of Link 1 Strain Signals for Move 3 From The TMM Based Model and FEA Analysis

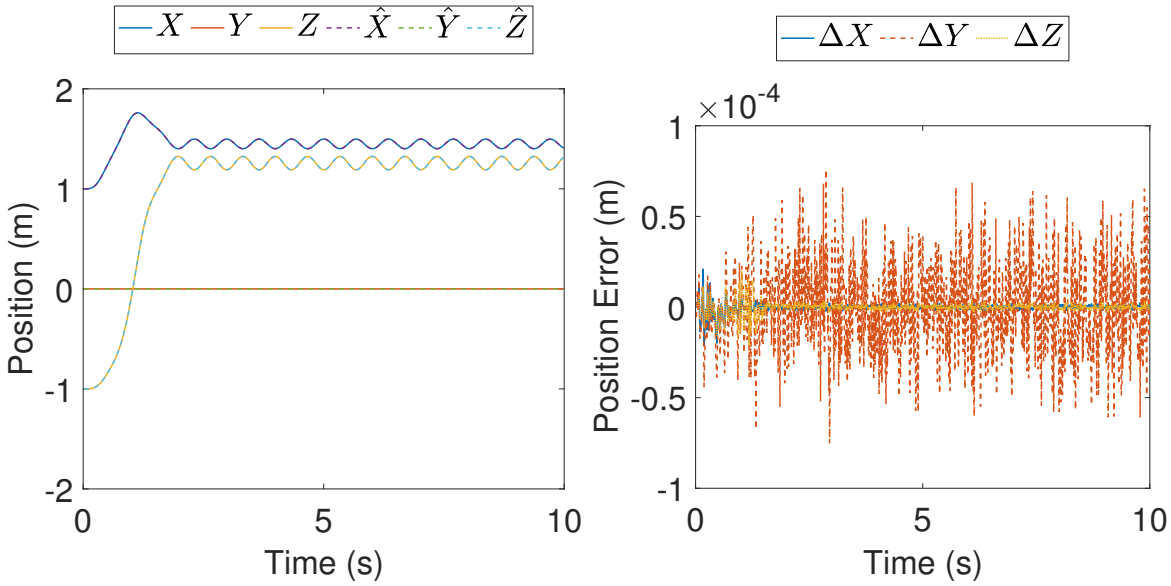


(a) Tip Position and Estimates for Move 1 from Strain Feedback

(b) Tip Position Error for Move 1

Figure 70: Move 1 Tip Position

lines) and the estimates (dashed lines) for move 1. It is clear that the estimator was able to accurately track the modal amplitudes using strain data. Figure 70b shows the error in



(a) Tip Position and Estimates for Move 2

(b) Tip Position Error for Move 2

Figure 71: Move 2 Tip Position

the tip position for move 1. There is noisy *random walk* around 0 which is normal behavior from a Kalman filter.

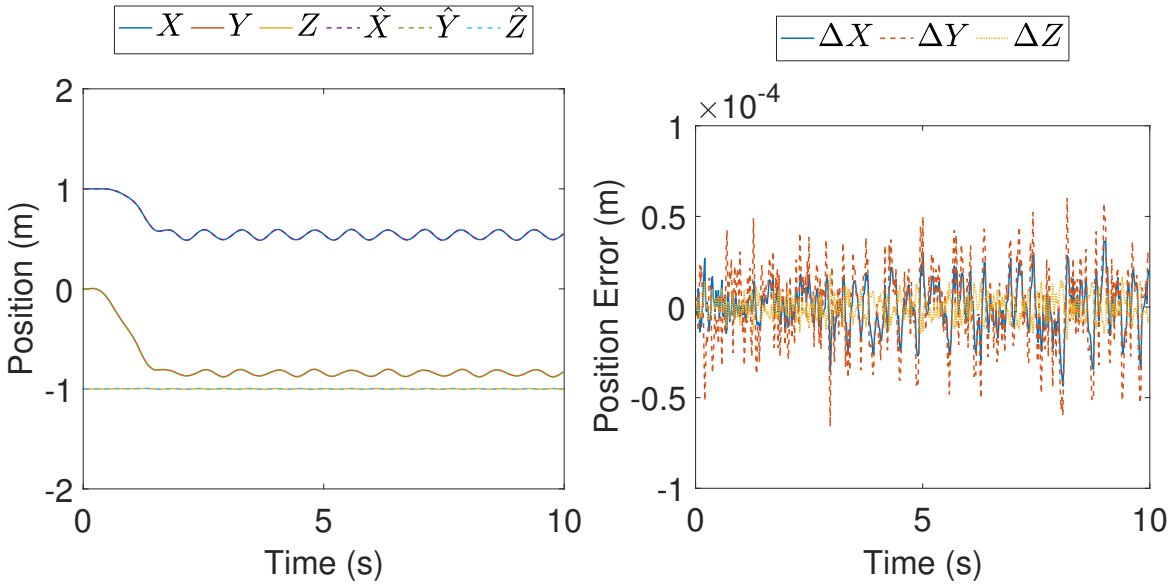
Figure 71a shows the position of the tip for move 2. Again, the tip position from the estimated modal amplitudes matches well with the modeled position. Figure 71b shows the same tip position error with the estimated modal amplitudes and as expected it has similar characteristics to 71b.

Figure 72a shows the position of the tip for move 3. Similarly to moves 1 and 2 there is very little difference between the estimated tip position and Figure 71b shows the same tip position error with the estimated modal amplitudes Figure 72b.

Extra Figures for all moves showing details of modal amplitude errors and modal velocity errors can be found in Appendix G.2

5.5.2.1 Robustness Analysis

If the initial guess initial guess and the model dynamics are known by the estimator exactly, the sensor data is unbiased, and the correct noise properties are known, then the states are estimated correctly. Figure 73a shows the state variables δ and their estimates



(a) Tip Position and Estimates for Move 3

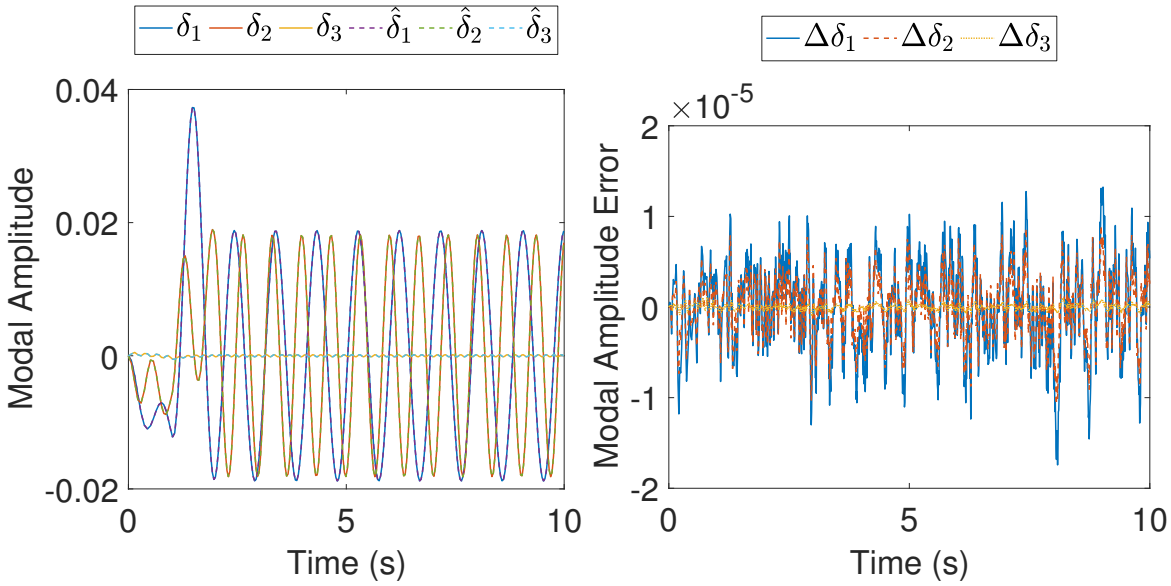
(b) Tip Position Error for Move 3

Figure 72: Move 3 Tip Position

$\hat{\delta}$ for move 1. In Figure 73b, the difference between the estimated and modeled modal amplitudes ($\Delta\delta = \hat{\delta} - \delta$) is shown. It can be seen that the error is three orders of magnitude smaller than the modeled values. Similarly Figure 74a shows the modal velocities for move 1. The error magnitude ($\Delta\dot{\delta} = \dot{\delta} - \hat{\dot{\delta}}$) is shown in Figure 74b. As with the modal amplitudes, the error is three orders of magnitude smaller than the modeled modal amplitudes. In this section various parameters in the model and sensor signals are varied to find out how robust the proposed EKF is. Similarly to measurements from accelerometers, the robustness of strain based estimation is evaluated with move 1.

Changing the initial guess to 0.1 for all states yielded results seen in Figure 75. Figure 75a shows the modal amplitude error, and it can be seen that the estimator is able to quickly recover from the erroneous state estimate. The modal velocity error, seen in Figure 75b, converged relatively quickly but it did induce a large error directly after the start of estimation.

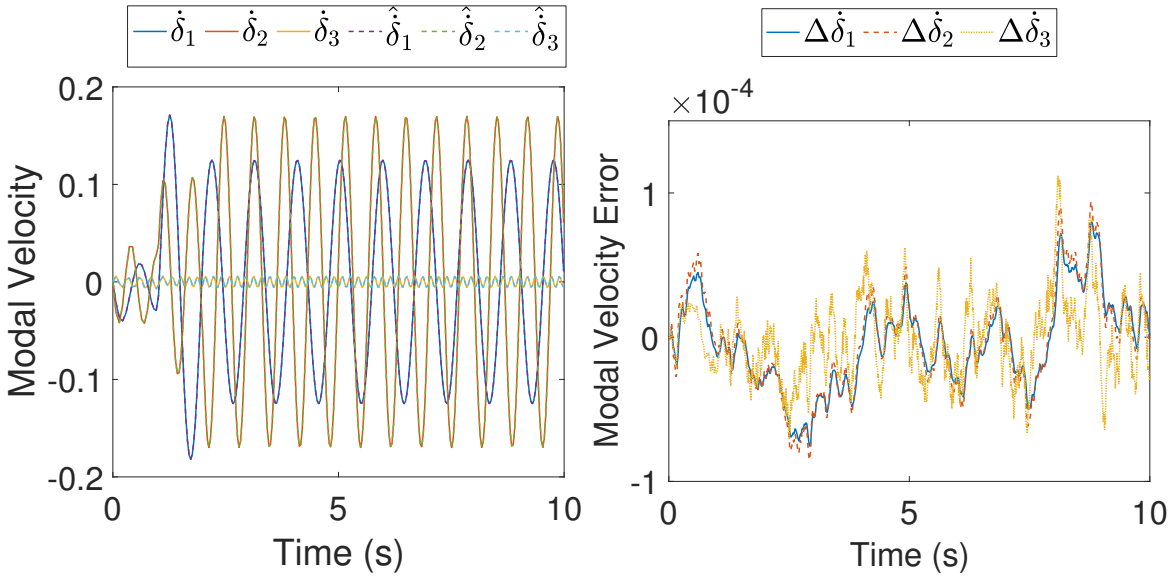
Adding a strain offset of $5 \cdot 10^{-5}$ (10% of the signal magnitude) to strain signals yielded results that can be seen in Figure 76. The modal amplitude errors seen in Figure 76a show



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 73: Move 1 with Initial Guess Errors

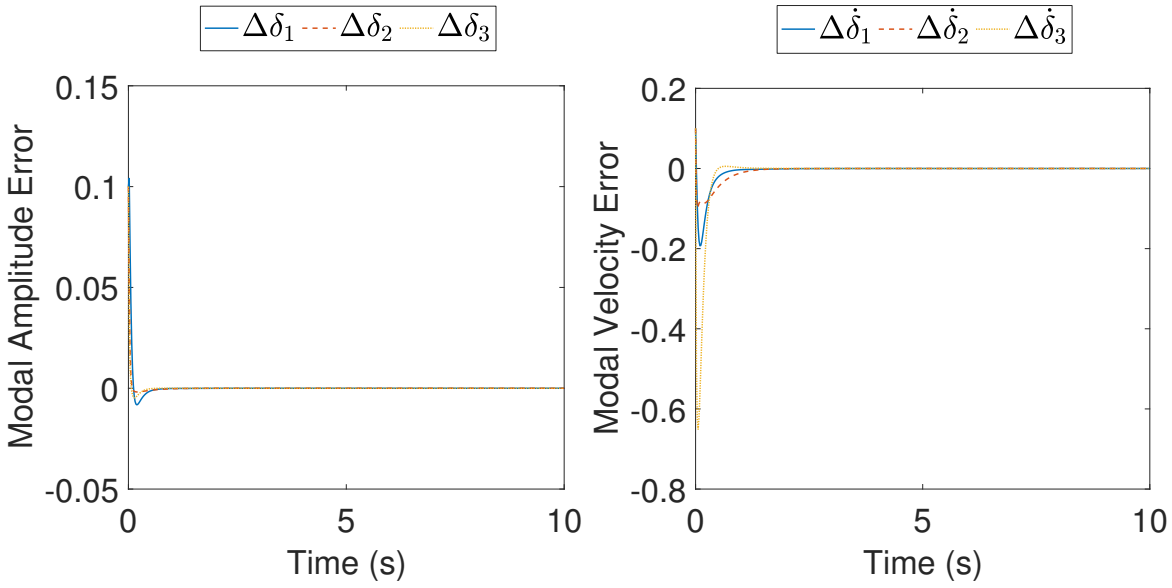


(a) Modal Velocities and Estimates

(b) Modal Velocity Error

Figure 74: Modal Amplitudes for Move 1

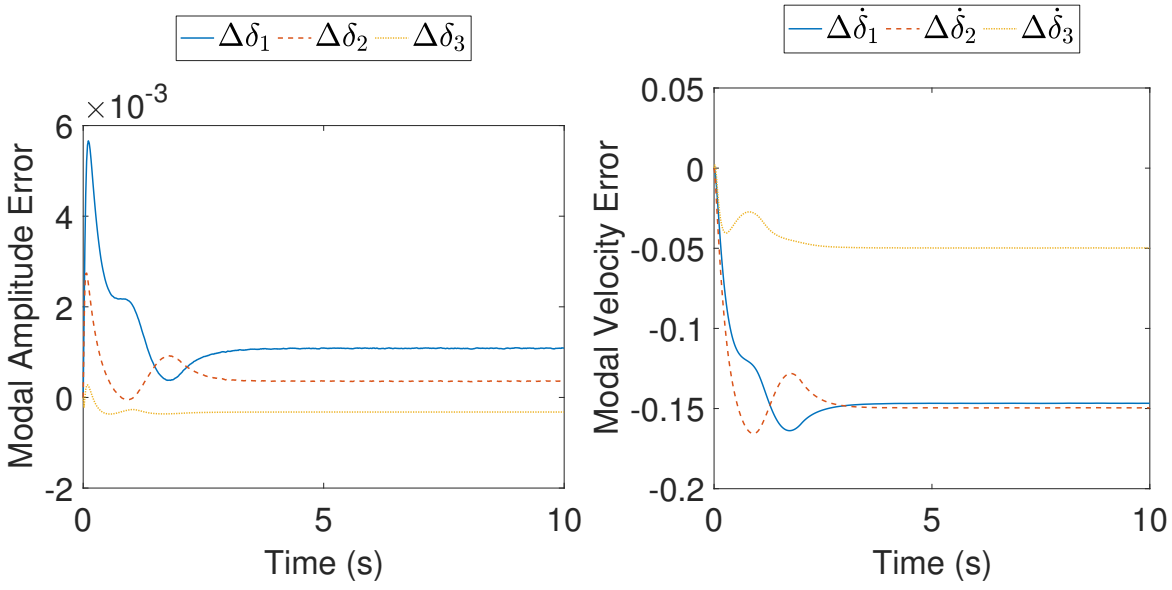
that the error is still one order of magnitude lower than the modeled modal amplitudes seen in Figure 73a and has a bias. The modal velocity errors, however, seen in Figure 76b, have an error magnitude about as large as the modeled modal velocities as seen in Figure 74a.



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 75: Move 1 with Initial Guess Error

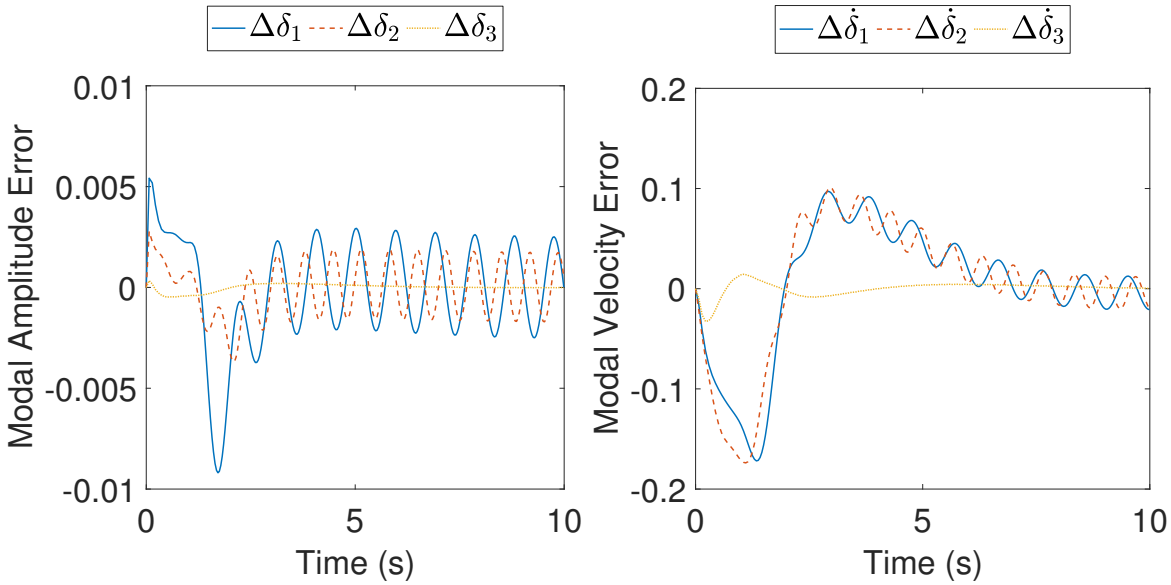


(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 76: State Errors to Sensor Off-Set

Adding a high pass-filter to the strain signals should help the estimation process. A high-pass filter with a cut-off frequency of 1/10 of the lowest natural frequency in the system was used. Figure 77 shows the state errors when the high-pass filter was used. Figure 77a shows



(a) Modal Amplitude and Estimates

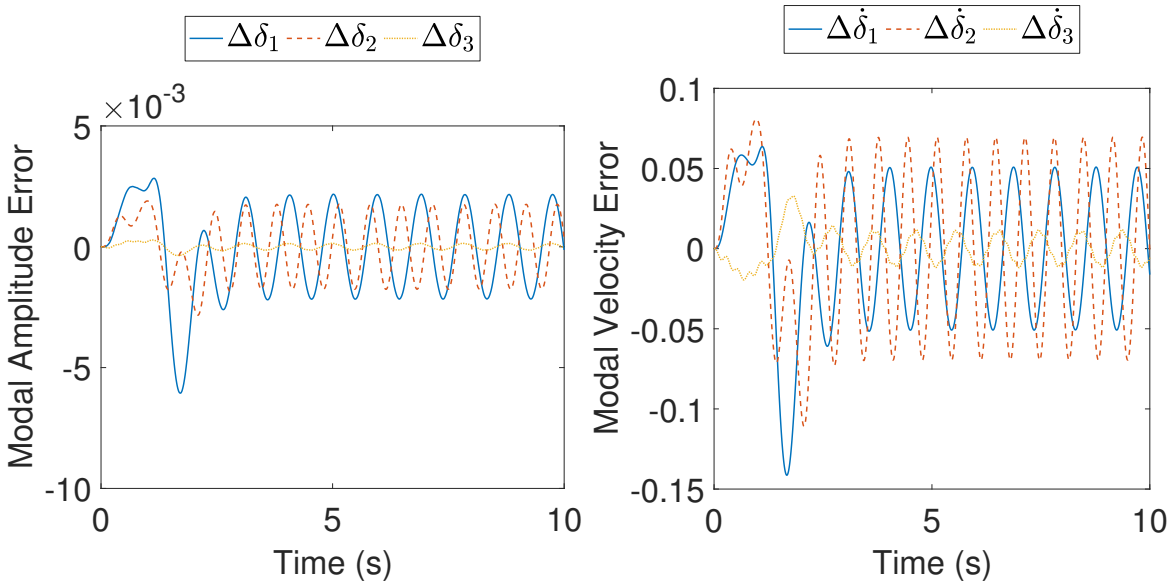
(b) Modal Amplitude Error

Figure 77: State Errors to Sensor Off-Set

magnitude of the error has not decreased, however the error is now zero mean. Figure 77b implies that the high-pass filter has improved the estimation of modal velocities as the modal velocity error now has a smaller magnitude and the response is trending towards zero-mean. The modal amplitudes and the modal velocities have an oscillating error because the high-pass filter adds a phase shift into the signal.

Figure 78 shows the modal amplitude and velocity errors when the system model in the estimator has a 40% model error in the stiffness Matrix. As can be seen in Figure 78a the error magnitude is about 4 times smaller than the nominal values shown in Figure 73a. Figure 78b shows relatively large error when compared to Figure 74a. The error magnitude is oscillating since the frequencies in the estimator differ from the model used to create the strain signals.

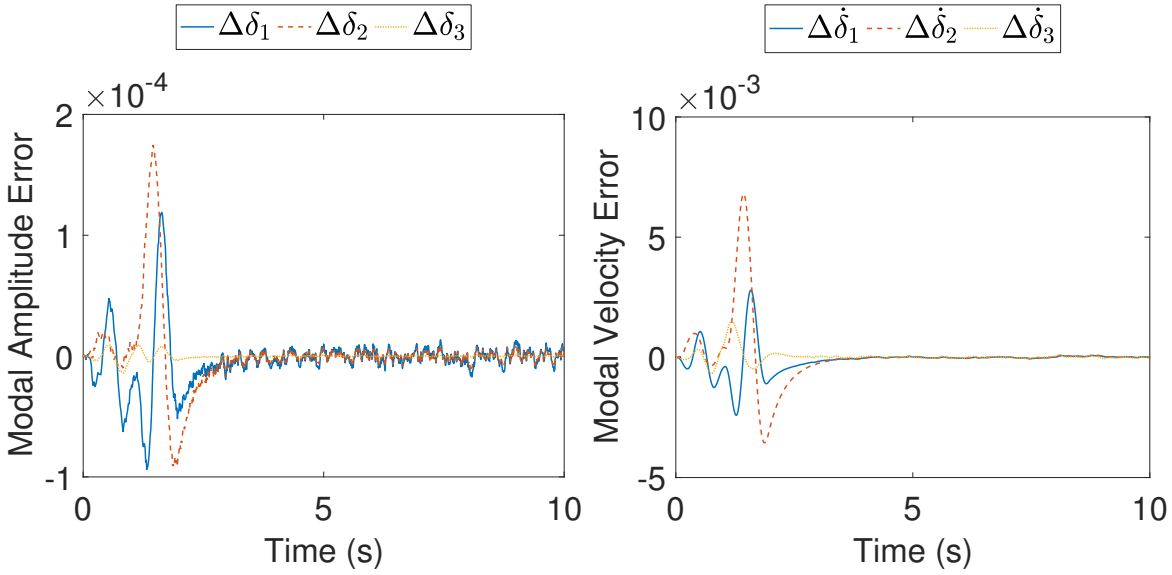
The state errors when the estimator model excludes Coriolis and centrifugal terms are shown in Figure 79. Figure 79a shows small modal amplitude errors during the transient motion and then quickly converging to the modeled states once the joint motion has stopped. Figure 79b shows similar results as the modal velocities have small error during the transient



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 78: State Errors to Increased Stiffness



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 79: State Errors to Exclusion of Coriolis and Centrifugal Terms

phase and after two seconds the states converge to the modeled states.

5.5.3 Simulation Study Summary

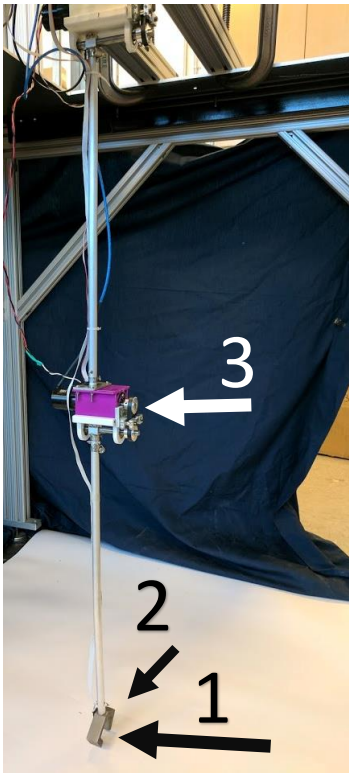
The simulations showed that the EKF is a good solution for estimating flexible states even in the presence of uncertainties. These simulations were performed with noise characteristics found on certain real-world sensors. Therefore the estimator performance is not representative of other accelerometer or strain measurements. The simulation results also depend heavily on the process noise covariance, which is hard to estimate for any real model. Additionally, because flexible robot dynamics are derived in continuous time, real-time implementation requires fixed step-solvers and they are numerically less robust that variable step solvers because the latter can use small step sizes when the solutions are not converging. It was empirically determined that different R and Q matrices result in different numerical stability. Therefore care must be taken before real-world implementation.

5.6 Experiments using Accelerometer Feedback

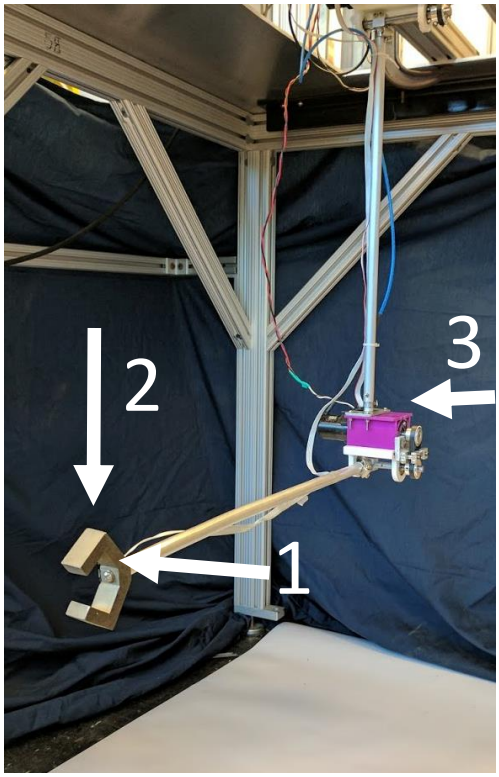
In order to illustrate the effectiveness of the estimator further, tests were conducted on the test bed that was described in Chapter 4. The estimator implemented on the test bed is used to observe the first three modes. Although, the fourth mode has a similar frequency to mode three, it was neglected because the belt drives damp it out very quickly. The mode shapes for the estimated modes were shown in Section 4.4 for two illustrative configurations. Tracking three modes instead of four allows for a significant drop in the computational demand, and allows the estimator to run at 1 kHz. This loop rate was empirically determined to be the speed at which the fixed step ODE solvers are numerically stable for integrating the estimator models.

5.6.1 Impulse Tests

This section illustrates the estimators response to impulse-type disturbances. Figure 80 shows the two configurations that are tested. Figure 80a shows the order and direction where the impulses were given to the arm when $q_3 = 270°$. Analogously, Figure 80b shows the locations and directions where impulses were given to the arm when $q_3 = 360°$. All of



(a) Impulse Locations and Directions at $q_3 = 270°$



(b) Impulse Locations and Directions at $q_3 = 360°$

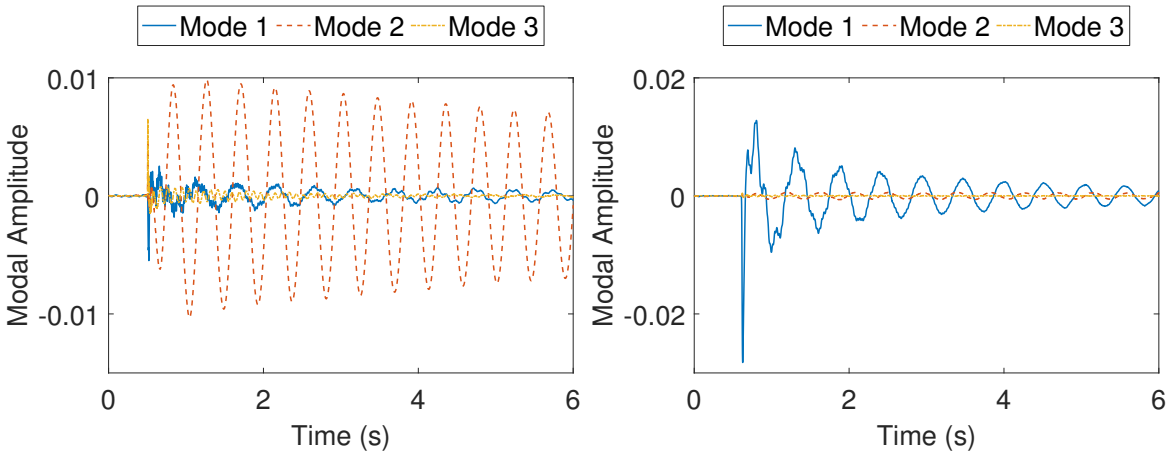
Figure 80: Direction and Location of Impulses for Two Configurations

the impulses were manually induced by a rapid tap.

Figure 81 shows the responses from impulse tests in the $q_3 = 270°$ configuration. Figure 81a shows the response of the estimator to the first impulse. As can be expected from the mode shapes, shown in Section 4.4, mode 2 is excited the most. The estimator was able to quickly recover the state from the disturbance.

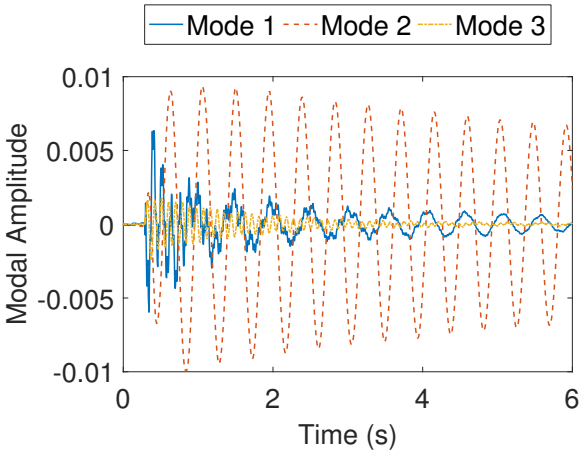
Figure 81b shows the response from the estimator for the second impulse. This time mode 1 is excited most since the impulse is in the direction of its mode shape. Comparing Figures 81a and 81b illustrates that mode 1 has a significantly higher damping ratio, due to the joint orientations, than mode 2, even though their natural frequencies are very close (2 vs 2.4 Hz).

Figure 81c shows the response to the third impulse. It can be seen that, unlike the responses from the first 2 impulses, the third mode is clearly evident, even though it damps



(a) Response to Impulse 1

(b) Response to Impulse 2



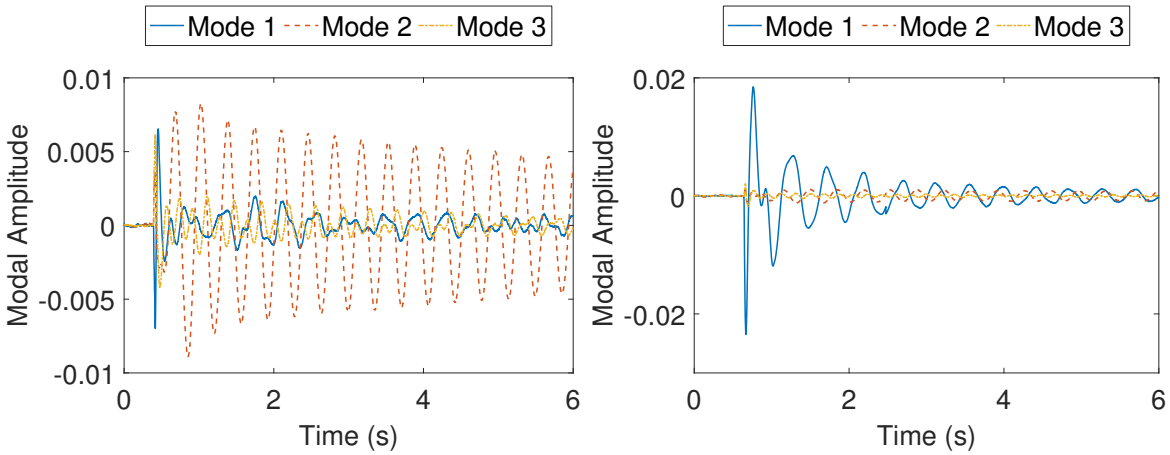
(c) Responses to Impulse 3

Figure 81: Response to Impulses at $q_3 = 270°$

out relatively quickly due to its higher frequency.

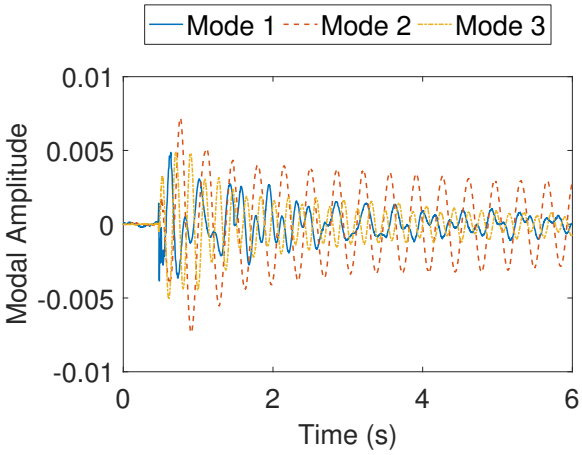
Figure 82 shows the responses from impulse tests in the $q_3 = 360°$ configuration. Figure 82a shows the response of the estimator to the first impulse. Mode 2 is excited the most and it dominates the response. Mode 1 also shows up in the respons; however, its magnitude is much lower. In general, we would expect that other modes would be excited, but with lower magnitudes, because the impulse is not exactly applied in the orthogonal direction of modes 1 and 3.

Figure 82b shows the response from the estimator for the second impulse. Again mode 1 is excited the most since the impulse is in the direction of its mode shape. Similarly to Figure 81b, the magnitude of mode 1 damps out relatively fast due to imperfect clamping



(a) Responses to Impulse 1

(b) Responses to Impulse 2



(c) Responses to Impulse 3

Figure 82: Response to Impulses at $q_3 = 360°$

at the joints.

Figure 82c shows the response to the third impulse. It can be seen that all of the modes are present in the response. Unlike in Figure 81c, mode 3 retains it magnitude for longer since its frequency is lower in this configuration than the previous one. Recall that the configuration-dependent frequencies were presented in Section 4.4.

During all of these tests the estimator responded to the disturbances in the order on milli-seconds. Taking into account the locations and directions of the impulses, and the mode shapes, all of the responses made sense physically. Therefore, it can be concluded that the estimator is a suitable disturbance observer.

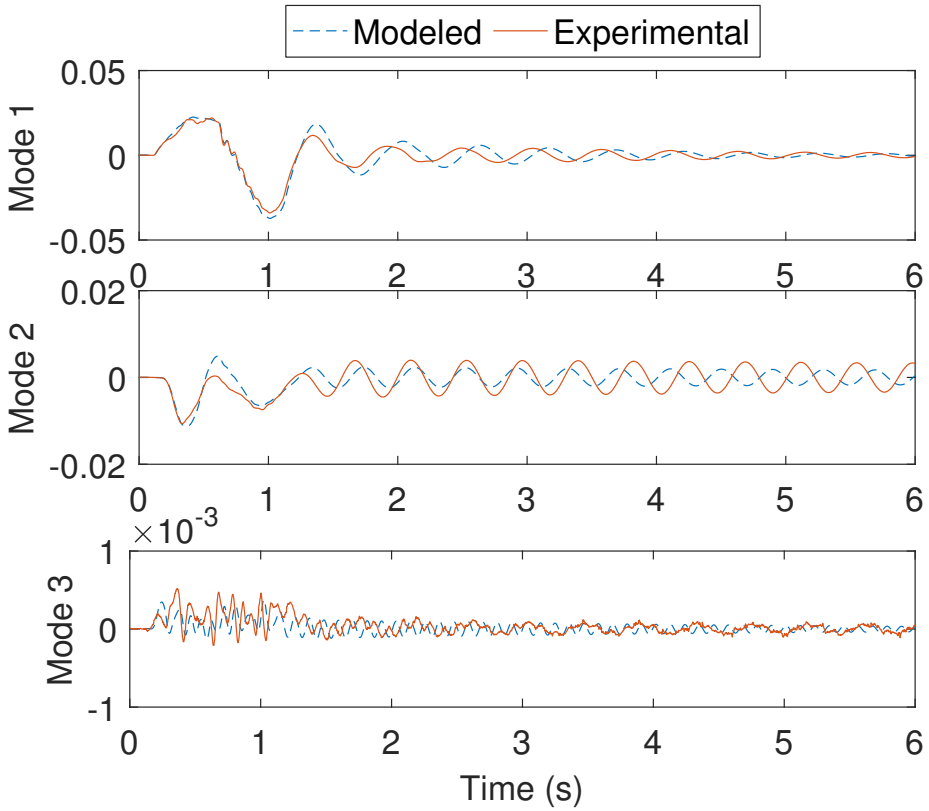


Figure 83: Modeled and Experimental Modal Amplitudes for Move 1

5.6.2 Motion Tests

In order to experimentally validate the estimator during the motion of the arm, three motions were tested. The same three motions that were used in Section 4.5.2 were used for this evaluation.

During move 1 all joints are given a bang-bang acceleration command resulting in a change of 1 *rad* for each joint. The duration of the motion is 1 s. Joint q_1 moves from 0 to -1 *rad*, joint q_2 moves from 1 to 2 *rad*, and joint q_3 moves from 5.2 to 4.2 *rad*. Figure 83 shows how the modal amplitudes, obtained from simulating the motion using recorded data, match the experimentally obtained ones. Mode 1 slightly lags the model; however, the amplitudes match well. The experimentally obtained Mode 2 data slightly leads the modeled data and has a similar magnitude. Mode 3 data does not match well between the model and the experiment. However, their amplitudes are very low compared to modes 1

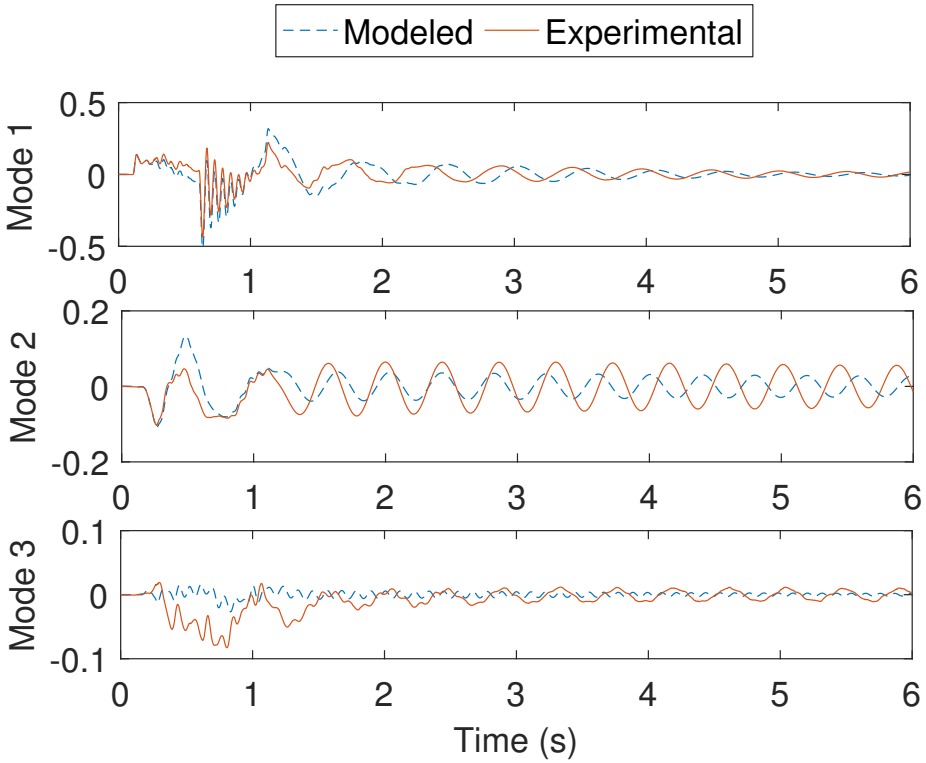


Figure 84: Modeled and Experimental Modal Velocities for Move 1

and 2, and do not have a big effect on the end-effector position.

Figure 84 shows the modal velocities from simulating move 1 using recorded data, and the experiment. The experimentally obtained mode shape lags the simulated one, but has a similar amplitude. There is a region of fast vibrations at around the 1 second mark. This is due to the noisy calculation of the second derivative of the encoder position. The value is then used in the estimator model and in the simulation. The experimentally obtained mode 2 leads the simulated data but has similar magnitudes. The experimentally obtained mode 3 is not zero mean in the transient motion phase. This is due to imperfect gravity compensation of the accelerometers, and the high pass filter not being able to get rid of the offset terms fast enough. Interestingly, the frequency in the third mode follows mode 1 after 2 seconds. This might also be an artifact of the improper gravity compensation.

Move 2 is a planar move where joint $q1$ remains stationary, joint q_2 moves from 1 to 2 rad, and joint q_3 moves from 6.28 to 5.28 rad. Figure 85 shows the experimentally-obtained

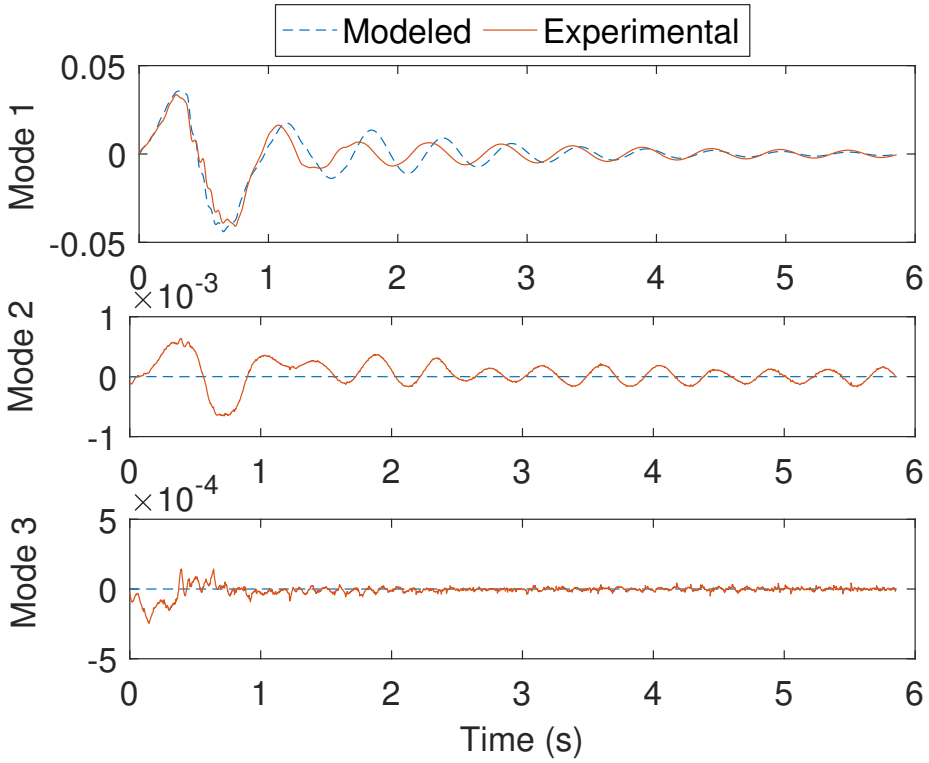


Figure 85: Modeled and Experimental Modal Amplitudes for Move 2

modal amplitudes and those obtained by simulating the model. In this, move only the first mode is excited. The experimentally obtained modal amplitude matches the simulation well. After the transient phase, the frequency is slightly lower than the simulated one. This is consistent with the results from move 1. The second and third modes are, theoretically, not excited. Their experimental values are 2 orders of magnitude lower than mode 1 and do not perceivably affect the motion of the tip.

During move 3 joints q_2 and q_3 are held stationary and joint q_1 moves from 0 to -1 rad. As described in Section 4.5.2, the accelerometer readings obtained from this move did not match well with the simulation. It can be seen that the experimentally-obtained mode 1 and mode 2 data match their simulated counterparts well. The experimental mode 3, however, has a much lower amplitude after the transient phase when compared to the simulation. Because mode 3 has a big influence in the accelerometer readings, the accelerations presented in Section 4.5.2 are inaccurate. However, only mode 3 that is damped out due to an modeled effect. The same kind of behavior could be seen with the different move time and distances.

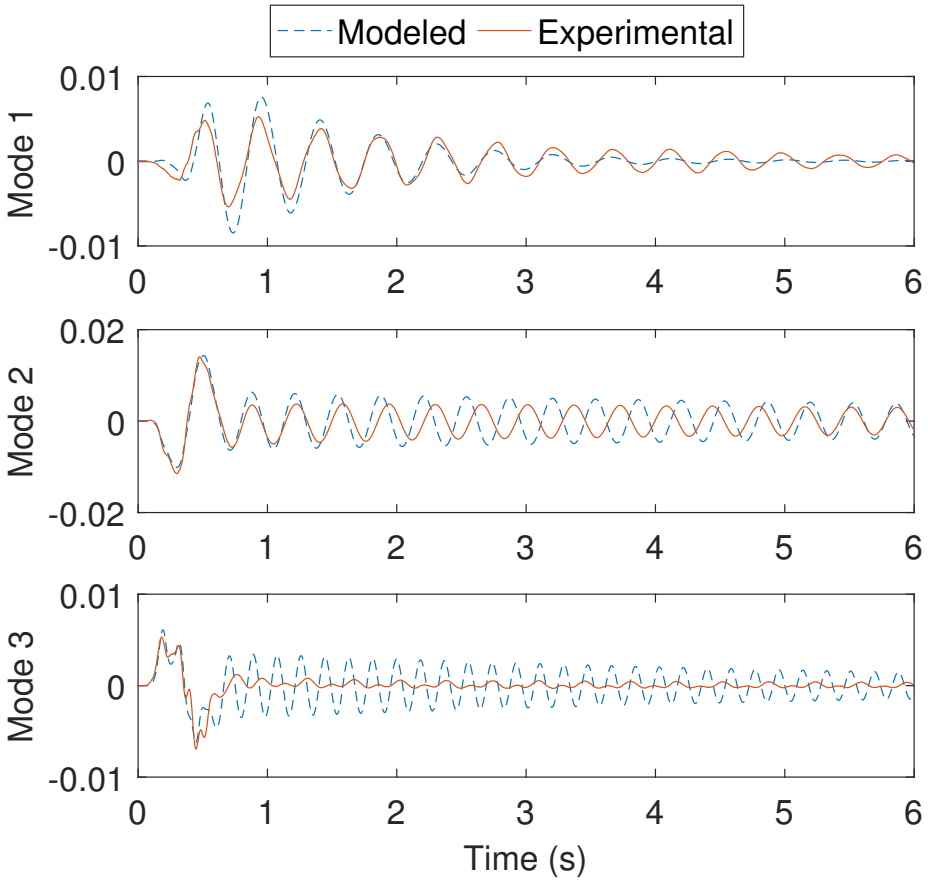


Figure 86: Modeled and Experimental Modal Amplitudes for Move 3

Overall, the estimator works well. The modal discrepancies seen in this section can be explained by the accelerometer readings presented in Section 4.5.2 and vice-versa. Noted that it took a considerable amount of effort fine tuning the initial conditions for the estimator to make it numerically stable. Furthermore, during some experiments it would go unstable. This behavior was not consistent and usually re-running the experiment would correct the problem. Additionally, the model for the accelerometers requires joint acceleration, that is generally not smooth after two derivatives from the encoder readings. Aggressive low-pass filtering helps smooth the signal, but also introduces a phase lag into the system and does not allow genuine fast acceleration signals through.

Additional plots from modal velocities can be found in Appendix G.3.

5.7 Chapter Summary

In this chapter, a novel use of Kalman filtering techniques was used to estimate the flexible states for a multi-link spatial flexible robot arm using data from accelerometers and strain gages. The use of a very efficient model for flexible robots allows real-time computation when using a modern computer. The filters were proven to be robust to various errors in the signals and the model.

Chapter VI

CONTROL OF FLEXIBLE SERIAL MANIPULATORS

For rigid robot arms, the controller be can generated with feedback linearizion to cancel the robot's dynamics. The controllers for flexible robot arms are more complicated, because the controller also has to minimize oscillations. The most challenging problem in the design of controllers for flexible link manipulators is under actuation and non-minimum phase dynamics. Under actuation occurs because there are only a finite number of actuators to control the infinite degrees of freedom that arise from link flexibility.

A high-level block diagram of the proposed controller is shown in Figure 87. The trajectory generator gives the desired positions, velocities, and accelerations along the entire move based on solving the inverse kinematics problem. The command generator is an open-loop controller that modifies the original trajectories, such that significant flexure induced by the joint motion will be suppressed. The drawback of using this control element is that the command duration increases. The feedback controller forces the robot to follow the prescribed trajectory and reduces oscillations in the arm due to disturbances. The state observer estimates the flexible states for use in the feedback controller.

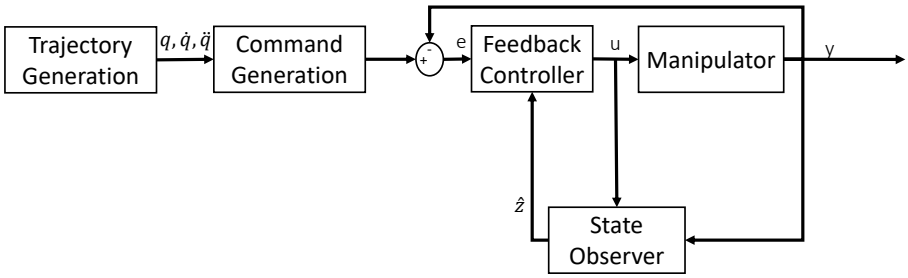


Figure 87: Controller Schematic

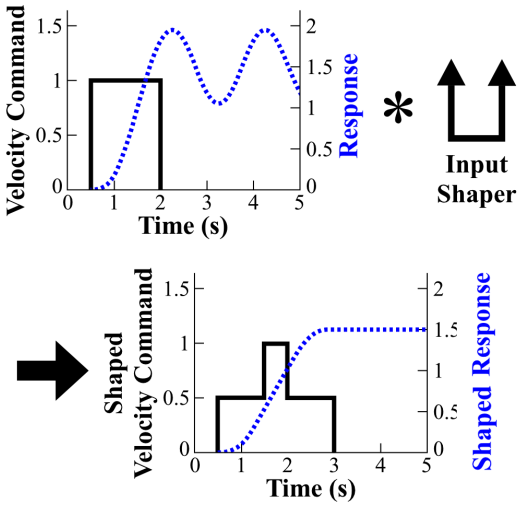


Figure 88: Input Shaping Process

6.1 Open Loop control of Flexible Serial Manipulators

Open loop control can play a vital role in the system's response since, if designed properly, the unwanted system dynamics can be reduced significantly. It has been shown that input shaping has the lowest rise time in addition to being one of the most robust open-loop techniques available. The lowest rise time is given priority since move speeds are one of the main reasons why one might want to use flexible robot arms instead of rigid alternatives.

6.1.1 Input Shaping

Input shaping convolves the baseline input command with a series of impulses at specific time intervals. The result is a shaped command that is a little slower than the original command, but greatly reduces residual vibration. This process is illustrated in Figure 88.

In order to determine the impulse amplitudes and time locations of an input shaper, certain design constraints must be satisfied. The primary design constraint is a limit on the amplitude of vibration caused by the shaper. The normalized, percentage residual vibration (PRV) amplitude of an under-damped, second-order system from a sequence of n-impulses is given by [121]:

$$PRV = V(\omega,\zeta) = e^{-\zeta\omega t_n}\sqrt{[C(\omega,\zeta)]^2 + [S(\omega,\zeta)]^2} \tag{187}$$

where,

$$C(\omega,\zeta) = \sum_{i=1}^{n} A_i e^{\zeta\omega t_i}\cos(\omega t_i\sqrt{1-\zeta^2}) \quad (188)$$

$$S(\omega,\zeta) = \sum_{i=1}^{n} A_i e^{\zeta\omega t_i}\sin(\omega t_i\sqrt{1-\zeta^2}) \quad (189)$$

and ω is the natural frequency of the system, ζ is the damping ratio, and A_i and t_i are the i^{th}-impulse amplitude and time, respectively.

Equation (187) gives the ratio of vibration with input shaping to that without input shaping. A constraint on residual vibration amplitude can be formed by setting (187) less than or equal to a tolerable level of residual vibration at the estimated natural frequency and damping ratio. For the simplest Zero Vibration (ZV) shaper, the tolerable amount of vibration is set to zero. This results in a shaper of the form [121, 125]:

$$ZV = \begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{1+K} & \frac{K}{1+K} \\ 0 & \frac{\pi}{\omega\sqrt{1-\zeta^2}} \end{bmatrix}, \quad (190)$$

where,

$$K = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}. \quad (191)$$

The sensitivity of the shaper that has been calculated for 1 Hz can be seen in Figure 89. Note that the response is not only zero at 1 Hz but also at its odd number multiples (3,5,7... Hz). The slope of the PRV curve near 1 Hz is relatively steep, which means that if the frequency of the system is not known very well, then the resulting response might be poor. However, if the frequency is known well, then the response will have minimal oscillations. The downside of convolving the baseline command with this shaper is that the command duration has increased by half the period of the unwanted frequency.

6.1.2 Multi-Mode Shaping

Because there can be several significant modes in a flexible robot arm, an input shaper designed for a single frequency is generally not enough to suppress the vibration. Input shapers designed for individual modes can be convolved together to create a multi-mode

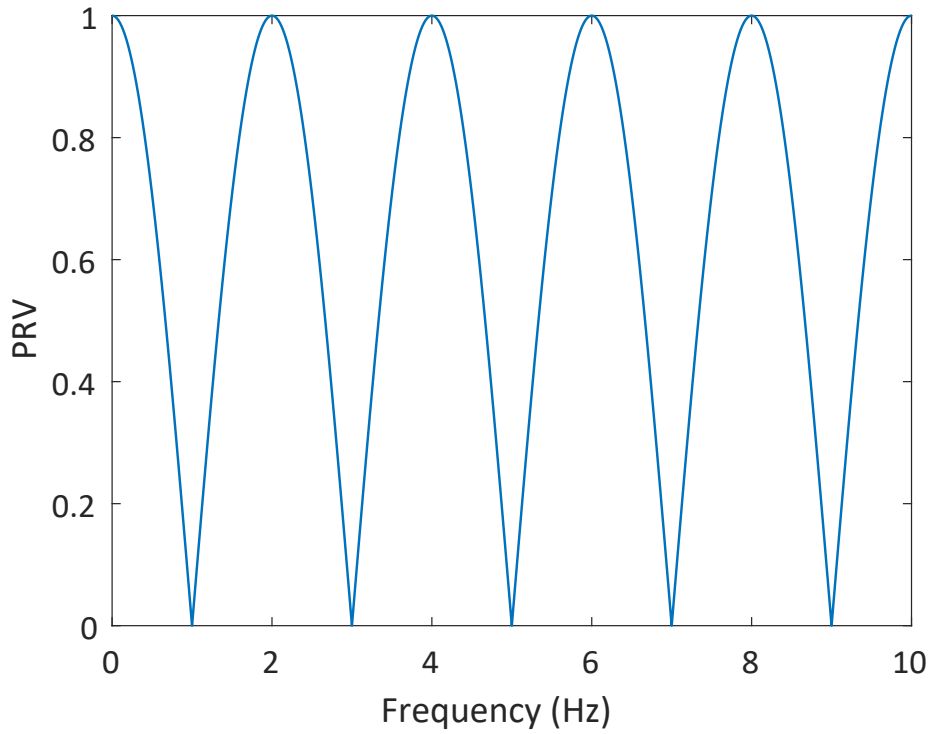


Figure 89: ZV Shaper PRV

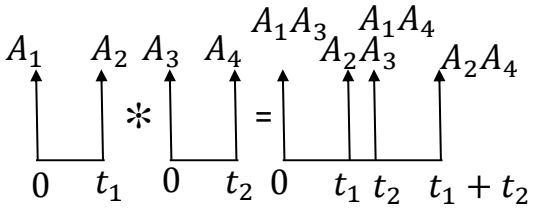


Figure 90: Convolving 2 ZV Shapers Together

input shaper. An illustration on how two ZV shapers are convolved together can be seen in Figure 90. Figure 91 shows the PRV graph of a two mode input shaper that was derived from a ZV shaper for $1Hz$ and a ZV shaper for 2.5 Hz. It can be seen that the vibration is exactly zero at those targeted frequencies. Note that the PRV curve is also generally lower than in Figure 89. The biggest downside of convolving 2 shapers together is that the shaper length is the sum of the the length of the individual shapers, and therefore, all commands will last longer.

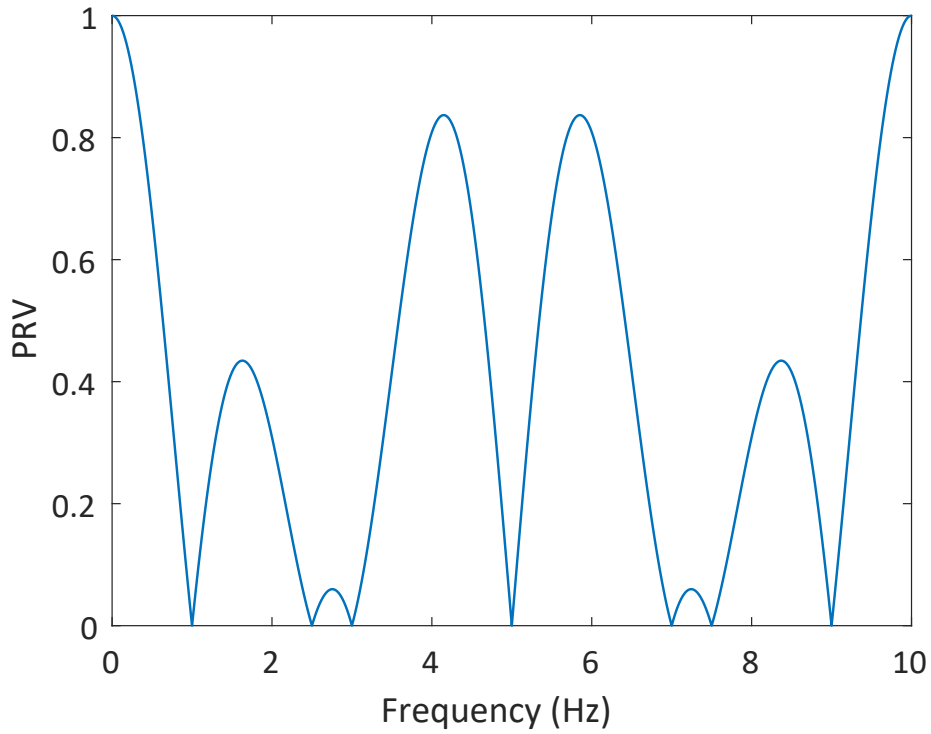


Figure 91: 2 mode ZV Shaper PRV

6.1.3 Specified Insensitivity Shaping

Although, robust expansions of the ZV shaper such as ZVD, EI, etc. exist, and they have been used in numerous instances where the natural frequencies of the system are not well known, they are not necessarily robust enough for large variations in natural frequency during a motion of a flexible robot arm. Directly convolving shapers together, as in the previous section, with the goal of having a PRV curve that has a low magnitude at all of the possible natural frequencies the robot arm might have, will most likely increase the rise time to unacceptable durations. This is a poor choice because one of the main reasons to use a flexible robot arm is the increased speed of motion.

A better approach would be to use a specified insensitivity (SI) shaper. The most straightforward method for generating a shaper with specified insensitivity to frequency errors is to use the frequency-sampling technique. This method requires repeated use of the vibration amplitude equation (187). In each case, $V(\omega, \zeta)$ is set less than or equal to a

tolerable level of vibration V_{tol}:

$$V_{tol} \leq e^{-\zeta\omega t_n}\sqrt{[C(\omega_s,\zeta)]^2 + [S(\omega_s,\zeta)]^2}, s = 1, ..., m, \tag{192}$$

where m is the number of points at which the vibration is limited. The other constraints for finding an SI shaper are:

- All of the amplitudes are positive ($A_i > 0$)
- The sum of all amplitudes in the shaper must equal 1 ($\sum_{i=1}^{n} A_1 = 1$)
- Minimize the time location of the last amplitude ($\min(t_n)$)

Note that although in general A_i can be negative, this approach has the potential drawbacks of exiting unmodeled high modes and saturating the actuators.

A more thorough derivation of SI shapers can be found in [124].

If there are multiple ranges of frequencies that must be suppressed then, (192) can be defined for each of those frequency ranges. Increasing m on each of those ranges increases the number of constraint equations that need to be solved for.

An algorithm for suppressing n frequency ranges with an SI shaper is:

1. Define frequency ranges $\omega_{i_{min}}$ - $\omega_{i_{max}}$, damping ζ_i, and tolerable amount of residual vibration V_{tol_i}, $i = 1, 2, ...n$
2. Define the insensitivity points I_i on each frequency range i where V_{tol_i} will be calculated
3. Crate an initial guess for the shaper amplitudes and their time locations. For example convolving together n EI shapers designed for the average frequency in range i ($\omega_{avg} = \frac{\omega_{i_{begin}} - \omega_{i_{end}}}{2}$). This gives in amplitudes A_j and time locations t_j, $j = 1, 2...max(j)$
4. Minimize $t_{max(j)}$ with constraint equations $V_{tol}(I_i) < V_{tol_i}$, $A_j > 0$, $t_j > 0$, $\sum_{j=1}^{max(j)} A_j = 1$, $t_0 = 0$
5. Remove all amplitudes $A_j < \epsilon_A$ and sum A_j and A_{j+1} if $(t_j - t_{j+1}) < \epsilon_t$. Go to step 4 with this new initial guess if any any changes in impulses, end algorithm otherwise.

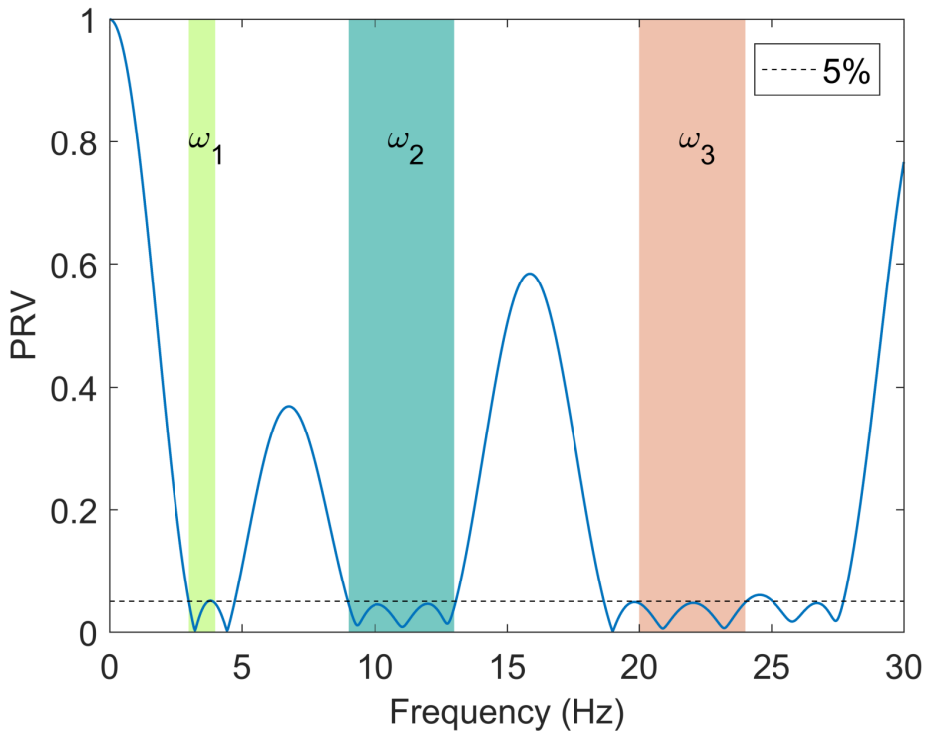


Figure 92: SI Shaper for 3 Frequency Ranges

In the algorithm, A_j and t_j are the amplitudes and time locations of impulse j, with 1 = 1, 2...; $\omega_{i_{min}}$ is the start of frequency range i, $\omega_{i_{max}}$ is the end of frequency range i; ζ is the damping of the i^{th} frequency range; ϵ_A is the lowest impulse amplitude allowed and ϵ_t is the shortest time duration allowed between the impulses.

Figure 92 shown the PRV curve for a SI shaper designed for frequency ranges of:

- $\omega_{1_{min}} = 3$, $\omega_{1_{max}} = 4$, $\zeta_1 = 0$, $V_{tol_1} = 5\%$
- $\omega_{2_{min}} = 9$, $\omega_{2_{max}} = 13$, $\zeta_2 = 0$, $V_{tol_2} = 5\%$
- $\omega_{3_{min}} = 20$, $\omega_{2_{max}} = 24$, $\zeta_3 = 0$, $V_{tol_3} = 5\%$

It can be observed that the PRV value is under the 5% line in those frequency ranges and variable elsewhere. The duration of the shaper is 0.3192 s. For comparison, an ZV shaper designed for 3.5 Hz has a duration of 0.1429 s. A more robust EI shaper designed for 3 Hz has a duration of 0.286 s. Designing EI shapers for the midpoints of the frequency ranges results in a shaper of with a PRV Curve that can be seen in Figure 93. The EI shaper for 3

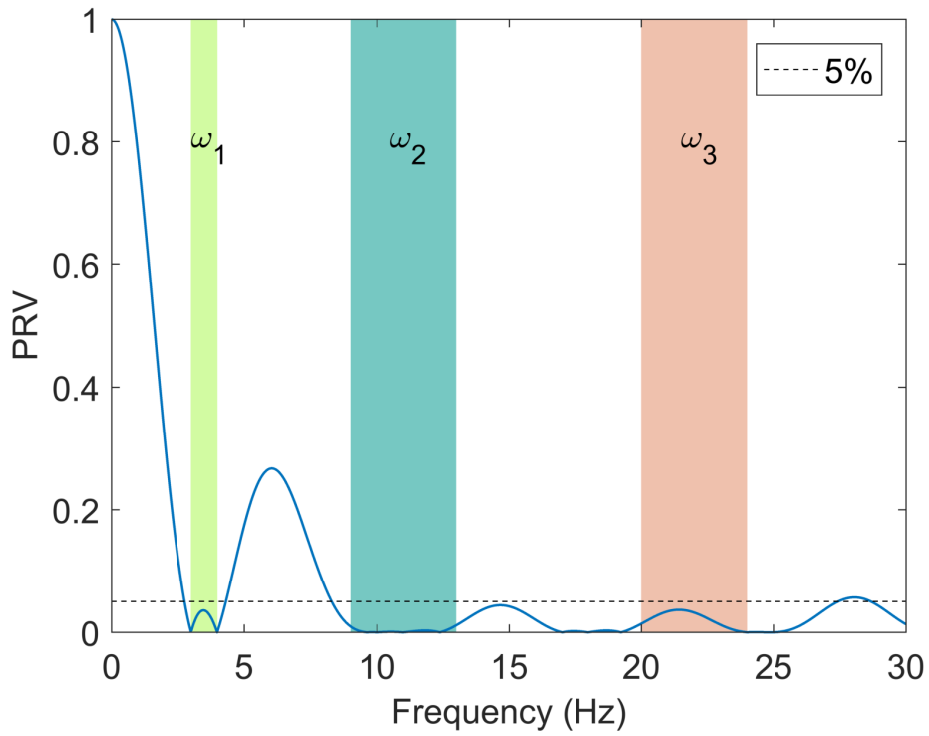


Figure 93: EI Shaper for 3 Frequency Ranges

modes generally has a PRV magnitude that is lower that the one for the SI shaper, however the duration of the shaper is 0.422 s, that is 38% longer than the SI shaper. Furthermore the EI shaper does not guarantee that the entire frequency range is under the required V_{tol} line.

6.1.4 Optimal SI Shaping For Flexible Arms

In the previous section a routine was presented to create an SI shaper that effectively cancels out vibration from all of the significant modes. How to optimally choose the V_{tol} levels is discussed next.

In Chapter 3 the equations of motion governing the flexible robot were:

$$\begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & \mathbb{I} \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} C_r(q,\dot{q},\dot{\delta}) \\ C_f(q,\dot{q},\dot{\delta}) \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & K_{\delta\delta}(q) \end{pmatrix} \begin{pmatrix} q \\ \delta \end{pmatrix} = \begin{pmatrix} \tau \\ 0 \end{pmatrix} \tag{193}$$

It can be seen that the joint states q are coupled to the flexible states, δ, through matrices M, and C. Matrix C affects the vibrations non-linearly and input-shaping will not directly

be able to cancel out the contributions from that matrix because the method relies on the super-position principle holding. In addition, in numerous flexible robot experiments, Coriolis and centrifugal forces have been neglected from the equations of motion since they do not have a large effect [46,76]. Writing out the second row of (193) shows how the flexible and rigid states are coupled:

$$\ddot{\delta} + K_{\delta\delta}(q)\delta = -M_{q\delta}^{T}(q)\ddot{q} \tag{194}$$

Therefore, $M_{q\delta}^{T}(q)$ determines which modes get excited what their contribution are overall response.

If the desired trajectory, q_d is known before hand then (194) can be used to determine the V_{tol} levels based on $-M_{q\delta}^{T}(q)\ddot{q}$. Furthermore the deflection of a point p at time t is calculated as:

$$w(p,t) = \sum_{j=1}^{m} w_j(p)\delta_j(t), \tag{195}$$

where m is the number of considered modes, $w_j(p)$ is the displacement of mode j at point p, and δ_j is the j^{th} modal amplitude. However note that, as described in Chapter 3, the mode shapes w_j for flexible manipulators change with the configuration q. The direct method for finding the contributions of each mode at the end of the motion would be to integrate (194) and evaluate the amplitudes of δ at the end of the trajectory. Such a calculation, however, can be cumbersome to perform. An approximation for this method would be to simply calculate:

$$r = \int_{t_0}^{t_f} |M_{q\delta}^{T}(q_d(t))||\ddot{q}_d(t)|dt \cdot w_j^i(p) \tag{196}$$

where r is the column vector of relative contributions from each mode; t_0 is the start time of the trajectory, t_f is the end time of the trajectory; q_d is the desired trajectory; i is the link where point p lies, and

$$w_j^i(p) = \sqrt{\left(w_{j_x}^i(p)\right)^2 + \left(w_{j_y}^i(p)\right)^2 + \left(w_{j_z}^i(p)\right)^2}_{|q_f} \tag{197}$$

is the total modal displacement evaluated at the end of the desired trajectory q_f. Assuming that mode j has the highest value in the vector r, then V_{tol_r} is set to V, the V_{tol} values for the rest of the modes s can be assigned based on the relative difference in r:

$$V_{tol_s} = \frac{r_j}{r_s}V, \; V_{tol_s} <= 1 \tag{198}$$

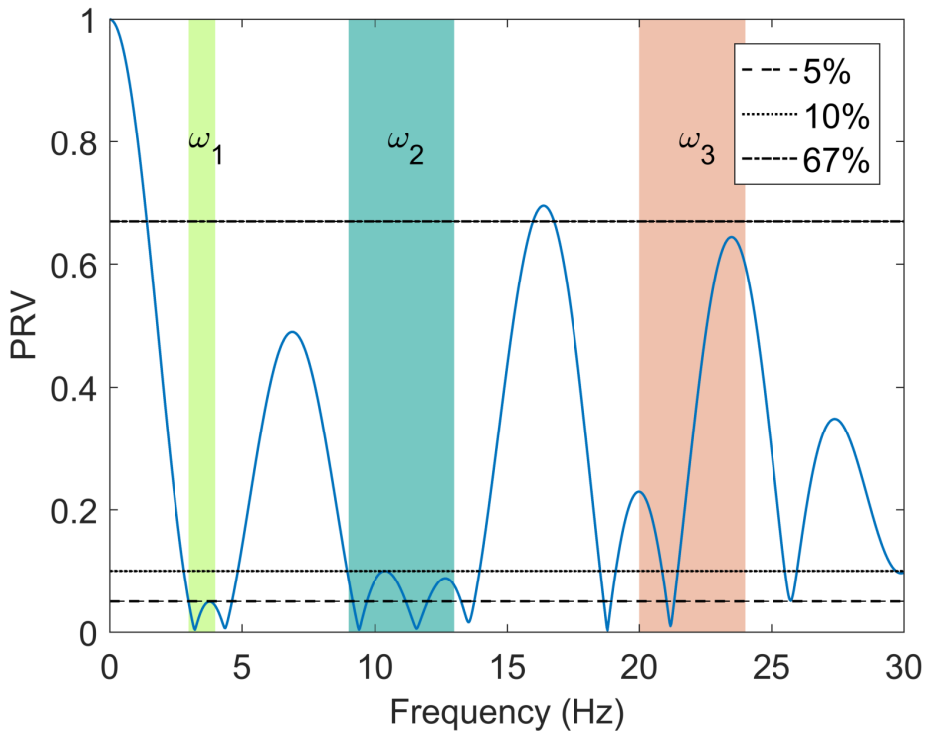


Figure 94: Optimized SI Shaper for 3 Frequency Ranges

When $r_s = r_j$, then $V_{tol_s} = V_{tol_j}$. If $r_s \to 0$ then $V_{tol_s} = 1$.

Using the example from the previous section and assuming that the relative contributions for a trajectory are $r = [4\ 2\ 0.3]^T$. Then, $V_{tol_1} = 0.05$, $V_{tol_2} = 0.05$, $V_{tol_3} = 0.67$. The corresponding shaper will have a duration of 0.30 s. This is 6% shorter than the shaper without the optimal V_{tol} values and is only 5% longer than an EI shaper designed for 3.5 Hz, which is an optimal shaper for that frequency when only positive impulses are used. Figure 94 shows the resulting PRV curve of the optimized SI shaper. Notice that the overall PRV curve has shifted upwards in most areas but is still lower than V_{tol} at the corresponding frequency ranges. Although, it is well known that the higher modes have a lower effect on the magnitude of the oscillations than the lower modes, it is worth noting that in real systems damping will also play a role. The curve of $e^{-\zeta\omega_n t}$ is the envelope function under which the oscillations occur, as illustrated in Figure 95.

For example, assuming that mode 3 has a damping ratio of $\zeta = 0.05$, the contribution from that mode would damp down to 5% of its original amplitude in 0.73 seconds. That is

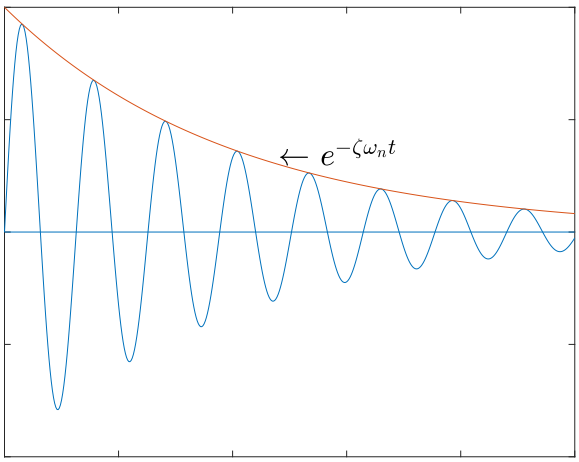


Figure 95: Free Response of an Under Damped Second Order System

about 3 periods of Mode 1 in this example. Therefore, even if the higher modes are excited they will damp down very fast and having a high V_{tol} fir mode 3 is appropriate.

Although the previous example only gave a 6% reduction in shaper length it's true value of the technique will become evident when it is determined that a low mode is not excited during the motion. Assigning V_{tol_1} = 0.99, V_{tol_2} = 0.05, V_{tol_1} = 0.05 yields a shaper with a duration of 0.11 s that is 64% shorter than if we had included a limitation on mode 1.

6.1.5 Case Study

In order to illustrate the effectiveness of the input-shaping control algorithm, experiments in both simulation and real hardware were conducted.

6.1.5.1 Simulations

The motions and the manipulator parameters for the simulation study are the same ones as used in Section 3.3.7.1. The natural frequencies of the system are shown in Figure 96. Only the first three modes are taken into consideration.

During move 1, all the joints move from 0 to -57° (-1 *rad*). The frequencies during the motion change in the range shown in Table 14. Calculating the relative deflection contributions of each mode using (196) yields the normalized (with respect to the highest contribution) r values of [0.647 1 0.13]. This means that the second mode has the biggest

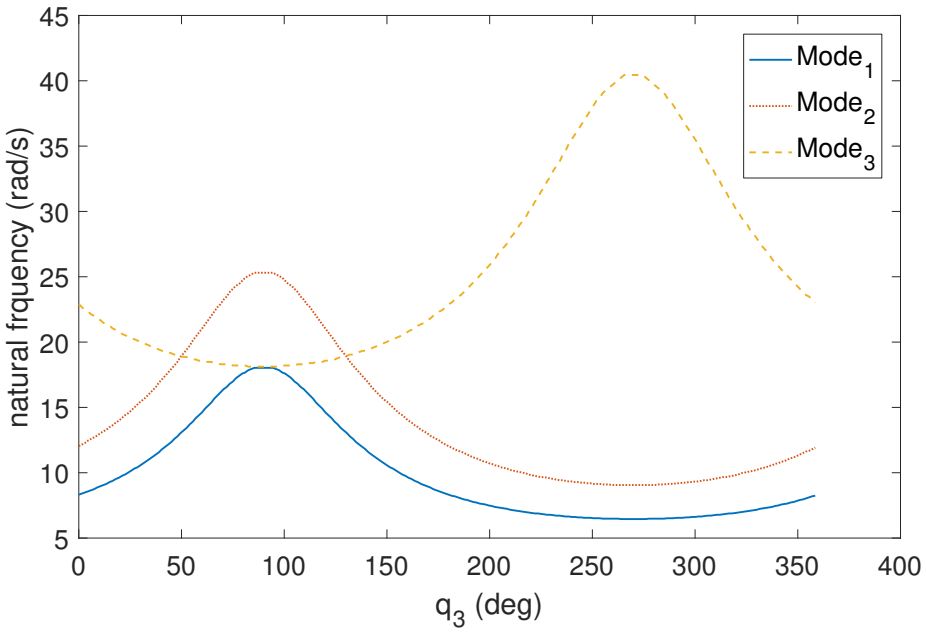


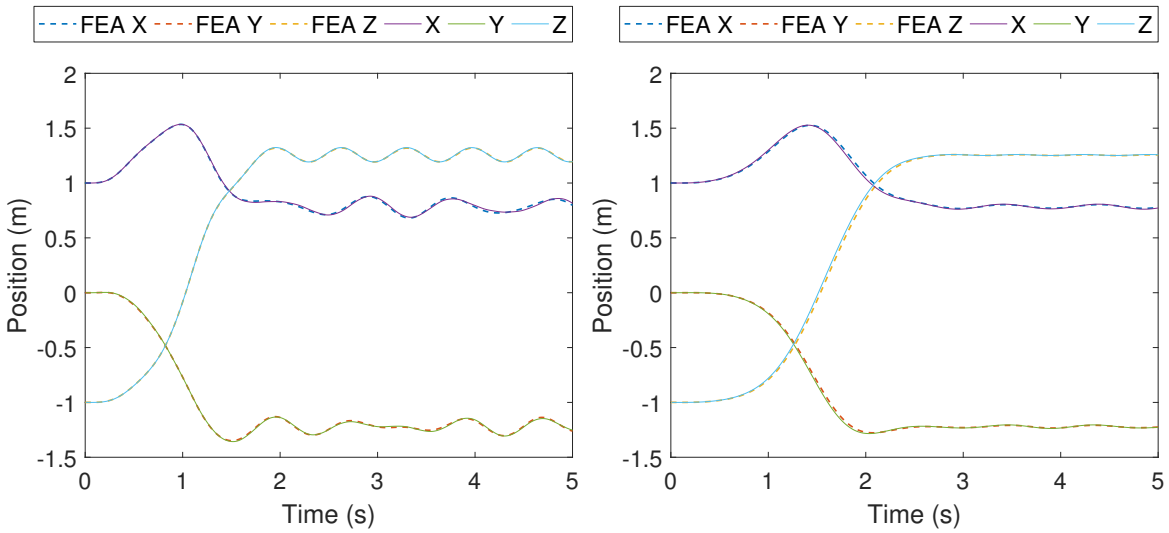
Figure 96: Natural Frequencies of the Robot Arm vs q_3

Table 14: Frequency ranges for Moves 1 and 2

	min	max
ω_1	6.64	8.26
ω_2	9.36	11.9
ω_3	23.36	35.21

influence in the response, the first mode has a slightly lower significance, and the third mode has the lowest significance Therefore based on (198) V_{tol_1} = 0.078, V_{tol_2} = 0.05, V_{tol_3} = 0.36. The shaper duration is 0.87 s. Figure 97 shows the comparison between the shaped and unshaped responses for move 1. In Figure 97a the unshaped response is shown and it shows that the tip of the arm oscillates in every direction. Figure 97b shows a much better response where the oscillations have almost completely been canceled due to input shaping. A small amount of residual oscillation exists in the response due to non-linear effects and the non-zero value of V_{tol} that were used to design the shaper.

During move 2, joints 2 and 3 move from 0 to -57° (-1 *rad*) and joint 1 is stationary, so the motion is fully planar. Because the trajectory for q_3 is the same as for move 1 the natural frequencies are the same as well and can be seen in Table 14. However, during this

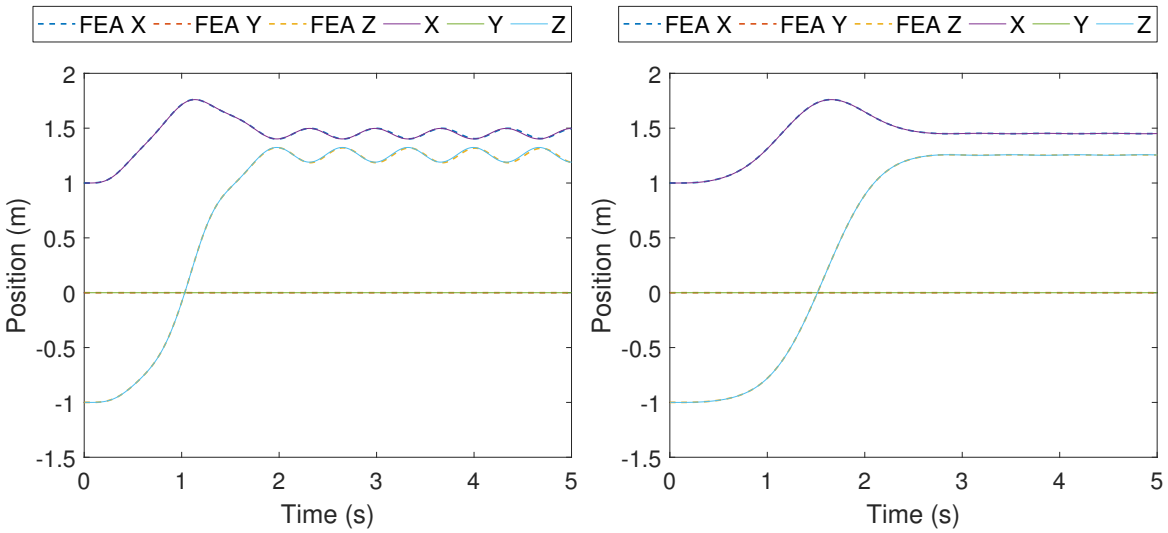


(a) Unshaped (b) Shaped

Figure 97: Move 1 Tip Position

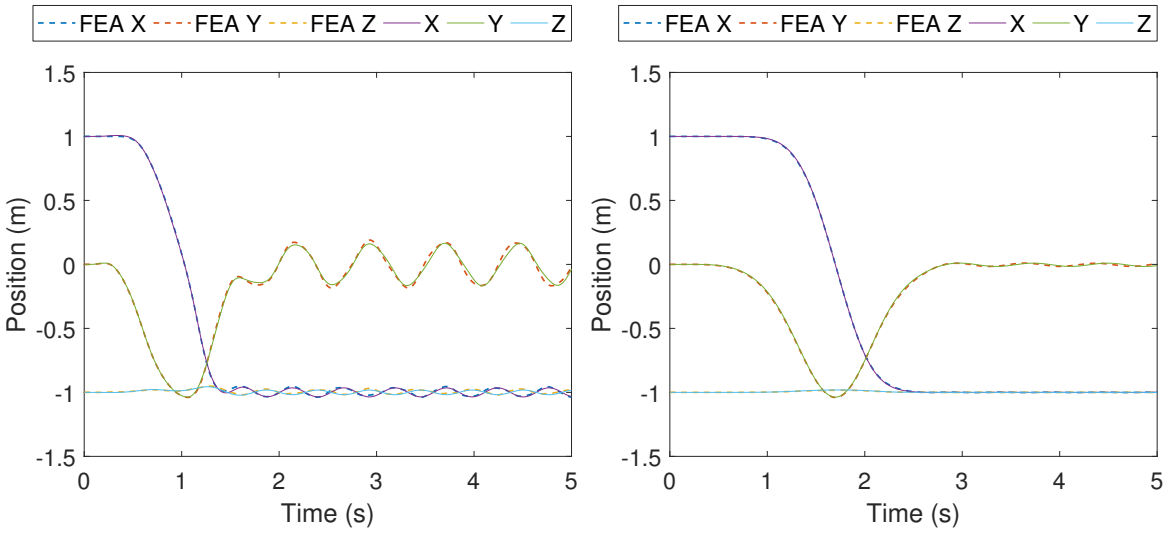
motion the relative deflection contributions are r=[0 1 0]. Therefore, V_{tol_1} = 1, V_{tol_2} = 0.05, V_{tol_3} = 1. The shaper duration is 0.53 s, which is 40% shorter than the one used for move 1. Figure 98 shows the comparison between the shaped and unshaped responses for move 2. Figure 98a shows the unshaped response and it can be seen that the tip of the arm oscillates in the X and Y directions. Figure 98b shows an improved response where almost all of the residual vibrations has been eliminated. Since this motion was planar, the non-linear effects have not influenced the motion much and the response is cleaner compared to move 1.

During move 3, joint q1 moves from 0 to -180°. The natural frequencies do not change during this motion because q_3 is stationary. The natural frequencies for this motion are ω_1 = 8.26 ω_2 = 11.93, ω_3 = 23.37, rad/s, however it is a good idea to add a range (for example ±20%) around those frequencies to the shaper calculation, since the model will always have errors. The relative contributions from the modes are r=[1 0 0.38]. Therefore, V_{tol_1} = 0.05, V_{tol_2} = 1, V_{tol_3} = 0.13. The duration of the shaper is 0.39 s, which is 55% shorter than during move 1 and 27% shorter than during move 2. Figure 99 shows the comparison between the shaped and unshaped responses for move 3. Figure 99a shows the unshaped response where the tip oscillates mainly in the Y direction. Figure 99b shows the shaped



(a) Unshaped (b) Shaped

Figure 98: Move 2 Tip Position



(a) Unshaped (b) Shaped

Figure 99: Move 3 Tip Position

response and where most of the residual vibration has been canceled out.

6.1.5.2 Experiments

Input shaping was applied to the test bed described in Chapter 4. The motions used in section are the exact same ones used to validate the test bed.

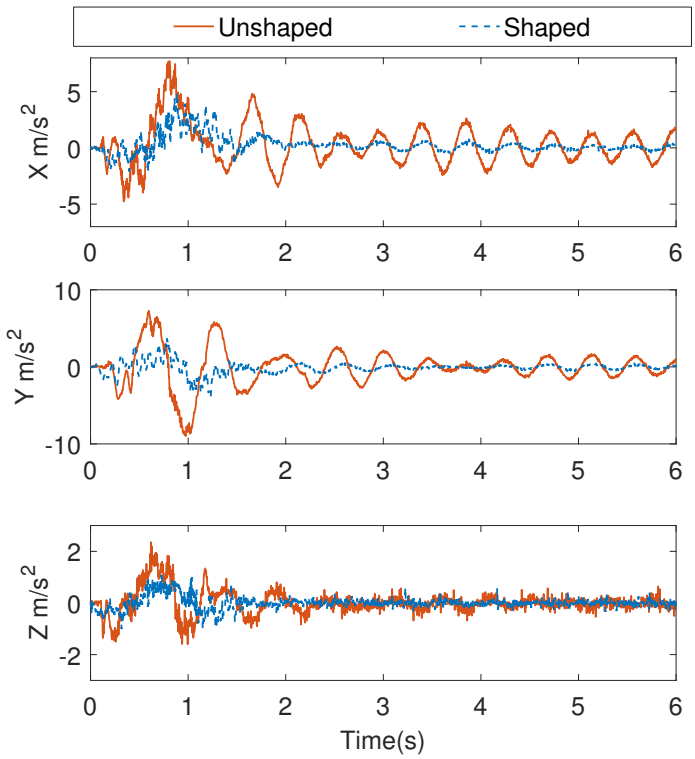


Figure 100: Unshaped vs Shaped Response For Move 1

During move 1, all joints are given a bang-bang acceleration refernce command resulting in a change of 1 *rad* for each joint. Joint q_1 moves from 0 to -1 *rad*, joint q_2 moves from 1 to 2 *rad*, and joint q_3 moves from 5.2 to 4.2 *rad*. The parameters to calculate the input shaper are given in Table 15. Robustness of ±20% was given to each frequency range.

Table 15: Frequency Ranges for Move 1

	min	max	V_{tol}
ω_1	1.9	1.95	0.05
ω_2	2.5	2.5	0.25
ω_3	9.0	11.2	0.74

Figure 100 shows the accelerometer readings from both the shaped and the unshaped cases. It is clear that input shaping has significantly reduced the vibration amplitude in both the transient and residual motion phases.

Utilizing the estimator presented in Chapter 5, the position of the tip can be estimated

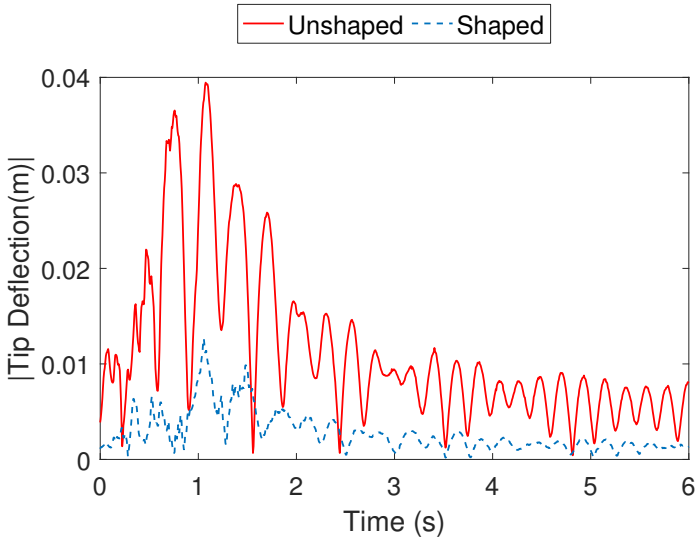


Figure 101: Shaped vs Unshaped Tip Deflection Move 1

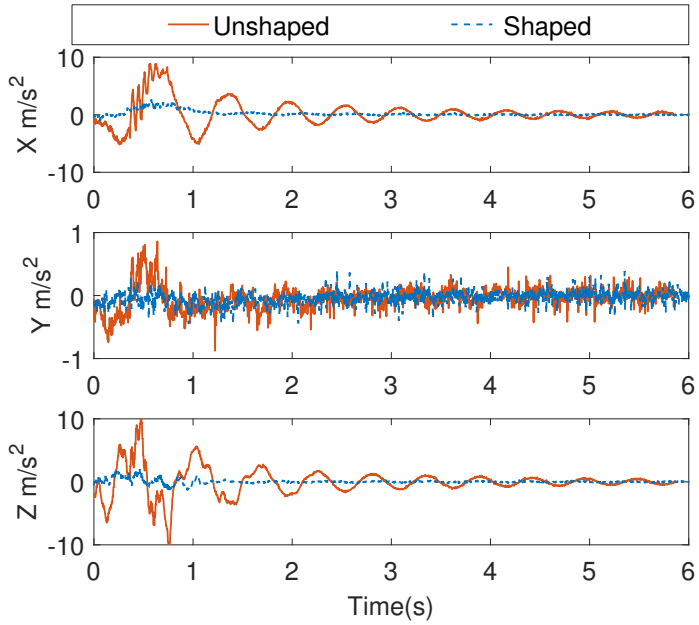


Figure 102: Unshaped vs Shaped Response For Move 2

based on the mode shapes and modal amplitudes during the motion. Figure 101 shows the magnitude in all spatial directions summed ($mag = \sqrt{(x^2 + y^2 + z^2)}$) of the tip oscillation. During the transient phase the input shaper reduced the vibration by 75% and about 80% during the residual period.

Move 2 is a planar move where joint $q1$ remains stationary, joint q_2 moves from 1 to 2

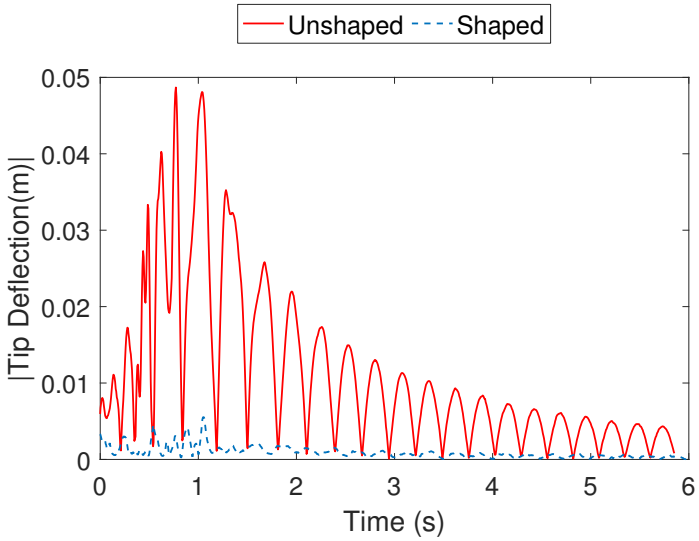


Figure 103: Shaped vs Unshaped Tip Deflection Move 2

rad, and joint q_3 moves from 6.28 to 5.28 *rad*. The parameters to calculate the input shaper are given in Table 16. Robustness of ±20% was given to each frequency range.

Table 16: Frequency Ranges for Moves 2

	min	max	V_{tol}
ω_1	1.93	1.95	0.05
ω_2	2.45	2.5	1
ω_3	9.0	11.2	1

Figure 102 shows the accelerometer readings from both the shaped and the unshaped cases. It can be seen that in the X axis the shaped response has a much lower amplitude than the unshaped one. The magnitude in the Y axis is near the sensor noise level for both cases. The acceleration in the Z axis is much lower with input shaping. Figure 103 shows the magnitude of the tip oscillation for both the shaped and the unshaped cases. During the transient phase the input shaper reduced the vibration by about 90% and about 85% during the residual period.

During move 3 joints, q_2 and q_3 are held stationary and joint q_1 moves from 0 to -1 *rad*. The parameters to calculate the input shaper are given in Table 17. Robustness of

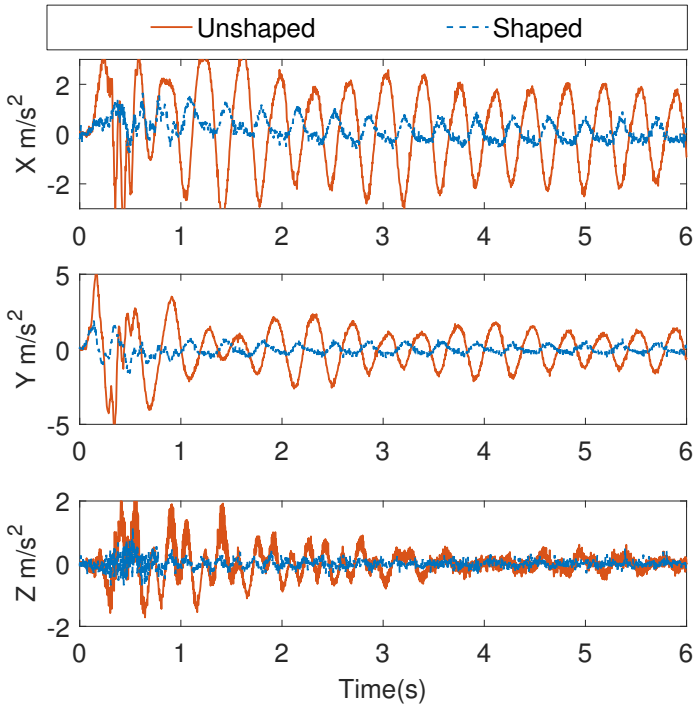


Figure 104: Unshaped vs Shaped Response For Move 3

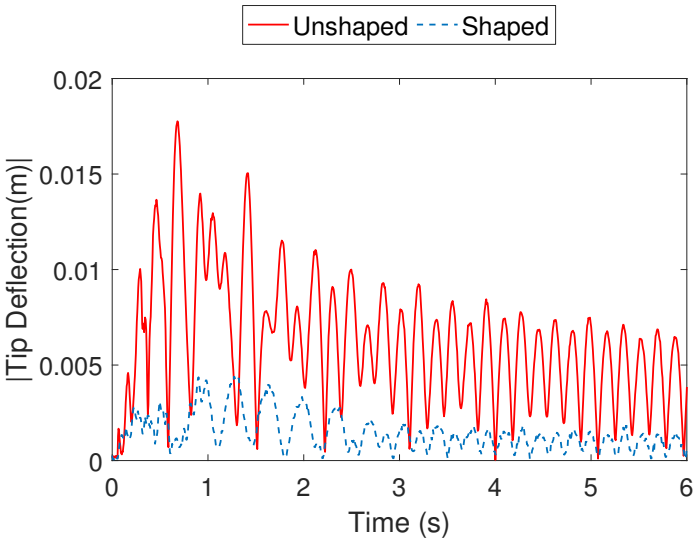


Figure 105: Shaped vs Unshaped Tip Deflection Move 3

±20% was given to each frequency range. Figure 104 shows the accelerometer readings from both the shaped and the unshaped cases. As described in Section 4.5.2, the accelerometer readings obtained from this move did not match well with the simulated one. It can be seen that input shaping is not as successful at canceling oscillations as with moves 1 and

Table 17: Frequency Ranges for Moves 3

	min	max	V_{tol}
ω_1	1.93	1.93	1
ω_2	2.5	2.5	0.05
ω_3	9.5	9.5	0.16

2. However, the magnitude of the accelerations in each axis has been lowered significantly. Figure 105 shows the magnitude of the tip oscillation for both the shaped and the unshaped cases. During the transient phase the input shaper reduced the vibration by about 70% and about 60% during the residual period.

6.1.6 Summary Of Open Loop Control

In this section open an open loop controller based on input shaping was presented. The design of the input shaper intelligently takes into consideration what frequency ranges are significant during the planned trajectory and reduces both transient and residual oscillations. This method was proved to be effective both in a simulation study with comparisons to FEA models, and in experiments.

6.2 Feedback Control of Flexible Serial Manipulators

The previous section illustrated that if the model is known relatively well, and there are no disturbances, then open-loop control can be used to effectively move a flexible arm from one location to the next without much residual vibration. In the presence of larger model error and/or disturbances feedback control is required to damp out the oscillations in the arm.

6.2.1 Modal controllability

Before control algorithms can be presented, the controllability for flexible robot arms must be discussed. The equations of motion that were derived in Chapter 3 are:

$$\begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & \mathbb{I} \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} C_{qq}(q,\dot{q},\dot{\delta}) & C_{q\delta}(q,\dot{q},\dot{\delta}) \\ C_{\delta q}(q,\dot{q},\dot{\delta}) & C_{\delta\delta}(q,\dot{q},\dot{\delta}) \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} g_r \\ g_f \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & K \end{pmatrix} \begin{pmatrix} q \\ \delta \end{pmatrix} = \begin{pmatrix} \tau \\ 0 \end{pmatrix} \quad (199)$$

Writing out the second row yields the differential equation governing the flexible variables δ:

$$\ddot{\delta} + C_{\delta\delta}(q,\dot{\delta})\dot{\delta} + C_{\delta q}(q,\dot{\delta})\dot{q} + K(q)\delta + g_{\delta}(q) = -M_{q\delta}(q)\ddot{q} \tag{200}$$

Without the loss of relevance the $C_{\delta\delta}$ and $C_{\delta q}$ terms that contain the centrifugal and Coriolis terms, and g containing the gravity terms are excluded from the controllability analysis. The centrifugal and Coriolis terms only affect the system when relatively high velocities are present in the system. Additionally, as many other researchers have noted, those terms do not have a large effect on the response of the system [46,76]. Therefore this section presents a conservative estimate of controllability.

From (200) it can be seen that the flexible states (δ), and the rigid states q are coupled through matrix $M_{q\delta}(q)$. The state space form of (200) is:

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & \mathbb{I} \\ -K(q) & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 \\ -M_{q\delta} \end{bmatrix} \ddot{q} \tag{201}$$

In short form, (201) is:

$$\dot{x} = A(q)x + B(q)u, \tag{202}$$

where A is of dimension $2m \times 2m$ and B is of dimension $2m \times n$; m is the number of modes considered, and n is the number of joints. A and B are configuration dependent and therefore, the controllability is configuration dependent as well. The controllability matrix G_c is defined as

$$G_c(q) = \left[B, AB \cdots A^{2m-1}B \right] \tag{203}$$

If $rank(G_c(q)) < 2m$ then at least one mode is not controllable.

Unfortunately, the rank condition does not give any information about which modes are not controllable. However, because $K(q)$ is always diagonal, the entries in $M_{q\delta}$ can be evaluated to physically interpret the "accessibility" of the modes. The accessibility of a mode i depends on the elements in row i of $M_{q\delta}(q)$. If all the elements are zero, then none of the joints can have any effect on mode i. The larger the relative magnitude of the elements in $M_{q\delta}$, the more accessible the mode is. Therefore, system is fully controllable at

configuration q if the $M_{q\delta}(q)$ has non-zero entries on each row. The accessibility index [77] of mode i at configuration q can be defined as a norm of the i^{th} row of $M_{q\delta}(q)$:

$$a_i(q) = sqrt\left(M_{q\delta_i}(q) \cdot (M_{q\delta_i}(q))^T\right), \tag{204}$$

where, $M_{q\delta_i}(q)$ is the i^{th} row of matrix $M_{q\delta}(q)$. Therefore the accessibility index serves as a configuration-dependent measure of controllability for the modes.

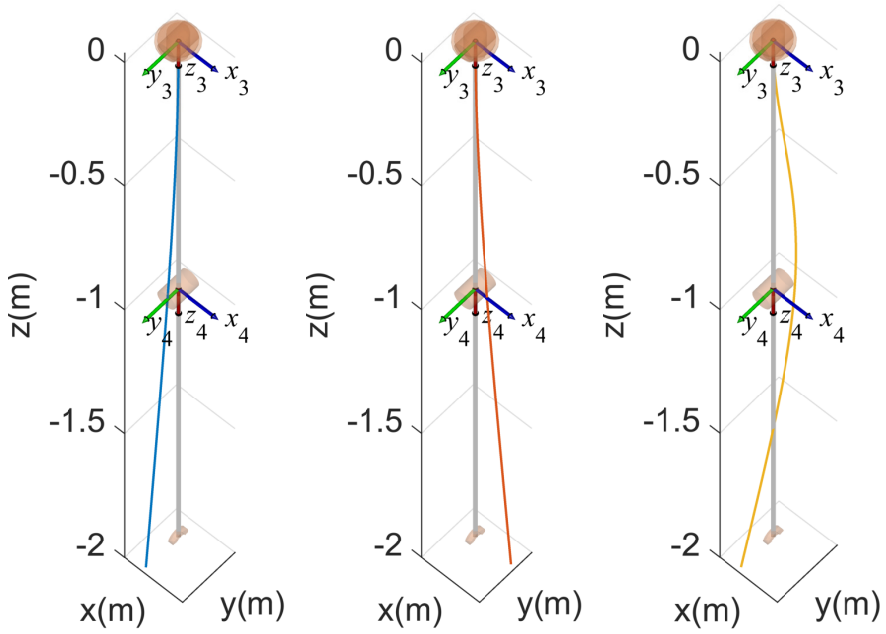
6.2.1.1 Case Study

Consider the 2 link robot model presented in Section 3.3.7. To investigate the controllability of the modes, the accessibility indexes are needed. To physically understand the indexes we need to look at the mode shapes of the system and how they change with the change of the configuration. Figure 106 shows the arm at a vertical configuration with $q_2 = 90°$ and $q_3 = 270°$. It can be seen that mode 1, shown in Figure 106a, has displacements in the direction of the joints (i.e perpendicular to the motion they cause). Mode 2, shown in Figure 106b, is in the direction of the joint motion. Mode 3, shown in Figure 106c, is in the same direction as mode 1, so it should also not be affected by any motion of the joints.

Figure 107 shows the mode shapes in the configuration $q_2 = 90°$ and $q_3 = 225°$. It can be seen that the overall shape of the modes has not changed greatly, however, mode 1, shown in Figure 107a, and mode 3, shown in Figure 107c, should now be more susceptible to changes in joint q_1.

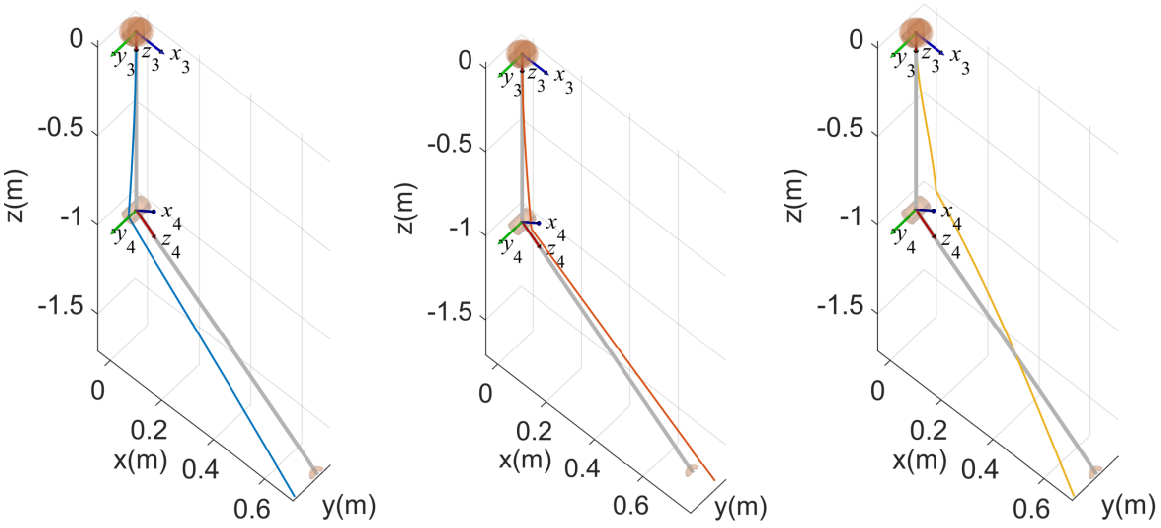
The magnitude of q_1 does not influence the accessibility index for any of the modes. Figure 108 shows the accessibility index for the first mode. It can be seen that the accessibility of the first mode is largely dependent on q_2. When q_2 is near 90° the first link of the arm is directly under q_1 and, therefore, it cannot have an effect on it. When $q_2 = 90°$ and $q_3 = 270°$ the mode is not accessible. As can be seen in Figure 107a the mode is exactly orthogonal to the motion of all of the joints. When q_2 changes from 90°, the mode shape becomes accessible by joint q_1, as is illustrated in Figure 107a.

Figure 109 shows the accessibility index for mode 2. It can be seen that q_3 has a small



(a) Mode 1 (b) Mode 2 (c) Mode 3

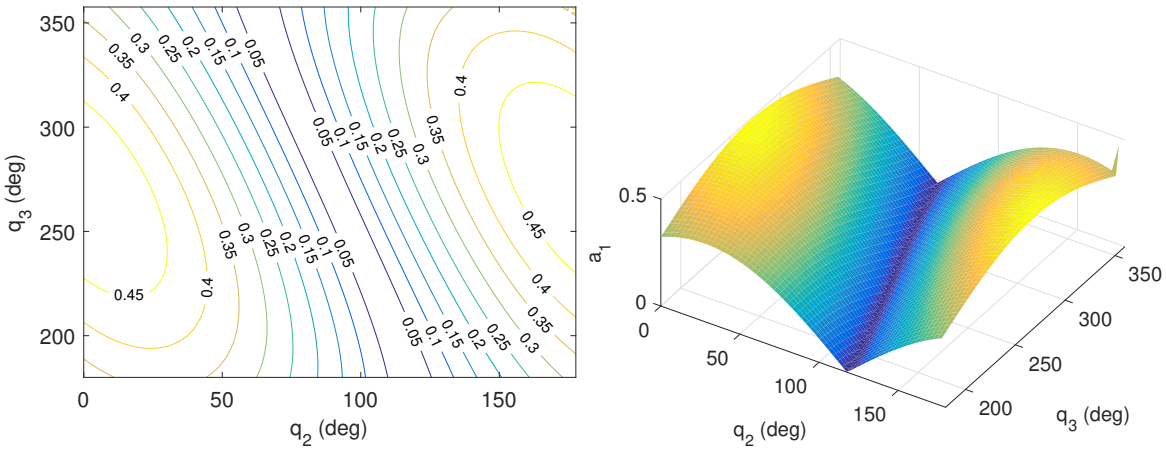
Figure 106: First Three Modes at $q_2 = 90°$, $q_3 = 270°$



(a) Mode 1 (b) Mode 2 (c) Mode 3

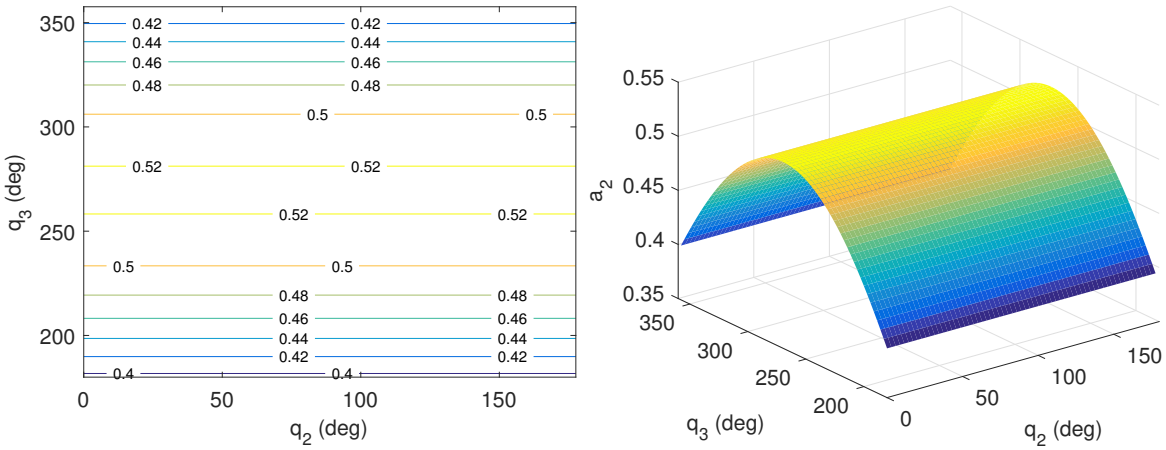
Figure 107: First Three Modes at $q_2 = 90°$, $q_3 = 225°$

effect on accessibility, and q_2 has no effect. Therefore, the mode can be controlled throughout the workspace. This makes physical sense since this mode is in the direction of joint q_2 and q_3, as was seen in Figures 106c and 107c.



(a) Countour Plot of Accessibility Index a_1 (b) Surface Plot of Accessibility Index a_1

Figure 108: Accessibility Index a_1

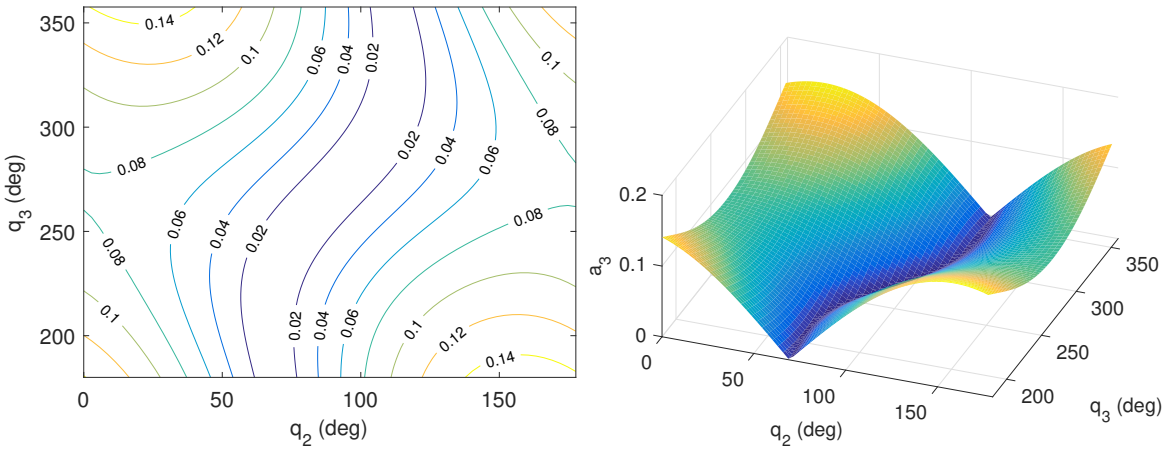


(a) Countour Plot of Accessibility Index a_2 (b) Surface Plot of Accessibility Index a_2

Figure 109: Accessibility Index a_2

Figure 110 shows the accessibility index for mode 3. The accessibility of the mode is similar to mode 1. The accessibility is mostly dependent on q_2. When $q_2 = 90°$ and $q_3 = 270°$, the mode is not accessible. As can be seen in Figure 107c, the mode is exactly orthogonal to the motion of all of the joints. When q_2 changes from 90°, the mode shape is accessible by joint q_1, as was illustrated in Figure 107c.

Therefore, the trajectory of the flexible robot must be carefully chosen such that at least the end-configuration allows for vibration controllability. Additionally, the controllers could leave the uncontrollable configurations to ones where the excited modes are accessible



(a) Countour Plot of Accessibility Index a_3 (b) Surface Plot of Accessibility Index a_3

Figure 110: Accessibility Index a_3

and then return to the uncontrollable configurations using a open loop controller so that additional vibrations would not be created.

6.2.2 Inversion Based Control

The feed-forward and computed torque methods used to control traditional serial manipulator arms are not necessarily guaranteed to be stable along arbitrary trajectories. The aim of this section is to provide a stable controller for joint trajectory tracking. This controller will not focus on damping out the oscillations in the arm. However, using this type of controller, in conjunction with an open-loop controller such as input-shaping can significantly reduce the transient and residual flexure in the arm, while guaranteeing stability of the joint trajectories.

Trajectory tracking for multi-input/multi-output nonlinear systems is most often done with input-output inversion control techniques. The premise of the method is that based on the output of the system, a nonlinear state feedback controller is designed so that the resulting closed-loop controller transforms the system into a linear one with a nonlinear feed-forward term. The stability of the system is determined based by stability of the unobservable, also known as, zero-dynamics. Recall from Chapter 3 that the equations of

motion of a flexible robot arm are:

$$\begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & \mathbb{I} \end{pmatrix}\begin{pmatrix} \ddot{q} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} C_{qq}(q,\dot{q},\dot{\delta}) & C_{q\delta}(q,\dot{q},\dot{\delta}) \\ C_{\delta q}(q,\dot{q},\dot{\delta}) & C_{\delta\delta}(q,\dot{q},\dot{\delta}) \end{pmatrix}\begin{pmatrix} \dot{q} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & K(q) \end{pmatrix}\begin{pmatrix} q \\ \delta \end{pmatrix} + \begin{pmatrix} g_r(q) \\ q_\delta(q) \end{pmatrix} = \begin{pmatrix} \tau \\ 0 \end{pmatrix} \tag{205}$$

From the second row the "flexible" subsystem is:

$$\ddot{\delta} = -C_{\delta q}(q,\dot{q},\dot{\delta})\dot{q} - C_{\delta\delta}(q,\dot{q},\dot{\delta})\dot{\delta} - K(q)\delta - M_{q\delta}^T(q)\ddot{q} \tag{206}$$

Plugging (206) in to the top row of (205) yields the following relation:

$$M_{qq}\ddot{q} + M_{q\delta}(-C_{\delta q}\dot{q} - C_{\delta\delta}\dot{\delta} - K\delta - M_{q\delta}^T\ddot{q}) + C_{qq}\dot{q} + C_{q\delta}\dot{\delta} = \tau \tag{207}$$

Grouping terms leads to:

$$(M_{qq} - M_{q\delta}M_{q\delta}^T)\ddot{q} + C_{qq}\dot{q} + C_{q\delta}\dot{\delta} - M_{q\delta}(C_{\delta q}\dot{q} + C_{\delta\delta}\dot{\delta} + K\delta) = \tau \tag{208}$$

It can be shown that $(M_{qq} - M_{q\delta}M_{q\delta}^T)$ has full rank by using the fact that the full inertia matrix has full rank and using Sylvester's rank equality property of the right side of the following expression:

$$\begin{bmatrix} M_{qq} & M_{q\delta} \\ M_{q\delta}^T & \mathbb{I} \end{bmatrix}\begin{bmatrix} \mathbb{I} & 0 \\ -M_{q\delta}^T & \mathbb{I} \end{bmatrix} = \begin{bmatrix} M_{qq} - M_{q\delta}M_{q\delta}^T & M_{q\delta} \\ 0 & \mathbb{I} \end{bmatrix} \tag{209}$$

Input-Output linearization works by differentiating a desired output y until the input u directly appears in the equations. Per (208), the input torques τ are already on the same differential level as the joint accelerations $\ddot{q}$. Therefore, the input torques τ can be directly obtained from (208):

$$\tau = (M_{qq} - M_{q\delta}M_{q\delta}^T)v + C_{qq}\dot{q} + C_{q\delta}\dot{\delta} - M_{q\delta}(C_{\delta q}\dot{q} + C_{\delta\delta}\dot{\delta} + K\delta) \tag{210}$$

where $v = \ddot{q}_d$.

For this control law to work, measurements or estimates of states q, $\dot{q}$, δ, and $\dot{\delta}$ are required. By plugging (210) into (208) we find the input-output linearized form:

$$\ddot{q} = v \tag{211}$$

The "flexible" subsystem, which is also the unobservable dynamics, is therefore:

$$\ddot{\delta} = -C_{\delta q}(q, \dot{q}, \delta)\dot{q} - C_{\delta\delta}(q, \dot{q}, \delta)\dot{\delta} - K(q)\delta - M_{q\delta}^{T} v \tag{212}$$

Trajectory tracking will be achieved by stabilizing the system with:

$$v = \ddot{q}_d + K_d(\dot{q}_d - \dot{q}) + K_p(q_d - q), \tag{213}$$

Where $K_d > 0$ and $K_p > 0$ are the feedback matrices that will stabilize the linear system shown in (211). Note that, unlike many other inversion-based methods, there is not a single inverted matrix present in the control law.

6.2.2.1 Stability Analysis

To prove the asymptotic stability along any trajectory $\ddot{q}_d$, the stability of (206) must be investigated. The equation can be represented in the following form:

$$\ddot{\delta} = -C_{\delta\delta}(q, \dot{q}, \delta)\dot{\delta} - K(q)\delta + u, \tag{214}$$

where $u = -C_{\delta q}(q, \dot{q}, \delta)\dot{q} - M_{q\delta}^{T} v$ can be treated as a disturbance to the system. Constraining the input to the system to a constant, and without the loss of generality, zero. Equation (214) is simplified to:

$$\ddot{\delta} = -C_{\delta\delta}(q, \dot{q}, \delta)\dot{\delta} - K(q)\delta \tag{215}$$

Recall that $\dot{M} - 2C$ is by definition skew symmetric, and because $M_{\delta\delta} = \mathbb{I}$, its derivative is 0, and therefore, $C_{\delta\delta}$ is skew symmetric. In the following derivation $C_{\delta\delta} \equiv C$ and $K(q) \equiv K$. Defining

$$x = \dot{\delta} + \alpha\delta, \tag{216}$$

where, α is a constant whose magnitude is to be determined. Writing out $\dot{\delta}$ from (216) yields:

$$\dot{\delta} = -\alpha\delta + x \tag{217}$$

Taking the derivative yields:

$$\ddot{\delta} = -\alpha\dot{\delta} + \dot{x} = \alpha^2\delta - \alpha x + \dot{x} \tag{218}$$

Therefore, rewriting (215) by inserting (217) and (218) leads to:

$$\alpha^2\delta - \alpha x + \dot{x} + C(-\alpha\delta + x) + K\delta = 0 \tag{219}$$

Grouping terms gives:

$$\dot{x} + (C - \alpha\mathbb{I})x + [K - C\alpha + \alpha^2]\delta = 0 \tag{220}$$

Rearranging yields:

$$\dot{x} + \bar{D}x + \bar{K}\delta = 0 \tag{221}$$

For non-linear system the Lyapunov's second method is a useful tool for stability analysis. The method requires that a positive definite candidate function, V, be defined, then its derivative, $\dot{V}$, determines the stability [74]. The following Lyapunov candidate can be used to prove that the origin of the system $\delta = \dot{\delta} = 0$ of the system (214) is stable:

$$V = x^T x + \delta^T \bar{K}\delta, \tag{222}$$

Taking derivative of the Lyapunov function (222) and inserting it into (221) and (217) yields:

$$\begin{aligned} \dot{V} &= 2x^T\dot{x} + 2\delta^T\bar{K}\delta = \\ &= -2x^T\bar{D}x - \cancel{2x^T\bar{K}\delta} + \cancel{2\delta^T\bar{K}x} - 2\alpha\delta^T\bar{K}\delta + \delta^T\dot{K}\delta \\ &= -x^T\bar{D}x - \delta^T(2\alpha\bar{K} - \dot{K})\delta \end{aligned} \tag{223}$$

Recall that

$$\bar{D} = C - \alpha\mathbb{I}, \tag{224}$$

where C is skew symmetric and *alpha* is a constant. Therefore, $\alpha > 0$ guarantees that:

$$x^T\bar{D}x > 0 \tag{225}$$

Therefore, the condition that guarantees the asymptotic stability of the system is:

$$(2\alpha\bar{K} - \dot{K}) > 0 \tag{226}$$

Because $K \equiv K(q)$, (226) defines the bounds for $\dot{q}$ that the joint trajectories have to satisfy. Furthermore, the system is exponentially stable if the following relation can be established [74]:

$$\dot{V} - \gamma V = 0 \tag{227}$$

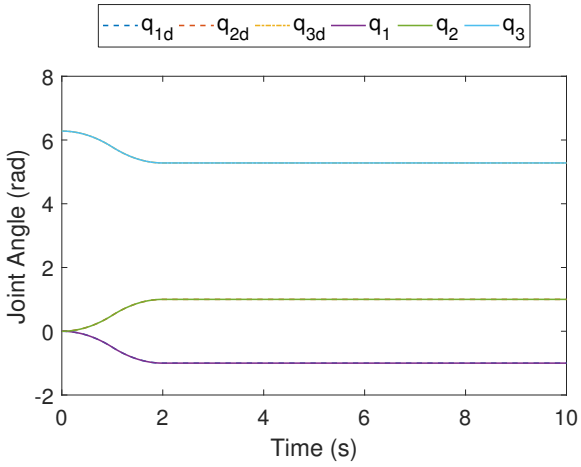


Figure 111: Joint Angles For Move 1

This can be achieved by enforcing the conditions:

$$x^T \bar{D} x \geq \lambda_{min}(x^T x) \tag{228}$$

and,

$$2\alpha\bar{K} - \dot{K} \geq \beta \tag{229}$$

Defining:

$$\gamma = min(\lambda_{min}(x^T x), \beta) \tag{230}$$

This is called exponential stability since:

$$V(t) \leq e^{-\gamma t} V(0) \tag{231}$$

Input-to-State Stability can be inferred from the exponential stability [149]. Furthermore, real systems have modal damping which further ensures the stability of the system.

6.2.2.2 Case Study

Consider the model of the two-link spatial robot presented in Section 3.3.7. The gain matrices in (210) are $K_p = diag(1,1,1)Nm/rad$ and $K_d = diag(2,2,2)Nms/rad$. In order to illustrate the behavior of the inversion-based controller, three characteristic moves are performed in simulation.

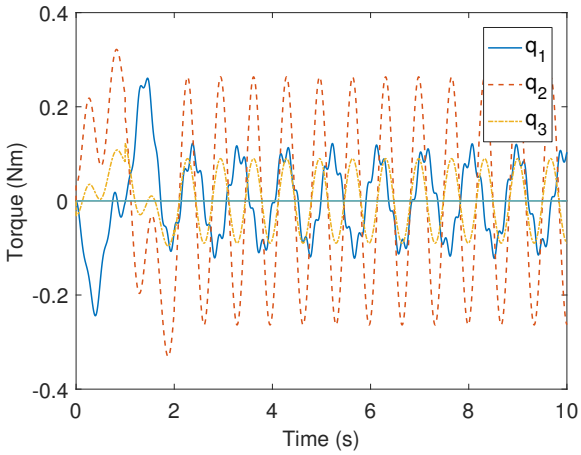
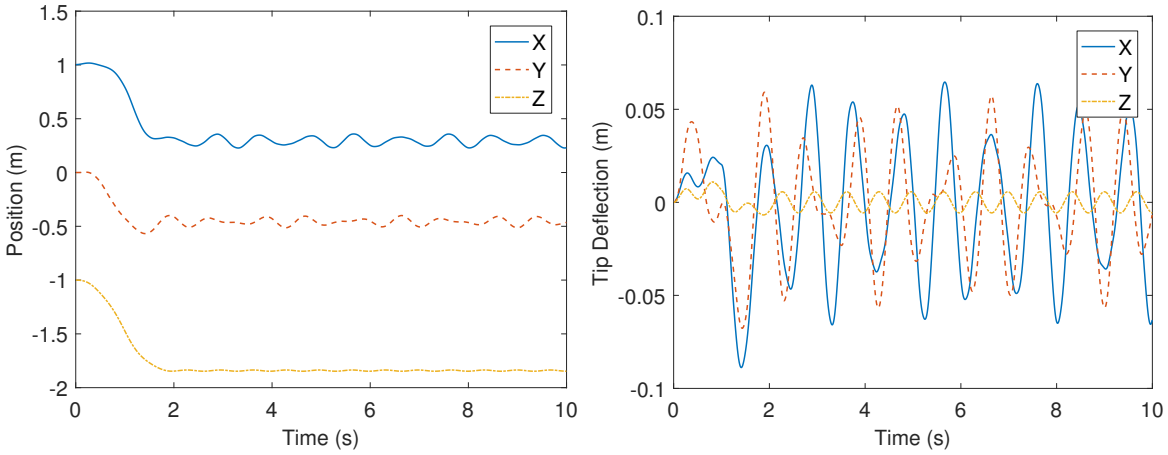


Figure 112: Torques For Move 1



(a) Tip Position

(b) Tip Deflection

Figure 113: Tip Response to Move 1

Move 1 is performed by giving all joints a bang-bang acceleration that results in a change of 1 *rad* for each joint. The duration of the motion is 2 s. Joint q_1 moves from 0 to -1 *rad*, joint q_2 moves from 1 to 2 *rad*, and joint q_3 moves from 6.2 to 5.2 rad. Figure 111 shows that the actual joint values, q_i, follow the desired joint values, q_d, well.

The torques that are required to perform the motion are shown in Figure 112. It can be seen that the torque amplitudes remain constant in steady state. This is due to the flexure not damping out.

Figure 113 show the response of the tip to move 1. Figure 113a shows the tip position

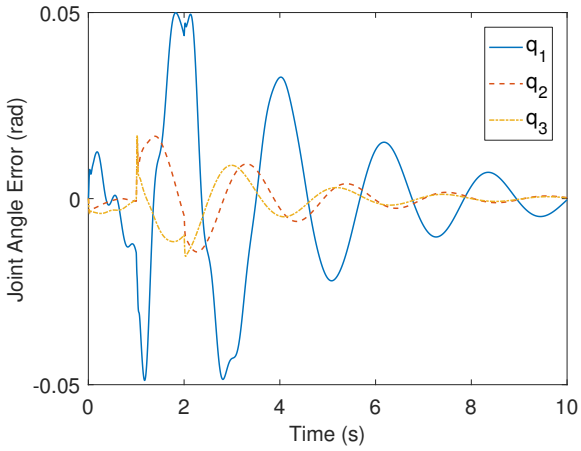


Figure 114: Joint Angle Errors for Move 1 for Traditional Feed forward Control Control Law

in the inertial frame and Figure 113b shows the tip deflection. Due to the joint remaining constant there is nothing that damps out the oscillations.

Figure 114 shows the joint motion for the same desired trajectory using the well known feed-forward control law for serial robots:

$$\tau = M_{qq}\left(\ddot{q}_d + K_d(\dot{q}_d - \dot{q}) + Kp(q_d - q)\right) + C_{qq}\dot{q}, \tag{232}$$

where, q_d is the desired joint position and q is the actual joint position. The same values of K_p and K_d are used in (232) as in (210). It can be seen that the oscillations in the arm cause perturbations in the joint position, in contrast to the control law presented in this section that allows for very precise joint trajectory tracking.

Simulations were also performed of move 2, where, joint $q1$ remains stationary, joint q_2 moves from 1 to 2 rad, and joint q_3 moves from 6.28 to 5.28 rad; and of move 3, where, joints q_2 and q_3 are held stationary and joint q_1 moves from 0 to -1 rad. The results are analogous to move 1 and are shown is Appendix H.1.

6.2.3 Singular Perturbation based control

The inversion-based controller presented in the previous section was not designed to directly damp out vibration, rather, provided a stable joint trajectory response in the presence of flexure. In this section, a controller that is designed to actively damp out oscillations is

presented. The equations of motion, that were derived in Chapter 3 are:

$$\begin{pmatrix} M_{qq}(q) & M_{q\delta}(q) \\ M_{q\delta}^T(q) & \mathbb{I} \end{pmatrix}\begin{pmatrix} \ddot{q} \\ \ddot{\delta} \end{pmatrix} + \begin{pmatrix} C_{qq}(q,\dot{q},\dot{\delta}) & C_{q\delta}(q,\dot{q},\dot{\delta}) \\ C_{\delta q}(q,\dot{q},\dot{\delta}) & C_{\delta\delta}(q,\dot{q},\dot{\delta}) \end{pmatrix}\begin{pmatrix} \dot{q} \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & K(q) \end{pmatrix}\begin{pmatrix} q \\ \delta \end{pmatrix} = \begin{pmatrix} \tau \\ 0 \end{pmatrix} \tag{233}$$

Without loss of generality, the gravity terms have been omitted. Equation (233) has $n+m$ states, but only n inputs. The singular perturbation based approach allows for the creation of a composite controller where the controller effort is divided into two components: slow control and fast control.

Since M is by definition symmetric positive definite, its inverse always exists:

$$M^{-1} = H = \begin{bmatrix} H_{qq} & H_{q\delta} \\ H_{\delta q} & H_{\delta\delta} \end{bmatrix}, \tag{234}$$

where,

$$H_{qq} = (M_{qq} - M_{q\delta}M_{q\delta}^T)^{-1} \tag{235}$$

$$H_{q\delta} = -M_{qq}^{-1}M_{q\delta}(\mathbb{I} - M_{q\delta}^T M_{qq}^{-1} M_{q\delta})^{-1} \tag{236}$$

$$H_{\delta q} = -M_{q\delta}^T(M_{qq} - M_{q\delta}M_{q\delta}^T)^{-1} \tag{237}$$

$$H_{\delta\delta} = (\mathbb{I} - M_{q\delta}^T M_{qq}^{-1} M_{q\delta})^{-1} \tag{238}$$

Therefore, solving (233) for $\ddot{q}$ and $\ddot{\delta}$ yields:

$$\ddot{q} = -H_{qq}(C_{qq}\dot{q} - C_{q\delta}\dot{\delta}) - H_{q\delta}(C_{\delta q}\dot{q} - C_{\delta\delta}\dot{\delta}) - H_{q\delta}K\delta + H_{qq}\tau \tag{239}$$

$$\ddot{\delta} = -H_{\delta q}(C_{qq}\dot{q} - C_{q\delta}\dot{\delta}) - H_{\delta\delta}(C_{\delta q}\dot{q} - C_{\delta\delta}\dot{\delta}) - H_{\delta\delta}K\delta + H_{\delta q}\tau \tag{240}$$

By using the shorthand:

$$C_{qq}\dot{q} - C_{q\delta}\dot{\delta} = c_r \tag{241}$$

$$C_{\delta q}\dot{q} - C_{\delta\delta}\dot{\delta} = c_f \tag{242}$$

The equation can be represented in a more compact form as:

$$\begin{aligned} \ddot{q} &= -H_{qq}c_r - H_{q\delta}c_f - H_{q\delta}K\delta + H_{qq}\tau \\ \ddot{\delta} &= -H_{\delta q}c_r - H_{\delta\delta}c_f - H_{\delta\delta}K\delta + H_{\delta q}\tau \end{aligned} \tag{243}$$

Selecting the size of the pertubation parameter is not a trivial task, but is often based on the physical interpretation of system dynamics. In [85, 117] it has been suggested that the reciprocal of lowest natural frequency k in the stiffness matrix, K, be chosen as the pertubation parameter, ϵ^2 (ϵ^2 is often defined as μ in singular pertubation based works). Introducing the coordinate change $\epsilon^2\xi = \delta$ in (243) yields the standard form for singularly perturbed models:

$$\ddot{q} = -H_{qq}(q)c_r(q,\dot{q},\epsilon^2\dot{\xi}) - H_{q\delta}c_f(q,\dot{q},\epsilon^2\dot{\xi}) - H_{q\delta}\tilde{K}(q)\xi + H_{qq}(q)\tau \tag{244}$$

$$\epsilon^2\ddot{\xi} = -H_{\delta q}(q)c_r(q,\dot{q},\epsilon^2\dot{\xi}) - H_{\delta\delta}c_f(q,\dot{q},\epsilon^2\dot{\xi}) - H_{\delta\delta}(q)\tilde{K}(q)\xi + H_{\delta q}(q)\tau, \tag{245}$$

Typical steps of singular perturbation formulation can be taken [75]. Due to ϵ, the system of equations (244,245) exhibit a boundary layer phenomenon with the fast variables ξ. If the boundary layer decays, then q and δ vary slowly. Setting $\epsilon = 0$ in (245) reduces the order of the system (244,245) to n and solving for ξ from (245) yields:

$$\bar{\xi} = \tilde{K}(\bar{q})^{-1} - {H_{\delta\delta}}^{-1}(\bar{q})[H_{\delta q}(\bar{q})c_r(\bar{q},\bar{\dot{q}},0) + H_{\delta q}(\bar{q})\tau] - \tilde{K}(\bar{q})^{-1}c_f(\bar{q},\bar{\dot{q}},0) \tag{246}$$

The overbars indicate that the system was evaluated at $\epsilon = 0$. Substituting (246) into (244) with $\epsilon = 0$ yields the reduced order subsystem for the slow part:

$$\bar{\ddot{q}} = [H_{qq}(\bar{q}) - H_{q\delta}(\bar{q},\bar{\dot{q}},0)H_{\delta\delta}^{-1}(\bar{q})][-c_r(\bar{q},\bar{\dot{q}},0,0) + \bar{\tau}] \tag{247}$$

From (235) it can be seen that (247) is the well known equation that governs the dynamics of a traditional rigid robot:

$$M_{qq}(\bar{q})\bar{\ddot{q}} + C_{qq}(\bar{q},\bar{\dot{q}})\bar{\dot{q}} = \bar{\tau} \tag{248}$$

To derive the fast subsystem, also know as the boundary layer system, it is assumed that $\xi = \bar{\xi} = 0$ and $q = \bar{q} = const.$ Therefore, the equilibrium trajectory for the fast variables is $\eta = \xi - \bar{\xi}$ and, therefore, $\dot{\eta} = \dot{\xi}$. By holding the slow variables constant from (245), the fast subsystem becomes:

$$\frac{d^2\eta}{dt_f^2} = -H_{\delta\delta}(\bar{q})\tilde{K}(\bar{q})\eta + H_{\delta q}\tau_f, \tag{249}$$

where fast time scale is defined as:

$$t_f = \frac{t}{\epsilon} \tag{250}$$

The system in (249) is parameterized in the slow variables $\bar{q}$. The state space form of system (249) is given by:

$$\frac{d\eta}{dt_f} = \begin{bmatrix} 0 & \mathbb{I} \\ -H_{\delta\delta}(\bar{q})\tilde{K}(\bar{q}) & 0 \end{bmatrix} \eta + \begin{bmatrix} 0 \\ H_{\delta q}(\bar{q}) \end{bmatrix} \tau_f \tag{251}$$

6.2.3.1 Composite Control

The dynamics of the flexible manipulator have now been divided into two reduced-order subsystems (248) and (251). A composite control strategy [75] can now be pursued. Therefore, the controller consists of two parts: a slow controller for the rigid coarse motion of the robot $\bar{\tau}$, and a fast controller τ_f to damp out the vibrations in the system caused by the motion of $\bar{q}$.

For the slow part, well established control techniques for serial arms can be used. For the fast subsystem, a Linear-quadratic regulator (LQR) can be used.

For a system defined as:

$$\dot{x} = Ax + Bu \tag{252}$$

with a cost function:

$$J = \int_0^\infty (x^T Q x + u^T R u) dt \tag{253}$$

the feedback control law that minimizes the value of the cost function is:

$$u = -Kx \tag{254}$$

The feedback gain K is given by:

$$K = R^{-1} B^T P, \tag{255}$$

where P satisfies the continious time Algebraic Riccati equation:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0, \tag{256}$$

where $Q > 0$. Technically, the Riccati equation would need to be recalculated every joint position. However, if that is not computationally feasible then, the feedback gain matrices can be calculated at the end of the trajectory [118]. This is a valid approach since the main

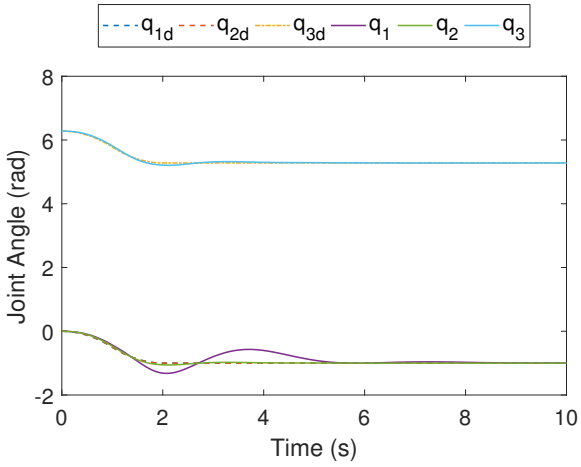


Figure 115: Joint Angles For Move 1

objective of the controller for flexible robots is to damp out the oscillations as fast as possible at steady state. However, care must be taken so that the boundary layer system does not go unstable along the trajectory q_d.

6.2.3.2 Case Study

In order to illustrate the singular pertubation based control law, a simulation study was performed. Consider the model of the two-link spatial robot presented in Section 3.3.7. In order to keep the slow and fast systems separated, the slow control uses K_p = $diag(1,1,1)Nm/rad$ and K_d = $diag(2,2,2)Nms/rad$, corresponding to a double pole at 1 rad/s. These gains were also used in the inversion based controller. The lowest natural frequency in the system for any trajectory is 6.35 rad/s. A standard LRQ approach with R = 10$\mathbb{I}$ and Q = 1$\mathbb{I}$, where $\mathbb{I}$ is the identity matrix, was used. The feedback gains for the fast control were calculated at the end of the trajectory, where the controller must quickly eliminate the oscillations in the arm.

Recall that Move 1 is performed by giving all joints a bang-bang acceleration that results in a change of 1 rad for each joint. The duration of the motion is 2 s. Joint q_1 moves from 0 to -1 rad, joint q_2 moves from 1 to 2 rad, and joint q_3 moves from 6.2 to 5.2 rad. Figure 115 shows that the actual joint values, q_i, do not perfectly follow the desired trajectory, q_{id}.

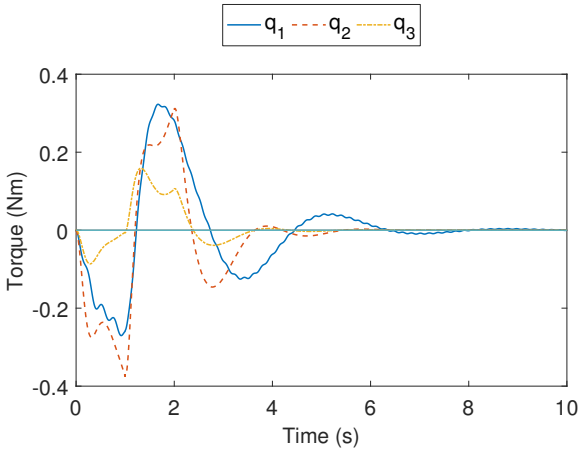
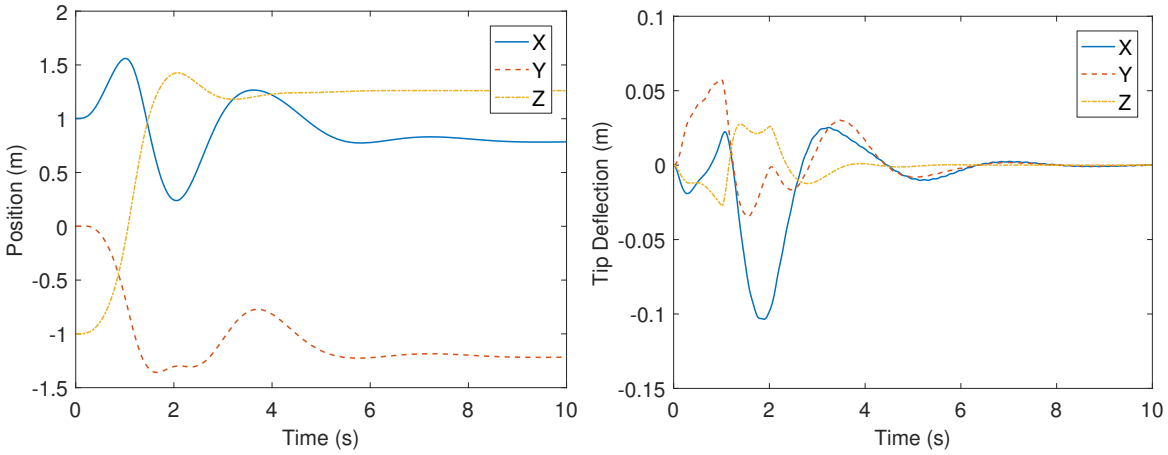


Figure 116: Torques For Move 1



(a) Tip Position

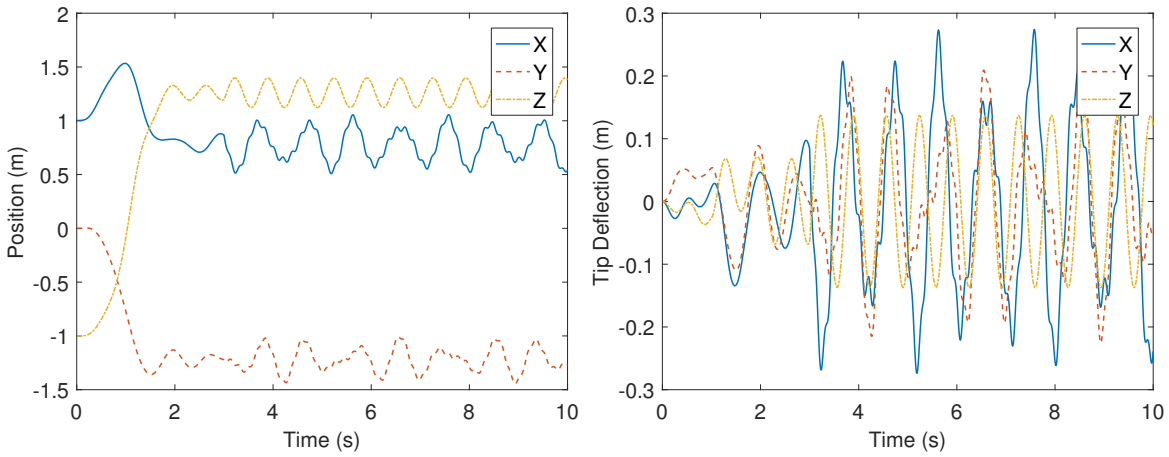
(b) Tip Deflection

Figure 117: Tip Response to Move 1

The torques that are required to perform the motion are shown in Figure 116. It can be seen that shortly after the transient phase is over, the torques go to zero, this is because the oscillations have been damped out.

Figure 117 shows the response of the tip to move 1. Figure 117a shows the tip position in the inertial frame and Figure 117b shows the tip deflection. After the transient period, the oscillations are quickly damped out and the arm remains stationary for the rest of the simulation.

Simulations were also performed of move 2, where, joint q_1 remains stationary, joint q_2



(a) Tip Position (b) Tip Deflection

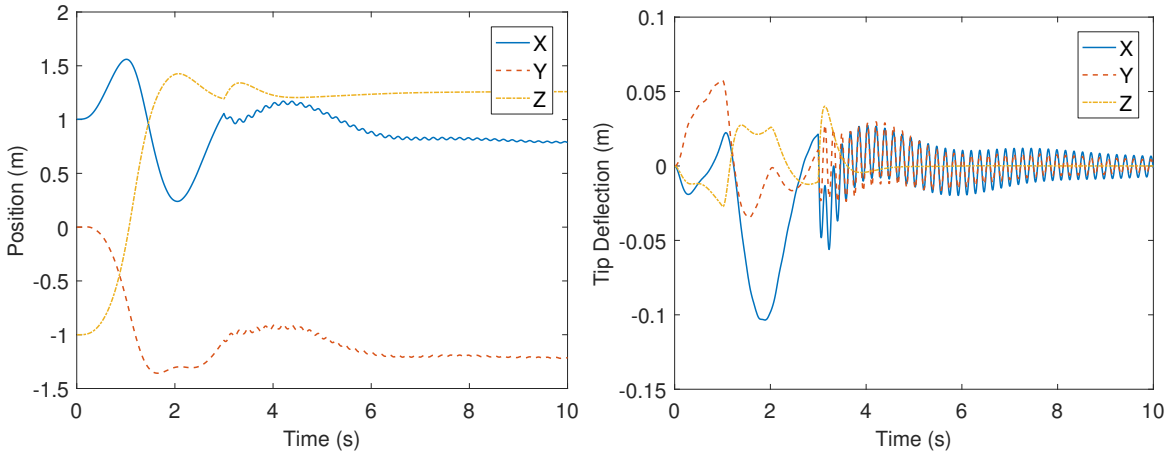
Figure 118: Tip Response to Move 1 with an Impulse disturbance at $t = 3s$ without Fast Controller enabled

moves from 1 to 2 *rad*, and joint q_3 moves from 6.28 to 5.28 *rad*; and of move 3, where, joints q_2 and q_3 are held stationary and joint q_1 moves from 0 to -1 *rad*. The results are analogous to move 1 and are shown is Appendix H.2.

To illustrate the disturbance rejection characteristics, an impulse disturbance was added to the model at $t = 3s$ during move 1.

Figure 118 shows the response of the tip, when the fast controller is enabled i.e there is no active oscillation cancellation. It can be seen that deflection is about 20 *cm* in all spatial directions after the impulse and the oscillation amplitude remains consistent during the rest of the simulation. Figure 119 shows a greatly improved result when the fast controller is enabled. After the impulse disturbance at $t = 3s$, the controller keeps the deflection much smaller compared to when it was not enabled. The deflection of the tip damps out as the simulation progresses. It can be seen from Figure 119a that the tip reaches its steady state value about 3 seconds after the impulse.

In order to illustrate the effect of modal accessibility that was presented in Section 6.2.1, responses to non-zero initial conditions in the flexible variables δ at three configurations are presented. All configurations have the same elbow angle (q_3=-π/2). The shoulder angles



(a) Tip Position (b) Tip Deflection

Figure 119: Tip Response to Move 1 with an Impulse disturbance at $t = 3s$ with Fast Controller enabled

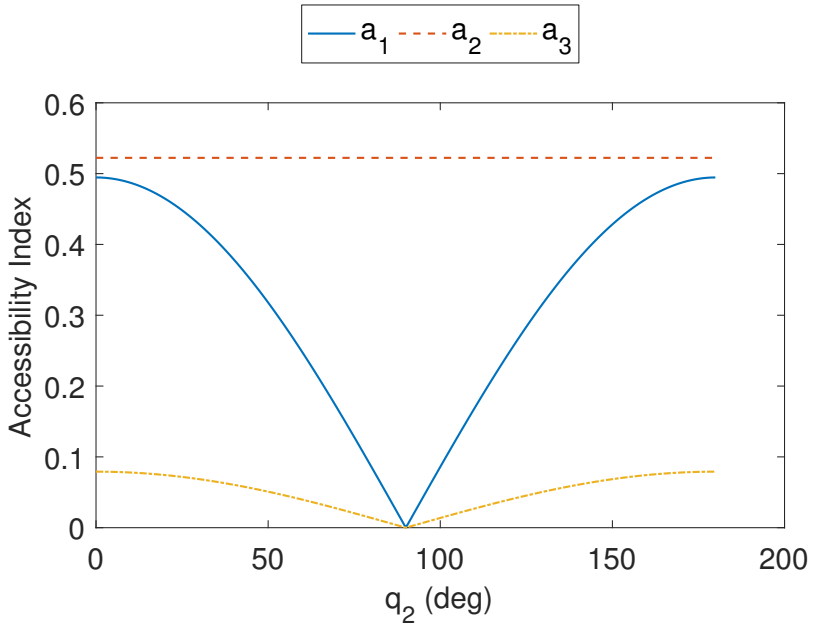
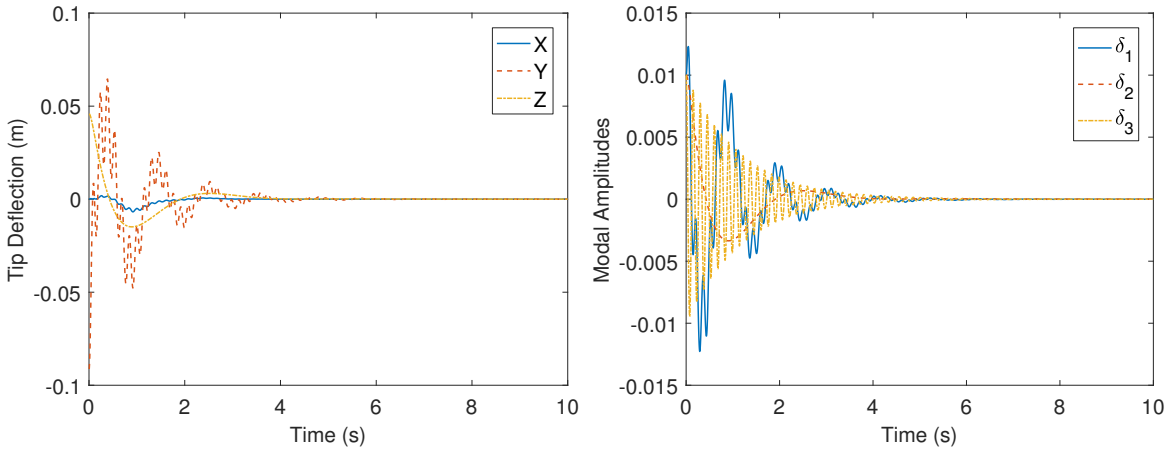


Figure 120: Accessibility indexes at $q_3 = -\pi/2$

used for this illustration are: $q_2 = 0\ rad$ (arm fully horizontal configuration); $q_2 = 1\ rad$; and $q_2 = \pi/2\ rad$ (arm fully vertical configuration). The accessibility indexes for the first modes (a_1, a_2, a_3) when $q_3 = -\pi/2$ are shown in Figure 120. It can be seen that the accessibility of mode 2 is not affected by the change in the shoulder angle. The accessibility for modes 1 and 3 drops significantly when the arm nears the vertical configuration ($q_2 = 90°$). This



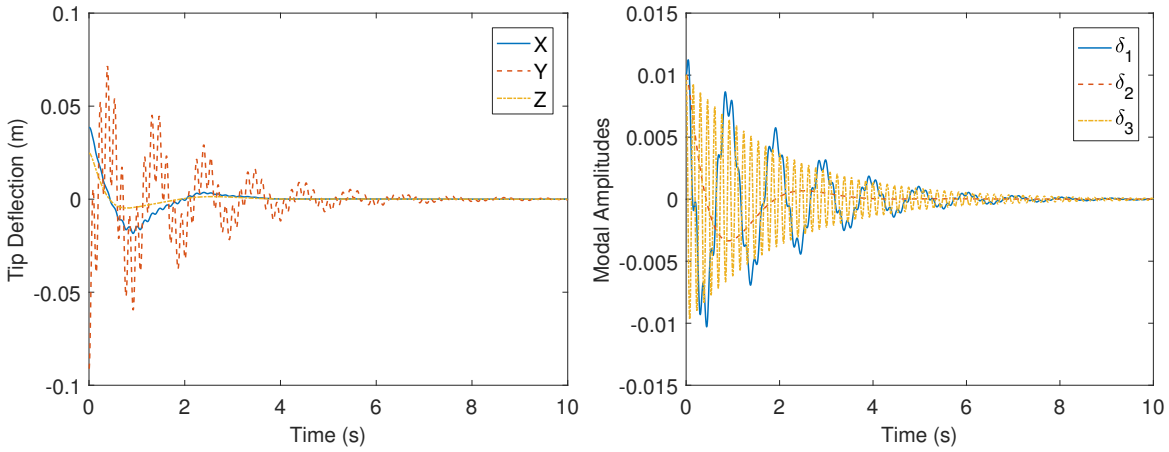
(a) Tip Deflection (b) Modal Amplitudes

Figure 121: Response to initial condition at $q_2 = 0$, $q_3 = -\pi/2$

configuration was illustrated in Figure 107. The gains for the controller were calculated for the horizontal configuration ($q_2 = 0$), and were kept consistent for all configurations to illustrate the concept of modal accessibility.

Figure 121 shows the response of the robot arm to non-zero initial conditions in the flexible variables when the arm is in the horizontal configuration ($q_2 = 0$). Figure 121a shows the tip deflection from the undeformed structure. It can be seen that the initial deflection of about 5 cm is damped out in about 4 seconds. The modal amplitudes for this response can be seen in Figure 121b. It can be seen that all the modes damp down to insignificant magnitudes at the same time.

Figure 122 shows the response of the robot arm to non-zero initial conditions in the flexible variables when the arm is at configuration ($q_2 = 1\ rad$). Figure 122a shows the tip deflection from the undeformed structure. Relative to the response in the horizontal configuration, it took almost twice as long for the oscillations to damp out. The reason for this can be seen in Figure 122b, where the modal amplitudes are shown. Mode 2 is damped out at about the same rate as in the horizontal configuration, which is to be expected because the accessibility index for mode 2 is the same. However, for modes 1 and 3, the accessibility index has lowered and using the same gains it takes the controller much longer to damp out oscillations. In order to damp the oscillations faster, the controller gains would need to be



(a) Tip Deflection

(b) Modal Amplitudes

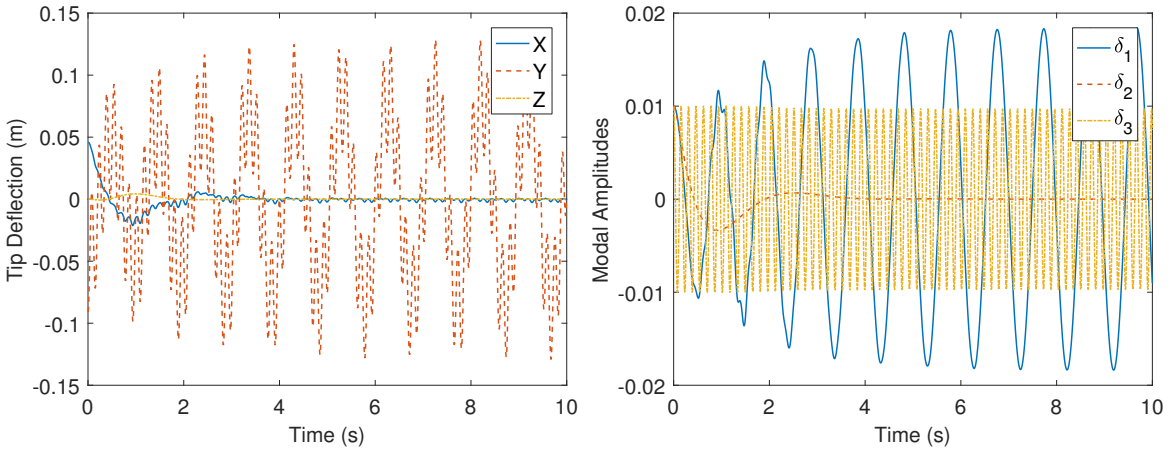
Figure 122: Response to initial condition at $q_2 = 1\ rad$, $q_3 = -\pi/2$

higher, this, however increases the actuator effort.

Figure 123 shows the response of the robot arm to non-zero initial conditions in the flexible variables when the arm is in the vertical configuration. Figure 123a shows the tip deflection from the undeformed state. It can be seen that the oscillations in the X and Z axis do not damp out. The modal amplitudes, shown in Figure 123b, show that modes 1 and 3 do not damp out, while mode 2 has the same behavior as in the earlier examples. The reason why modes 1 increases in amplitude is due to non-linear effects while damping out mode 2. This behavior is explained by the accessibility indexes dropping to 0 for both modes 1 and 3. Physically, this means that none of the joints has any effect on those modes, regardless what the controller does. In order to reduce a the oscillations the arm would need to be taken to a configuration where the accessibility indexes are non-zero.

6.3 Chapter Summary

This chapter described controlling a flexible manipulator arm with both open-loop and closed-loop techniques. An optimized input-shaping algorithm was presented and the results were confirmed with FEA analysis and physical experiments. The controllability of natural modes was discussed and analyzed. An inversion based closed-loop controller was presented that guarantees stable joint trajectory tracking for a flexible manipulator arm. A singular



(a) Tip Deflection

(b) Modal Amplitudes

Figure 123: Response to initial condition at $q_2 = \pi/2$, $q_3 = -\pi/2$

pertubation based controller was presented to actively damp out the vibrations in the arm. Simulation results were presented for the closed-loop controllers.

Chapter VII

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

The motivation for this work arose from the understanding that traditional serial manipulators are constructed to be very stiff, and therefore, inefficient with regards to numerous performance criteria. Reducing the weight and stiffness of manipulators generally introduces unwanted flexible behavior. Complications modeling flexible manipulators, especially the ones with multiple flexible links in spatial configurations, has consistently been a problem for researchers. In addition, without the availability of a low-order high-fidelity model, estimation and controller design are difficult to achieve. Therefore, a considerable amount of work in this dissertation is directed at deriving a universal model for flexible manipulators that includes tip attachments, motor inertias, and is applicable to arms of n links in spatial configurations. This model can be used as the basis for estimation and controller design.

A systematic approach to finding natural frequencies and mode-shapes for n-link spatial serial structures is presented. The model relies on using the transfer matrix modeling method extended to spatial degrees of freedom. Algorithms are presented for assembling the transfer matrices and combining them for any serial structure. The model was validated using Finite Element Analysis and experiments. The test body experiments were carried out with a shaker setup. There was good agreement between the predictions and the experimental measurements. This method can be used for generating more efficient models for structures that consists of serially connected beams, such as solar arrays, fixtures, and crane structures.

A new low-order dynamical model based on varying mode-shapes for serial flexible robot arms is presented. The model is derived using a Lagrangian approach, and allows for inclusion of rigid attachments. Because the model is based on locally defined mode shapes, a method called Global Modal Parametrization is used to ensure that the model varies continuously in the configuration space by scaling the modes appropriately. A modified

Modal Assurance Criterion algorithm is presented to track the physical interpretation of the modes, from one configuration to the next, during model generation. The model was verified with commercial FEA software, that produced results very similar to the proposed method. The main benefits of this model are the systematic approach for derivation and the computational efficiency compared to other high fidelity models.

In order to verify the claims made in this dissertation, a two-link, three-joint, test bed was designed and constructed. The joints of the test bed have harmonic drives and belt drive systems to move the links. The controller for the test bed runs on a real-time PC with a data acquisition card for inputs and outputs. Two accelerometers on the arm are used to sense flexure. The test bed illustrates how the natural frequencies of the system change with the configuration. The data obtained during motions of the manipulator did not match the model perfectly; however, it was close enough to use for estimation and control.

A novel use of Kalman filtering techniques is used to estimate the flexible states for a multi-link spatial flexible robot arm using data from accelerometers and strain gages. The system, and measurement models were derived for an extended Kalman filter for both strain and accelerometer feedback. A simulation study was conducted to illustrate the estimator's performance for various errors in the model. The filters were proven to be robust to various errors in the signals and the model. There was no inherent difference between the accuracy of strain and acceleration based feedback. Impulse tests were conducted to illustrate the disturbance observation characteristics of the estimator. The estimator successfully restored the state during motions of the robot arm, and the simulated states matched the experimental ones well.

Open-loop controllers can be used to cancel out the vibrations that trajectory tracking would induce. Therefore, an optimized input shaping algorithm based on Specified Insensitivity shaping for multiple frequency ranges is presented. The results were confirmed with FEA analysis and experiments with the test bed. The proposed shaper allows for a large reduction in both residual and transient deflection, while increasing the command duration only slightly.

The controllability of natural modes is discussed and analyzed. It was shown that in

certain configurations some modes are either completely inaccessible or very hard to access. An inversion based closed-loop controller is presented that guarantees stable joint trajectory tracking for flexible manipulator arms. Simulations were performed to show the response of the arm to the controller. A singular pertubation based controller is presented to actively damp out the vibrations in the arm. Simulations show that the oscillations in the arm can be damped out quickly.

7.1.1 Summary of Contributions

This dissertation makes contributions in the areas of modeling, estimation, open- and closed-loop control for flexible serial manipulators. All of the contributions were made possible thanks to the low-order high fidelity model. The specific contributions include:

- A systematic extension of the transfer matrix method for n-link spatial serial structures
  - The results were verified with FEA analysis and experiments
- A new low order model for flexible serial manipulator based on exact modes of the system
  - The results were verified with FEA analysis and experiments
- Development of a 2-link, 3-joint flexible manipulator testbed
- An extended Kalman filter based estimator for flexible states based on strain and acceleration feedback
  - The strain based feedback was investigated in simulation
  - Acceleration based feedback was investigated in simulation and verified experimentally
- An optimized input-shaping method based for flexible manipulators
  - The results were verified with FEA analysis and experiments
- Modal accessibility analysis for serial manipulators

- An inversion based and a singular pertubation based closed-loop controller for flexible manipulators
  - Numerical examples were provided

7.2 Future Work

7.2.1 Inclusion of Joint Flexibility

The dynamical model presented in Chapter 3 assumes that the flexibility in the joint is negligible. There are numerous examples from the real world where significant joint flexure. Therefore, the model could be extended to include such conditions. The system model would then be a function of the motor angles θ, link angles q, and flexible states δ:

$$M(q,\theta)\begin{bmatrix}\ddot{q}\\ \ddot{\theta}\\ \ddot{\delta}\end{bmatrix} + C(q,\dot{q},\dot{\delta})\begin{bmatrix}\dot{q}\\ \dot{\theta}\\ \dot{\delta}\end{bmatrix} + K(q)\begin{bmatrix}q\\ \theta\\ \delta\end{bmatrix} + G(q,\theta) = Q(q,\theta)\tau \tag{257}$$

There has been some work done in this area, for example [67,128]. However, their models do not take into account that the modes of the flexible robot arms can be used for a low-order high-fidelity model. The joint flexure can be added into the transfer matrix model as joint stiffness.

7.2.2 Robust Estimation

The estimation and controller algorithms presented in this dissertation work effectively when the model of the robot is known well. In reality, however, the kinematic properties change based on the task that is being performed. In general, the end-effector load might not be known. Therefore, the estimator would not only have to reconstruct the state of the robot, but also estimate which end-effector loads are present. To achieve this goal, multiple estimators can be used concurrently and then a high-level algorithm could select the model whose estimate fits the current situation best.

Running multiple estimators on traditional processors is computationally expensive. To achieve fast enough loop rates, a fairly powerful (Intel i7-2600K) processor was used in the real-time controller in this research. Due to the complexity of the model, the estimator

utilized over 90% of the computational power of a single core while, running a ODE4 solver at 1 kHz loop rates.

The solution to this limitation could be solved with graphical processing units (GPUs) and other specialized computing platforms that have a large number of parallel cores. An estimator could be implemented to run on its own core and then communicate to a high-level coordinator.

7.2.3 Collision Detection

One of the big benefits of using flexible arms is that they are intrinsically safer due to less moving mass. Therefore, it is conceivable that robots with flexible arms can be allowed to work alongside humans or fragile environments. With the inclusion of sensors on the manipulator arm, a collision detector can be derived that calculates the reference states of the robot. If an estimate based on a sensor signal, such as acceleration, significantly deviates from the reference states, then a collision event has occurred. Depending on the task, the robot then either stops or moves backward in the trajectory. The estimator presented in this dissertation is shown to respond quickly to disturbances and, therefore, can be used as the base for such a collision detector.

Appendix A

RIGID ROBOT KINEMATICS

A.1 Forward Kinematics

The frames of a serial robot are numbered consecutively from 0 to n starting from the base of the manipulator and ending with the tip of the end-effector. Each frame has a coordinate system and a position associated with it. We can use 4×4 homogeneous transformation matrices to transform coordinate system i to $i+1$

$$ {}_{i-1j}B = \begin{bmatrix} cos\theta_i & -sin\theta_i & 0 & a_{i-1} \\ cos\alpha_{i-1}sin\theta_i & cos\alpha_{i-1}cos\theta_i & -sin\alpha_{i-1} & -sin\alpha_{i-1}d_i \\ sin\alpha_{i-1}sin\theta_i & cos\alpha_{i-1}cos\theta_i & cos\alpha_{i-1} & cos\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{258} $$

where θ_i, d, a, α are the DH parameters. This transformation can also be represented as:

$$ {}_{i-1j}B = \begin{bmatrix} {}_{i-ij}O & {}_iP(o_io_j) \\ 0 & 1 \end{bmatrix}, \tag{259} $$

where ${}_{in}O$ is the rotation of the axes in frame j with respect to frame i and ${}_iP(o_io_j)$ is the position of the origin in frame i. We can multiply these matrices together to find the location and rotation of any frame in the system relative to one-another:

$$ {}_{ik}B = ({}_{ij}B)({}_{jk}B) \tag{260} $$

A.2 Inverse Kinematics

The inverse kinematics involves a procedure of finding the joint angles or distances based on the desired end-effector location and rotation. This procedure has to be completed for each robot separately. The inverse kinematics problem is not as simple as the forward kinematics one. Because the kinematic equations are nonlinear, their solution is not always easy (or even possible) in a closed form. Often multiple choices(branches) exist. There are

also configurations, known as singularities, where there is an infinite number of solutions. The existence or nonexistence of a kinematic solution defines the workspace of a given manipulator.

A.3 Velocity Analysis

Velocity analysis maps velocities in the Cartesian space to the joint space using a matrix quantity called a the Jacobian. The number of rows in the Jacobian equals number of degrees of freedom in the Cartesian space (displacements and rotations) and the number of columns equals the number of joints in the manipulator. For a general 6 joint robot in 3D space the Jacobian is of dimension 6x6. The general form of the Jacobian written in frame i is

$$\begin{Bmatrix} {}_i v \\ {}_i \omega \end{Bmatrix} =_i J(\theta)\dot{\theta} \tag{261}$$

To find the velocities of the joints we simply invert the Jacobian

$$\dot{\theta} =_i J^{-1}(0_{n+1}) \begin{Bmatrix} {}_i v \\ {}_i \omega \end{Bmatrix} \tag{262}$$

We can transform the Jacobian between different points by using the Jacobian shifting law:

$$J(o) = X(oc)J(c), \tag{263}$$

where

$$X(oc) = \begin{bmatrix} 1 & p(oc)\times \\ 0 & 1 \end{bmatrix}, \tag{264}$$

where p(oc)x is the matrix cross product operator. The transformation of the Jacobian to different frames is the following:

$${}_i J(o) =_{ij} Z(oc)_j J(o) =_i X(oc)_{ij} Y_j J(o) =_i X(oc) \begin{bmatrix} {}_{ij}O & 0 \\ 0 & {}_{ij}O \end{bmatrix}_j J(o) \tag{265}$$

Appendix B

DYNAMICAL MODEL FOR A N-LINK FLEXIBLE STRUCTURE

B.1 Kinetic and Potential Energy of the Flexible subsystem

When analyzing a rigid structure with n serial links equations of motion can be found easily using energy based methods. Kinetic energy, potential energy, and energy dissipation equations are formulated using assumed mode shapes, that satisfy the kinematic boundary conditions, as basis functions. The equation of motion can then determined by the Lagrange's Equation (266).

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = Q_i, \tag{266}$$

where T is the kinetic energy, V is the potential energy, D is the dissipative term, and Q is the Forcing. The Kinematics of a flexible manipulator in 3D space can be expressed as a sum of of mode shapes ϕ multiplied with the time-varying amplitudes $q(t)$. Remember from Chapter 3 that the mode-shapes consist of 4 components.

$$\phi(z,t) = \sum_{i=1}^{\infty} \delta_i(t) \begin{bmatrix} w_x(z) \\ w_x(z) \\ w_y(z) \\ \theta_z(z) \end{bmatrix} \tag{267}$$

The amplitudes and derivatives of $\delta_i(t)$ become the states of the model. Compatible joint angle variables and their derivatives are also included as the rigid state variables. The flexible and rigid kinematics combined describes the position and velocity of every point on

the arm and can be used to express the kinetic energy T and the potential energy V.

$$\begin{aligned} \Theta(z,t) &= \sum_i \delta_i(t)\theta_{i_z}(z) \\ W(z,t) &= \sum_i \delta_i(t) w_i(z) \\ w_i(z) &= \begin{bmatrix} w_{i_x}(z) \\ w_{i_y}(z) \\ w_{i_z}(z) \end{bmatrix} \end{aligned} \tag{268}$$

The Kinetic energy of the flexible system can be expressed as:

$$\begin{aligned} T = \frac{1}{2}\int_0^L \rho A\Big(\frac{\partial W}{\partial t}\Big)^2 dz + \frac{1}{2}\int_0^L \rho J\Big(\frac{\partial \Theta}{\partial t}\Big)^2 dz + \sum_n \frac{1}{2} m_n \Big(\frac{\partial W}{\partial t}\Big)^2 (z_n,t) + \\ \sum_n \frac{1}{2} I_{zz}\Big(\frac{\partial \Theta}{\partial t}\Big)^2 (z_n,t) + \sum_n \frac{1}{2} I_{yy}\Big(\frac{\partial^2 W_x}{\partial t \partial z}\Big)^2 (z_n,t) + \sum_n \frac{1}{2} I_{xx}\Big(\frac{\partial^2 W_y}{\partial t \partial z}\Big)^2 (z_n,t) \end{aligned} \tag{269}$$

By collecting the terms we can separate the time-dependent variables from the spatial variables

$$\begin{aligned} T = \frac{1}{2}\sum_j \sum_i \dot{\delta}_i \dot{\delta}_j \Bigg[\sum_n \Bigg(\int_{L_{n-1}}^{L_n} (\rho_n A_n w_i w_j + \rho_n J_n \theta_i \theta_j) dz \Bigg) + \sum_n m_n w_i(z_n) w_j(z_n) + \\ + \sum_n I_{zz_n} \theta_i(z_n)\theta_j(z_n) + \sum_n I_{(xx)_n} w'_{i_y}(z_n) w'_{i_y}(z_n) + \sum_n I_{(yy)_n} w'_{i_x}(z_n) w'_{i_x}(z_n) \Bigg] \end{aligned} \tag{270}$$

We then get the elements of the mass matrix $M_{\delta\delta}$

$$\begin{aligned} m_{ij} = \sum_n \Bigg(\int_{L_{n-1}}^{L_n} (\rho_n A_n w_i w_j + \rho_n J_n \theta_i \theta_j) dz \Bigg) + \sum_n m_n w_i(z_n) w_j(z_n) + \\ + \sum_n I_{zz_n} \theta_i(z_n)\theta_j(z_n) + \sum_n I_{(xx)_n} w'_{i_y}(z_n) w'_{i_y}(z_n) + \sum_n I_{(yy)_n} w'_{i_x}(z_n) w'_{i_x}(z_n) \end{aligned} \tag{271}$$

Analagous steps can be taken to find the potential energy to get the elements of the stiffness matrix K

$$V = \frac{1}{2}\int_0^L EI_y\Big(\frac{\partial^2 W_x}{\partial z^2}\Big)^2 dz + \int_0^L EI_x\Big(\frac{\partial^2 W_y}{\partial z^2}\Big)^2 dz + \frac{1}{2}\int_0^L GJ\Big(\frac{\partial \Theta}{\partial z}\Big)^2 dz \tag{272}$$

$$V = \frac{1}{2}\sum_j \sum_i \delta_i \delta_j \Bigg[\sum_n \Bigg(\int_{L_{n-1}}^{L_n} (E_n I_{(xx \vee yy)n} w''_{i_{xx \vee yy}} w''_{j_{xx \vee yy}} + G_n J_n \theta'_i \theta'_j) dz \Bigg) \Bigg] \tag{273}$$

$$k_{ij} = \sum_n \Bigg(\int_{L_{n-1}}^{L_n} (E_n I_{(xx \vee yy)n} w''_{i_{xx \vee yy}} w''_{j_{xx \vee yy}} + G_n J_n \theta'_i \theta'_j) dz \Bigg) \tag{274}$$

This results in a dynamical equation of the form:

$$\ddot{\delta} + D\dot{\delta} + K\delta = Q, \tag{275}$$

where Q=

The matrices D and K are constant with

$$K = \begin{bmatrix} \omega_1^2 & 0 & \cdots & 0 \\ 0 & \omega_2^2 & \cdots & 0 \\ \vdots & & \ddots & \\ 0 & 0 & \cdots & \omega_n^2 \end{bmatrix} \tag{276}$$

$$D = \begin{bmatrix} 2\zeta_1\omega_1 & 0 & \cdots & 0 \\ 0 & 2\zeta_2\omega_2 & \cdots & 0 \\ \vdots & & \ddots & \\ 0 & 0 & \cdots & 2\zeta_n\omega_n \end{bmatrix} \tag{277}$$

Appendix C

PROBLEM WITH TRADITIONAL MODELING APPROACHES

We will write out the full kinetic energy based on the methodology that many researchers have done in the past. The way we get the kinetic energy for link k is to modify the standard flexure $w_k(x,t) = \sum_{i=1}^{modes} \dot{\delta}_{ik}\phi_{ik}$ with the displacement at the beginning of the link k so we get $w_k(x,t) = \sum_{i=1}^{modes} \dot{\delta}_{ik}\phi_{ik} + \sum_{f=1}^{k-1}\sum_{i=1}^{modes} \dot{\delta}_{if}\phi_{if}(L_f)$

$$K = \frac{1}{2}\sum_{i=1}^{modes}\sum_{j=1}^{modes}\sum_{k=1}^{links}\left(\int_0^{L_k}\rho_k A_k\left(\dot{\delta}_{ik}\phi_{ik} + \sum_{f=1}^{k-1}\dot{\delta}_{if}\phi_{if}(L_f)\right)\left(\dot{\delta}_{jk}\phi_{jk} + \sum_{f=1}^{k-1}\dot{\delta}_{jf}\phi_{jf}(L_f)\right)dx\right.$$
$$\left. + m_k\left(\dot{\delta}_{ik}\phi_{ik}(L_k) + \sum_{f=1}^{k-1}\dot{\delta}_{if}\phi_{if}(L_f)\right)\left(\dot{\delta}_{jk}\phi_{jk}(L_k) + \sum_{f=1}^{k-1}\dot{\delta}_{jf}\phi_{jf}(L_f)\right)\right) =$$
$$\frac{1}{2}\sum_{i=1}^{modes}\sum_{j=1}^{modes}\sum_{k=1}^{links}(\rho_k A_k\int_0^{L_k}(\dot{\delta}_{ik}\dot{\delta}_{jk}\phi_{ik}\phi_{jk} + \dot{\delta}_{ik}\phi_{ik}\sum_{f=1}^{k-1}\dot{\delta}_{jf}\phi_{jf}(L_f) + \dot{\delta}_{jk}\phi_{jk}\sum_{f=1}^{k-1}\dot{\delta}_{if}\phi_{if}(L_f) +$$
$$\sum_{f=1}^{k-1}\dot{\delta}_{if}\dot{\delta}_{jf}\phi_{jf}(L_f)\phi_{if}(L_f))dx + m_k(\dot{\delta}_{ik}\dot{\delta}_{jk}\phi_{ik}(L_k)\phi_{jk}(L_k) + \dot{\delta}_{ik}\phi_{ik}(L_k)\sum_{f=1}^{k-1}\dot{\delta}_{jf}\phi_{jf}(L_f)$$
$$+ \dot{\delta}_{jk}\phi_{jk}(L_k)\sum_{f=1}^{k-1}\dot{\delta}_{if}\phi_{if}(L_f) + \sum_{f=1}^{k-1}\dot{\delta}_{if}\dot{\delta}_{jf}\phi_{jf}(L_f)\phi_{if}(L_f)))) \tag{278}$$

where m_k is the mass at the end of link k

if we have 2 modes per link and 2 links we can expand K to:

$$
\begin{aligned}
2K = \rho_1 A_1 \int_0^{L_1} \dot{\delta}_{11}\dot{\delta}_{11}\phi_{11}\phi_{11}dx + m_1\dot{\delta}_{11}\dot{\delta}_{11}\phi_{11}(L_1)\phi_{11}(L_1) + \\
\rho_1 A_1 \int_0^{L_1} \dot{\delta}_{21}\dot{\delta}_{21}\phi_{21}\phi_{21}dx + m_1\dot{\delta}_{21}\dot{\delta}_{21}\phi_{21}(L_1)\phi_{21}(L_1) + \\
\rho_2 A_2 \int_0^{L_2} \dot{\delta}_{12}\dot{\delta}_{12}\phi_{12}\phi_{12} + \dot{\delta}_{12}\dot{\delta}_{11}\phi_{12}\phi_{11}(L_1) + \dot{\delta}_{12}\dot{\delta}_{11}\phi_{12}\phi_{11}(L_1) + \dot{\delta}_{11}\dot{\delta}_{11}\phi_{11}(L_1)\phi_{11}(L_1)dx + \\
m_2\left(\dot{\delta}_{12}\dot{\delta}_{12}\phi_{12}(L_2)\phi_{12}(L_2) + \dot{\delta}_{12}\dot{\delta}_{11}\phi_{12}(L_2)\phi_{11}(L_1) + \dot{\delta}_{12}\dot{\delta}_{11}\phi_{12}(L_2)\phi_{11}(L_1) + \dot{\delta}_{11}\dot{\delta}_{11}\phi_{11}(L_1)\phi_{11}(L_1)\right) \\
2\left(\rho_1 A_1 \int_0^{L_1} \dot{\delta}_{11}\dot{\delta}_{21}\phi_{11}\phi_{21}dx + m_1\dot{\delta}_{11}\dot{\delta}_{21}\phi_{11}(L_1)\phi_{21}(L_1)\right) + \\
2\left(\rho_2 A_2 \int_0^{L_2} \dot{\delta}_{12}\dot{\delta}_{22}\phi_{12}\phi_{22} + \dot{\delta}_{12}\dot{\delta}_{21}\phi_{12}\phi_{21}(L_1) + \dot{\delta}_{22}\dot{\delta}_{11}\phi_{22}\phi_{11}(L_1) + \dot{\delta}_{11}\dot{\delta}_{21}\phi_{11}(L_1)\phi_{21}(L_1)dx + \right. \\
\left. m_2\left(\dot{\delta}_{12}\dot{\delta}_{22}\phi_{12}(L_2)\phi_{22}(L_2) + \dot{\delta}_{12}\dot{\delta}_{21}\phi_{12}(L_2)\phi_{21}(L_1) + \dot{\delta}_{22}\dot{\delta}_{11}\phi_{22}(L_2)\phi_{11}(L_1) + \dot{\delta}_{11}\dot{\delta}_{21}\phi_{11}(L_1)\phi_{21}(L_1)\right)\right) + \\
\rho_2 A_2 \int_0^{L_2} \dot{\delta}_{22}\dot{\delta}_{22}\phi_{22}\phi_{22} + \dot{\delta}_{22}\dot{\delta}_{21}\phi_{22}\phi_{21}(L_1) + \dot{\delta}_{22}\dot{\delta}_{21}\phi_{22}\phi_{21}(L_1) + \dot{\delta}_{21}\dot{\delta}_{21}\phi_{21}(L_1)\phi_{21}(L_1)dx + \\
m_2\left(\dot{\delta}_{22}\dot{\delta}_{22}\phi_{22}(L_2)\phi_{22}(L_2) + \dot{\delta}_{22}\dot{\delta}_{21}\phi_{22}(L_2)\phi_{21}(L_1) + \dot{\delta}_{22}\dot{\delta}_{21}\phi_{22}(L_2)\phi_{21}(L_1) + \dot{\delta}_{21}\dot{\delta}_{21}\phi_{21}(L_1)\phi_{21}(L_1)\right)
\end{aligned}
\tag{279}
$$

We will now take the derivatives that we will result in the mass matrix

$$
\begin{aligned}
\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{\delta}_{11}}\right) = \ddot{\delta}_{11}\left(\int_0^{L_1} \rho_1 A_1\phi_{11}\phi_{11}dx + m_1\phi_{11}(L_1)\phi_{11}(L_1)\right) \\
+ \ddot{\delta}_{21}\left(\int_0^{L_1} \rho_1 A_1\phi_{11}\phi_{21}dx + m_1\phi_{21}(L_1)\phi_{21}(L_1)\right) + \\
+ \ddot{\delta}_{12}\left(\int_0^{L_2} \rho_2 A_2\phi_{12}\phi_{11}(L_1)dx + m_2\phi_{12}(L_2)\phi_{11}(L_1)\right) + \\
+ \ddot{\delta}_{11}\left(\int_0^{L_2} \rho_2 A_2\phi_{11}(L_1)\phi_{11}(L_1)dx + m_2\phi_{11}(L_1)\phi_{11}(L_1)\right) + \\
+ \ddot{\delta}_{22}\left(\int_0^{L_2} \rho_2 A_2\phi_{22}\phi_{11}(L_1)dx + m_2\phi_{22}(L_2)\phi_{11}(L_1)\right) \\
+ \ddot{\delta}_{21}\left(\int_0^{L_2} \rho_2 A_2\phi_{11}(L_1)\phi_{21}(L_1)dx + m_2\phi_{11}(L_1)\phi_{21}(L_1)\right)
\end{aligned}
\tag{280}
$$

$$
\begin{aligned}
\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{\delta}_{12}}\right) = \ddot{\delta}_{12}\left(\int_0^{L_2} \rho_2 A_2\phi_{12}\phi_{12}dx + m_2\phi_{12}(L_2)\phi_{12}(L_2)\right) \\
+ \ddot{\delta}_{11}\left(\int_0^{L_2} \rho_2 A_2\phi_{12}\phi_{11}(L_1)dx + m_2\phi_{12}(L_2)\phi_{11}(L_1)\right) + \\
+ \ddot{\delta}_{22}\left(\int_0^{L_2} \rho_2 A_2\phi_{12}\phi_{22}dx + m_2\phi_{12}(L_2)\phi_{22}(L_2)\right) + \\
+ \ddot{\delta}_{21}\left(\int_0^{L_2} \rho_2 A_2\phi_{12}\phi_{21}(L_1)dx + m_2\phi_{21}(L_1)\phi_{11}(L_1)\right)
\end{aligned}
\tag{281}
$$

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{\delta}_{21}}\right) = \ddot{\delta}_{21}\left(\int_0^{L_1} \rho_1 A_1 \phi_{21}\phi_{21} dx + m_1\phi_{21}(L_1)\phi_{21}(L_1)\right) + \ddot{\delta}_{11}\left(\int_0^{L_1} \rho_1 A_1 \phi_{11}\phi_{21}() dx + m_1\phi_{21}(L_1)\phi_{11}(L_1)\right) + \ddot{\delta}_{22}\left(\int_0^{L_2} \rho_2 A_2 \phi_{22}\phi_{21}(L_1) dx + m_2\phi_{22}(L_2)\phi_{21}(L_1)\right) + \ddot{\delta}_{21}\left(\int_0^{L_2} \rho_2 A_2 \phi_{11}(L_1)\phi_{11}(L_1) dx + m_2\phi_{11}(L_1)\phi_{11}(L_1)\right) + \ddot{\delta}_{12}\left(\int_0^{L_2} \rho_2 A_2 \phi_{12}\phi_{21}(L_1) dx + m_2\phi_{12}(L_2)\phi_{21}(L_1)\right) + \ddot{\delta}_{11}\left(\int_0^{L_2} \rho_2 A_2 \phi_{11}(L_1)\phi_{21}(L_1) dx + m_2\phi_{11}(L_1)\phi_{21}(L_1)\right) \quad (282)$$

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{\delta}_{22}}\right) = \ddot{\delta}_{22}\left(\int_0^{L_2} \rho_2 A_2 \phi_{22}\phi_{22} dx + m_2\phi_{22}(L_2)\phi_{22}(L_2)\right) + \ddot{\delta}_{21}\left(\int_0^{L_2} \rho_2 A_2 \phi_{22}\phi_{21}(L_1) dx + m_2\phi_{22}(L_2)\phi_{21}(L_1)\right) + \ddot{\delta}_{12}\left(\int_0^{L_2} \rho_2 A_2 \phi_{12}\phi_{22} dx + m_2\phi_{12}(L_2)\phi_{22}(L_2)\right) + \ddot{\delta}_{11}\left(\int_0^{L_2} \rho_2 A_2 \phi_{22}\phi_{11}(L_1) dx + m_2\phi_{12}(L_2)\phi_{11}(L_1)\right) \quad (283)$$

using the state vector: $\begin{Bmatrix} \delta_{11} \\ \delta_{21} \\ \delta_{12} \\ \delta_{22} \end{Bmatrix}$ we get the mass matrix with elements:

$$m_{11} = \int_0^{L_1} \rho_1 A_1 \phi_{11}\phi_{11} dx + m_1 \phi_{11}(L_1)\phi_{11}(L_1) + \int_0^{L_2} \rho_2 A_2 \phi_{11}(L_1)\phi_{11}(L_1) dx + m_2 \phi_{11}(L_1)\phi_{11}(L_1)$$

$$m_{12} = m_{21} = \int_0^{L_1} \rho_1 A_1 \phi_{11}\phi_{21} dx + m_1 \phi_{21}(L_1)\phi_{21}(L_1) + \int_0^{L_2} \rho_2 A_2 \phi_{11}(L_1)\phi_{21}(L_1) dx + m_2 \phi_{11}(L_1)\phi_{21}(L_1)$$

$$m_{13} = m_{31} = \int_0^{L_2} \rho_2 A_2 \phi_{12}\phi_{11}(L_1) dx + m_2 \phi_{12}(L_2)\phi_{11}(L_1)$$

$$m_{14} = m_{41} = \int_0^{L_2} \rho_2 A_2 \phi_{22}\phi_{11}(L_1) dx + m_2 \phi_{22}(L_2)\phi_{11}(L_1)$$

$$m_{22} = \int_0^{L_1} \rho_1 A_1 \phi_{21}\phi_{21} dx + m_1 \phi_{21}(L_1)\phi_{21}(L_1) + \int_0^{L_2} \rho_2 A_2 \phi_{21}(L_1)\phi_{21}(L_1) dx + m_2 \phi_{21}(L_1)\phi_{21}(L_1)$$

$$m_{23} = m_{32} \int_0^{L_2} \rho_2 A_2 \phi_{12}\phi_{21}(L_1) dx + m_2 \phi_{12}(L_2)\phi_{21}(L_1)$$

$$m_{24} = m_{42} = \int_0^{L_2} \rho_2 A_2 \phi_{22}\phi_{21}(L_1) dx + m_2 \phi_{22}(L_2)\phi_{21}(L_1)$$

$$m_{33} = \int_0^{L_2} \rho_2 A_2 \phi_{12}\phi_{12} dx + m_2 \phi_{12}(L_2)\phi_{12}(L_2)$$

$$m_{34} = m_{43} = \int_0^{L_2} \rho_2 A_2 \phi_{12}\phi_{22} dx + m_2 \phi_{12}(L_2)\phi_{22}(L_2)$$

$$m_{44} = \int_0^{L_2} \rho_2 A_2 \phi_{22}\phi_{22} dx + m_2 \phi_{22}(L_2)\phi_{22}(L_2) \quad (284)$$

similarly we can derive the stiffness matrix by using the potential energy of the system which is due to the deformations in the links. The deflection of link k can be expressed as $w_k''(x,t) = \sum_{i=1}^{modes} \delta_{ik}\phi_{ik}'' + \sum_{i=1}^{modes} \sum_{f=1}^{k-1} \delta_{if}\phi_{if}''(L_f)$ Therefore following an analogous derivation we get

the stiffness matrix with elements of:

$$k_{11} = \int_0^{L_1} EI_1\phi_{11}''\phi_{11}''dx + \int_0^{L_2} EI_2\phi_{11}''(L_1)\phi_{11}''(L_1)dx$$

$$k_{12} = k_{21} = \int_0^{L_1} EI_1\phi_{11}''\phi_{21}''dx + \int_0^{L_2} EI_2\phi_{11}''(L_1)\phi_{21}''(L_1)dx$$

$$k_{13} = k_{31} = \int_0^{L_2} EI_2\phi_{12}''\phi_{11}''(L_1)dx$$

$$k_{14} = k_{41} = \int_0^{L_2} EI_2\phi_{22}''\phi_{11}''(L_1)dx$$

$$k_{22} = \int_0^{L_1} EI_1\phi_{21}''\phi_{21}''dx + \int_0^{L_2} EI_2\phi_{21}''(L_1)\phi_{21}''(L_1)dx)$$

$$k_{23} = k_{32} \int_0^{L_2} EI_2\phi_{12}''\phi_{21}''(L_1)dx$$

$$k_{24} = k_{42} = \int_0^{L_2} EI_2\phi_{22}''\phi_{21}''(L_1)dx$$

$$k_{33} = \int_0^{L_2} EI_2\phi_{12}''\phi_{12}''dx$$

$$k_{34} = k_{43} = \int_0^{L_2} EI_2\phi_{12}''\phi_{22}''dx$$

$$k_{44} = \int_0^{L_2} EI_2\phi_{22}''\phi_{22}''dx \quad (285)$$

The natural frequencies are found after solving the eigenvalue problem $|K - \omega^2 M|$. Bear in mind that the mode shapes on the second link are offset by the values at the end of the fist link i.e $\phi_{i2}(0) = \phi_{i1}(L_1)$ and $\phi_{i2}''(0) = \phi_{i1}''(L_1)$.

Appendix D

MOTOR DATASHEET

RH Mini Series DC Servo Actuators

Technical Data

Rating:	Continuous
Excitation device:	RE Permanent magnet
Insulation:	Class B
Insulation voltage:	AC 500V, one minute
Insulation resistance:	100MΩ or more (DC 500V Megger)

Vibration:	2.5 g (5 . . . 400 Hz)
Shock:	< 30 g (11 ms)
Construction:	Totally enclosed
Lubrication:	Grease (SK-2)
Ambient temperature:	0 ~ 40°C
Ambient humidity:	20 ~ 80% (non condensing)

Table 2

Item	Actuator	RH-5A			RH-8D		RH-11D		RH-14D	
		8002	5502	4402	6006	3006	6001	3001	6002	3002
Rated Output Power [1]	W	1.5	1.7	1.4	8.6	6.2	13.6	12.3	20.3	18.5
Rated Voltage [1]	V	12	12	12	24	24	24	24	24	24
Rated Current [1]	A	0.5	0.5	0.5	1.0	0.8	1.3	1.3	1.8	1.8
Rated Output Torque [1] T_N	in-lb	1.4	2.6	2.6	12	17	19	34	28	52
	Nm	0.16	0.29	0.29	1.4	2.0	2.2	3.9	3.2	5.9
Rated Output Speed [1] n_N	rpm	88	55	44	60	30	60	30	60	30
Max. Continuous Stall	in-lb	2.1	3.5	3.8	13	20	22	39	48	69
Torque [1] [2]	Nm	0.24	0.39	0.43	1.5	2.3	2.5	4.4	5.4	7.8
Peak Current [1] [2]	A	0.83	0.78	0.77	1.6	1.1	2.4	2.1	5.4	4.1
Maximum Output Torque [1] [2] T_m	in-lb	3.5	5.2	6.1	24	31	43	69	122	174
	Nm	0.39	0.59	0.69	2.7	3.5	4.9	7.8	14	20
Maximum Output Speed [1]	rpm	180	110	90	100	50	100	50	100	50
Torque Constant	in-lb/A	6	10	12	19	37	22	43	26	51
	Nm/A	0.69	1.11	1.38	2.10	4.20	2.46	4.91	2.92	5.76
Voltage Constant (B.E.M.F.)	V/rpm	0.08	0.12	0.15	0.22	0.44	0.26	0.50	0.30	0.60
Inertia at Output Shaft [3]	in-lb-sec^2	0.006	0.014	0.022	0.033	0.13	0.095	0.38	0.18	0.72
	kgm^2 x10^3	0.63	1.6	2.5	3.7	15.0	11.0	43.0	21.6	81.6
Mechanical Time Constant	msec	13.3	13.3	13.3	8.5	8.5	8.5	8.5	7.0	7.0
Rated Power Rate [1]	kW/sec	0.039	0.055	0.034	0.51	0.26	0.43	0.36	0.51	0.42
Thermal Time Constant [1]	min	5.2	5.2	5.2	9	9	10	10	11	11
Thermal Resistance [1]	°C/W	11.4	11.4	11.4	4.2	4.2	3.3	3.3	2.8	2.8
Gear Ratio	1:R	1:50	1:80	1:100	50	100	50	100	50	100
Maximum Radial Load [6]	lb	13	13	13	44	44	55	55	88	88
	N	59	59	59	196	196	245	245	392	392
Maximum Axial Load	lb	7	7	7	22	22	44	44	88	88
	N	29	29	29	98	98	196	196	392	392
Motor Rated Output [1] [5]	W	(2.6)	(2.6)	(2.6)	(10)	(10)	(20)	(20)	(30)	(30)
Motor Rated Speed [1]	rpm	4500	4500	4500	3000	3000	3000	3000	3000	3000
Armature Resistance	Ω	8.6	8.6	8.6	10	10	4.7	4.7	2.7	2.7
Armature Inductance	mH	2.7	2.7	2.7	2.2	2.2	1.6	1.6	1.1	1.1
Electrical Time Constant	ms	0.31	0.31	0.31	0.22	0.22	0.34	0.34	0.41	0.41
No-Load Running Current[4]	A	0.27	0.24	0.28	0.38	0.36	0.61	0.55	0.89	0.91
Actuator Accuracy	arc-min	4.5	4.5	4.5	2.5	2.5	2.0	2.0	2.0	2.0
Actuator Repeatability	arc-sec	±90	±90	±90	±60	±60	±60	±60	±60	±60
Servo Drive Combinations	DC 20V	DCJ-055-09, DDP-090-09, DEP-090-09			–	–	–	–	–	–
	DC 24V	–			DCJ-055-09, DDP-090-09, DEP-090-09		DCJ-055-09, DDP-090-09, DEP-090-09		DCJ-055-09, DDP-090-09, DEP-090-09	
	AC 100V	HS-360-1A			HS-360-1B		HS-360-1C		HS-360-1D	

Additional information

* Actuator specifications show output characteristics, including gear efficiency.
* All specifications are applicable for actuators mounted on aluminum heat sink of the following sizes:
RH-5: 100 x 100 x 3 mm,
RH-8, 11, 14: 150 x 150 x 6 mm.

Please Note:

[1] The values are for saturated actuator temperature. Other values (not marked with [1]) are for actuator temperature of 20°C.

[2] The values given represent an upper limit and actual load values should be lower.

[3] The tabulated value is the moment of inertia reflected to the output shaft resulting from the sum of the motor inertia and the gear inertia.

[4] Values are for rated output speed.

[5] Values are for reference only.

[6] Cantilevered load applied at the midpoint of the shaft extension.

Appendix E

ACCELEROMETER DATASHEET

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 = $1/(2 \times \pi \times 32\ k\Omega \times C)$. 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.

Appendix F

TEST BED EXTRA PLOTS

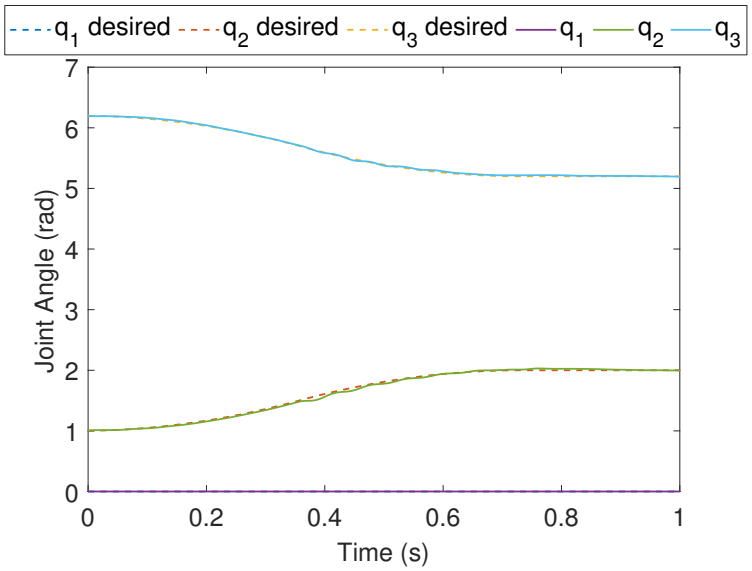


Figure 124: Desired and Measured Joint Angles For Move 2

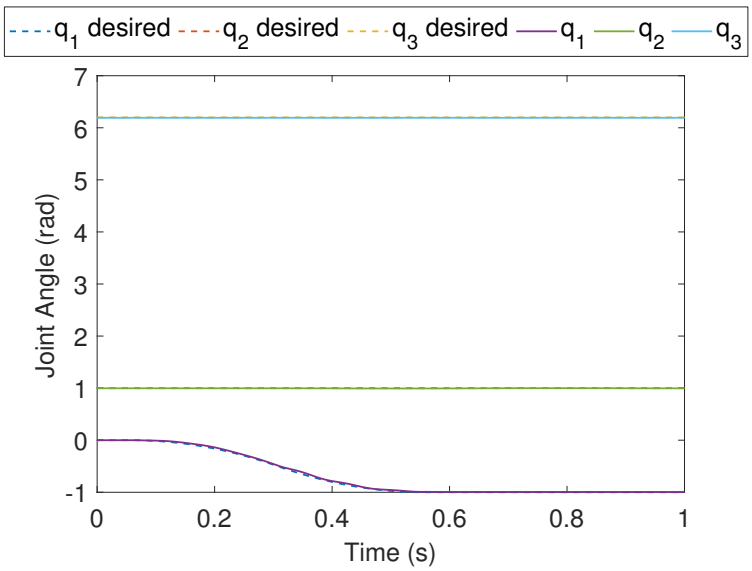
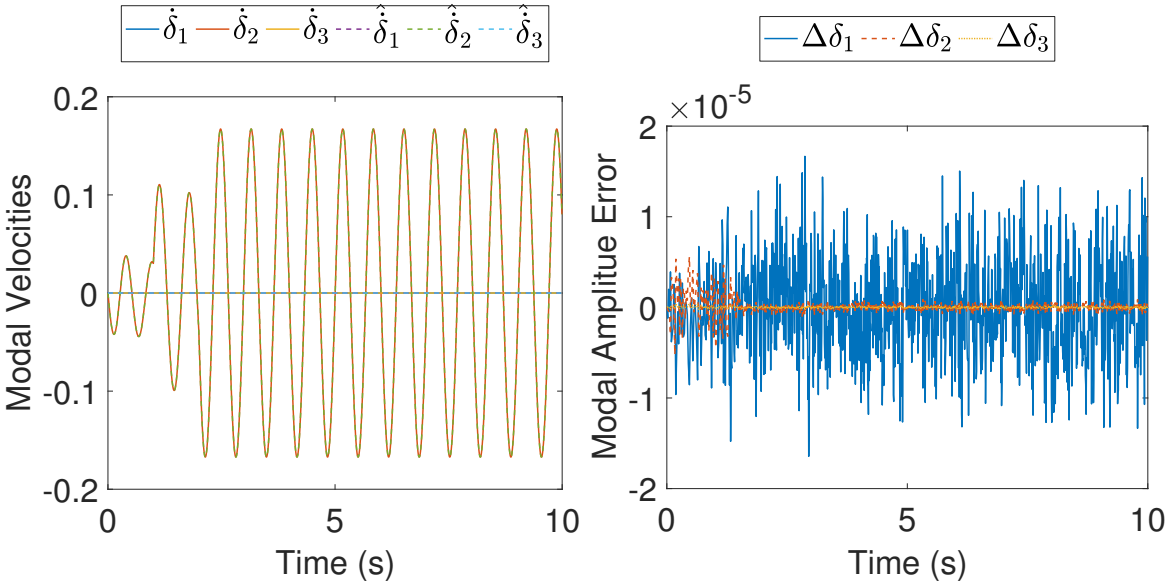


Figure 125: Desired and Measured Joint Angles For Move 3

Appendix G

ESTIMATOR VERIFICATION EXTRA PLOTS

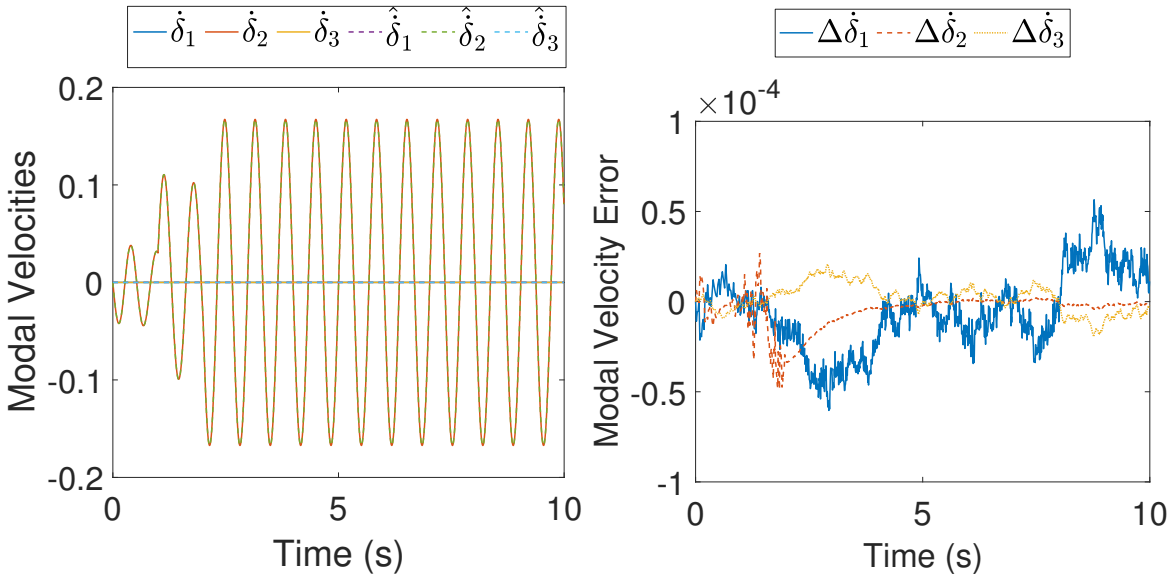
G.1 Acceleration Feedback Extra Plots



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

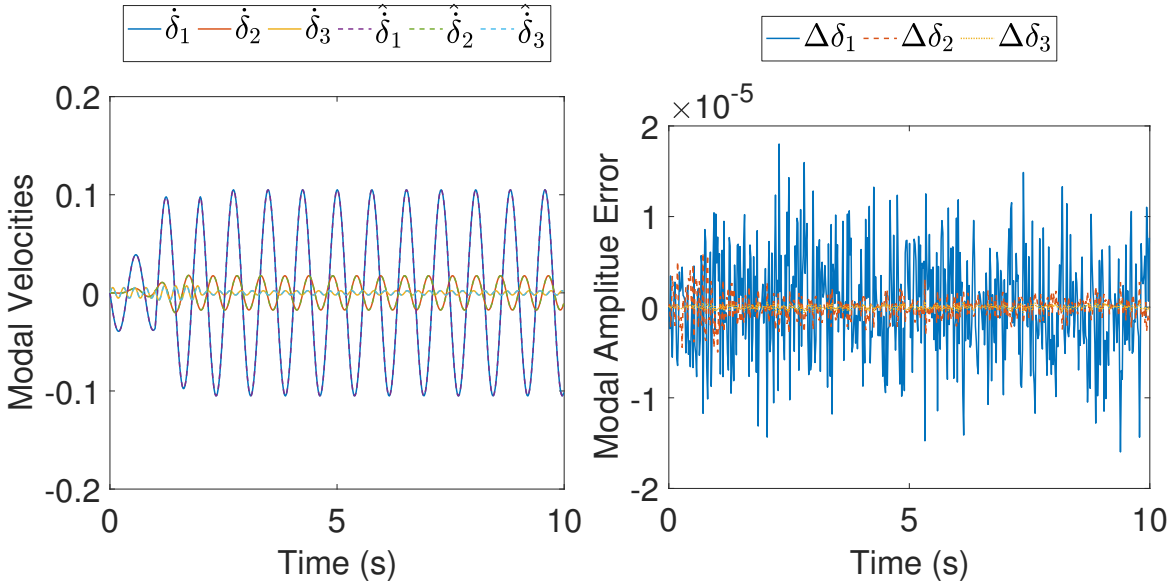
Figure 126: Modal Amplitudes for Move 2



(a) Modal Velocities and Estimates

(b) Modal Velocity Error

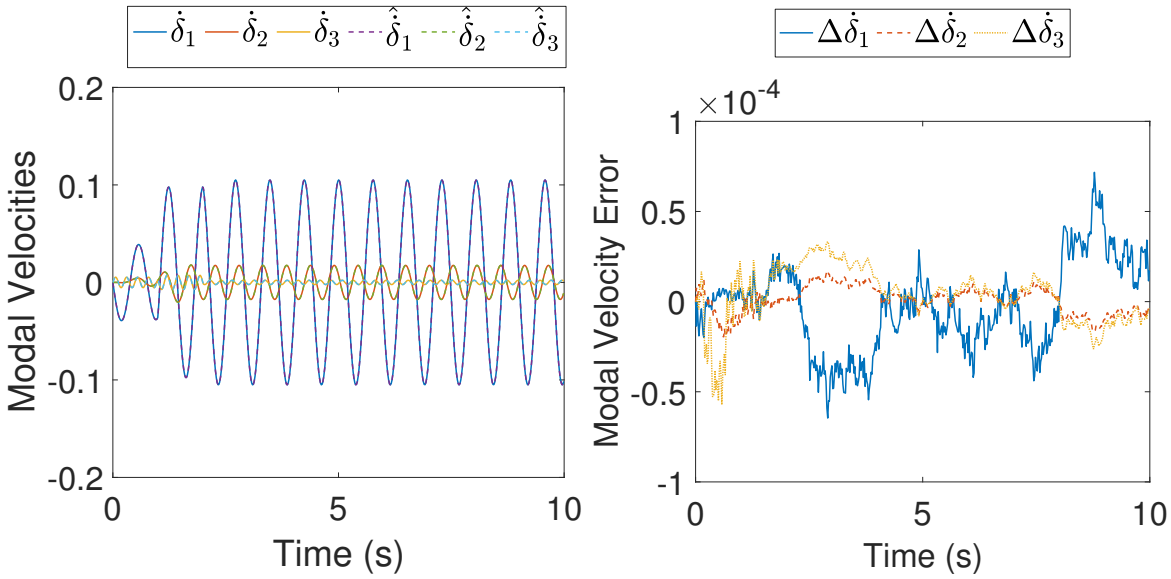
Figure 127: Modal Velocities for Move 2



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 128: Modal Amplitudes for Move 3

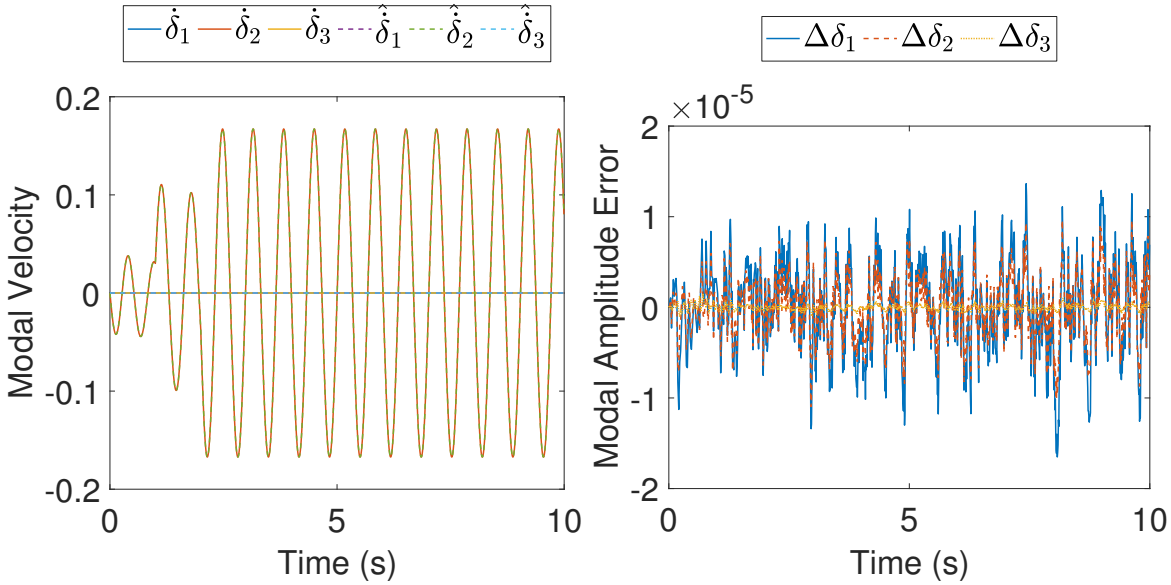


(a) Modal Velocities and Estimates

(b) Modal Velocity Error

Figure 129: Modal Velocities for Move 3

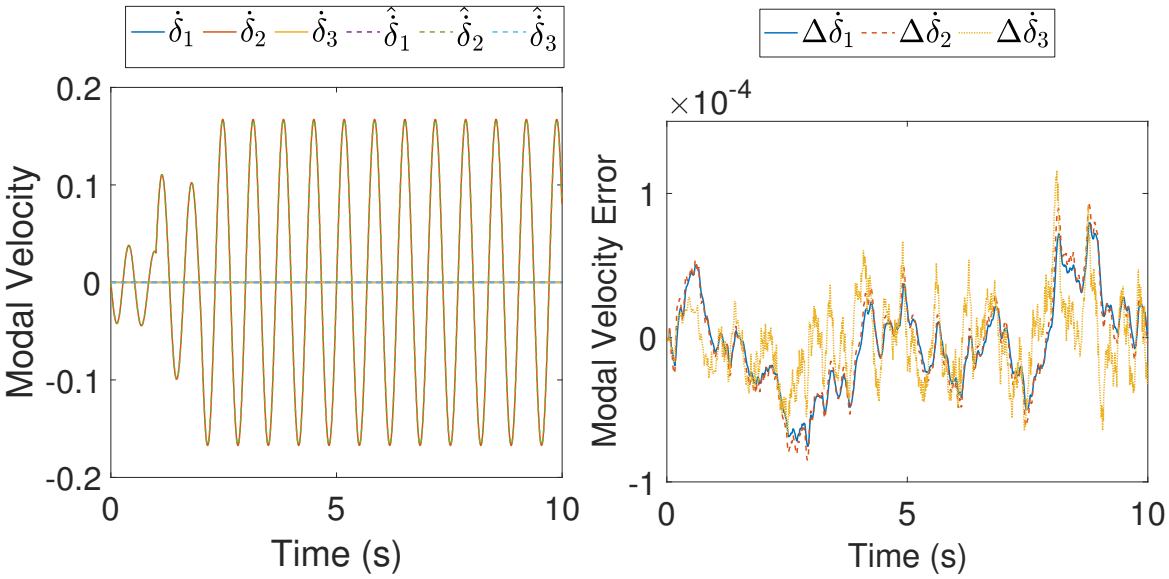
G.2 Strain Feedback Extra Plots



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

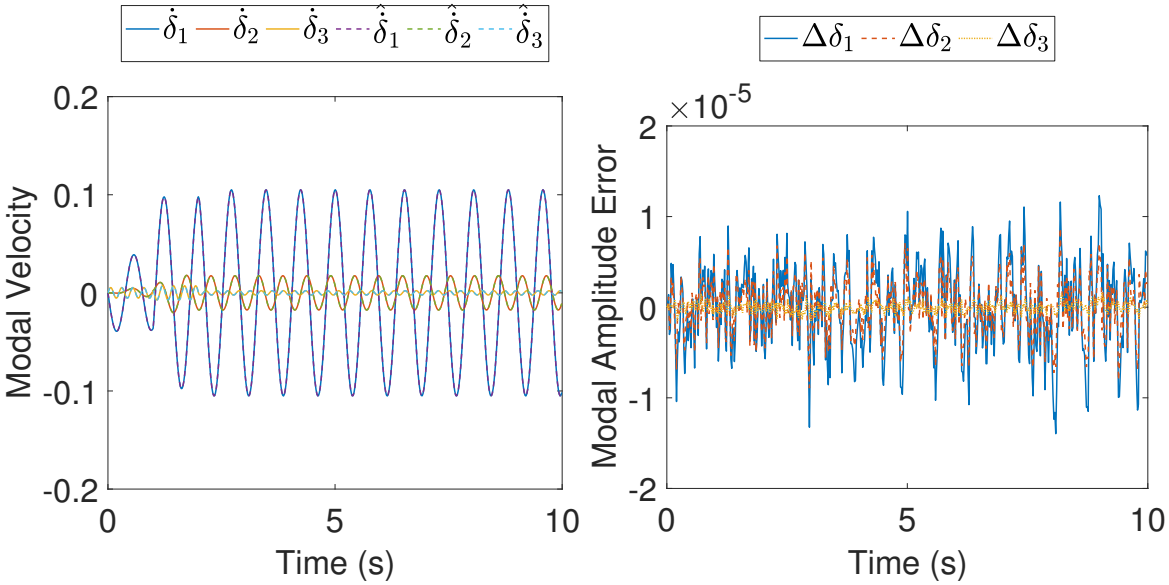
Figure 130: Modal Amplitudes for Move 2



(a) Modal Velocities and Estimates

(b) Modal Velocity Error

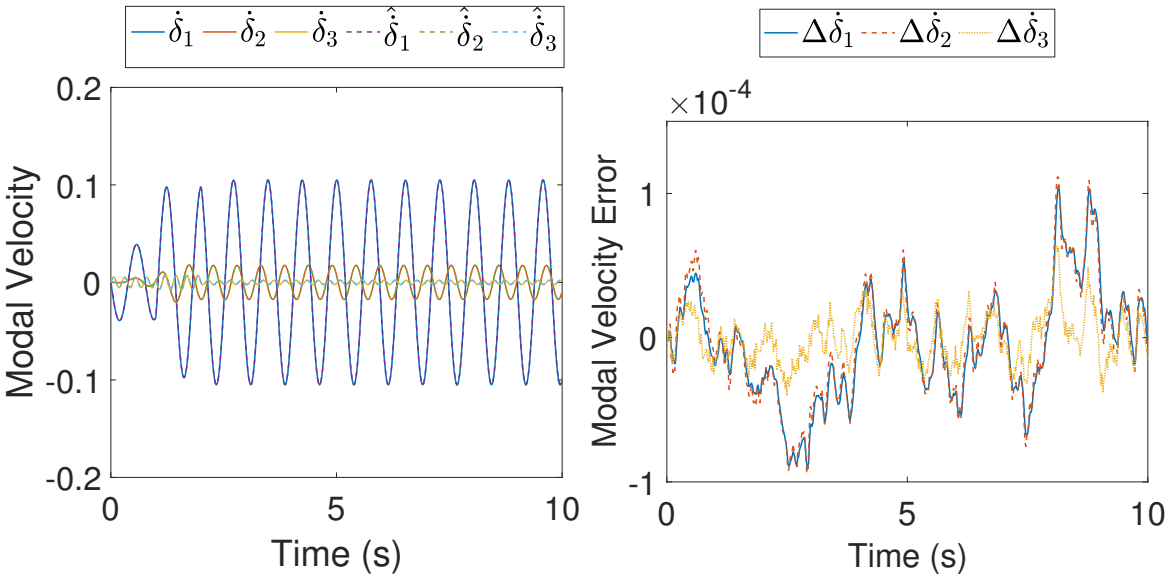
Figure 131: Modal Velocities for Move 2



(a) Modal Amplitude and Estimates

(b) Modal Amplitude Error

Figure 132: Modal Amplitudes for Move 3



(a) Modal Velocities and Estimates

(b) Modal Velocity Error

Figure 133: Modal Velocities for Move 3

G.3 Estimator Verification Experiments Extra Plots

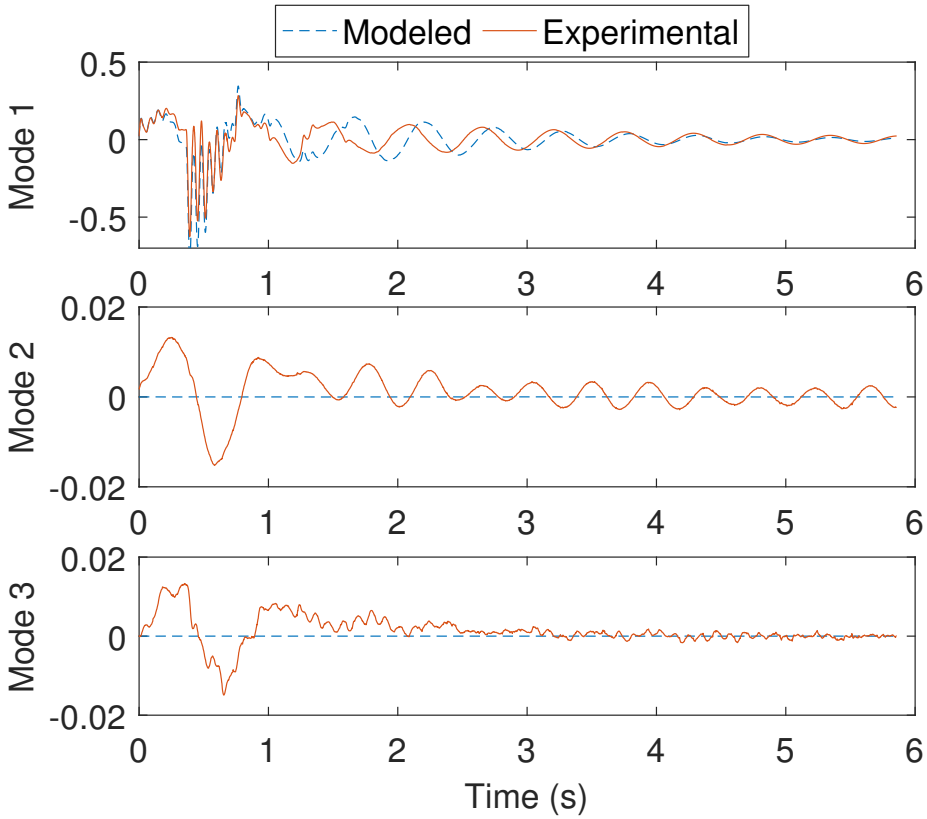


Figure 134: Modeled and Experimental Modal Velocities for Move 2

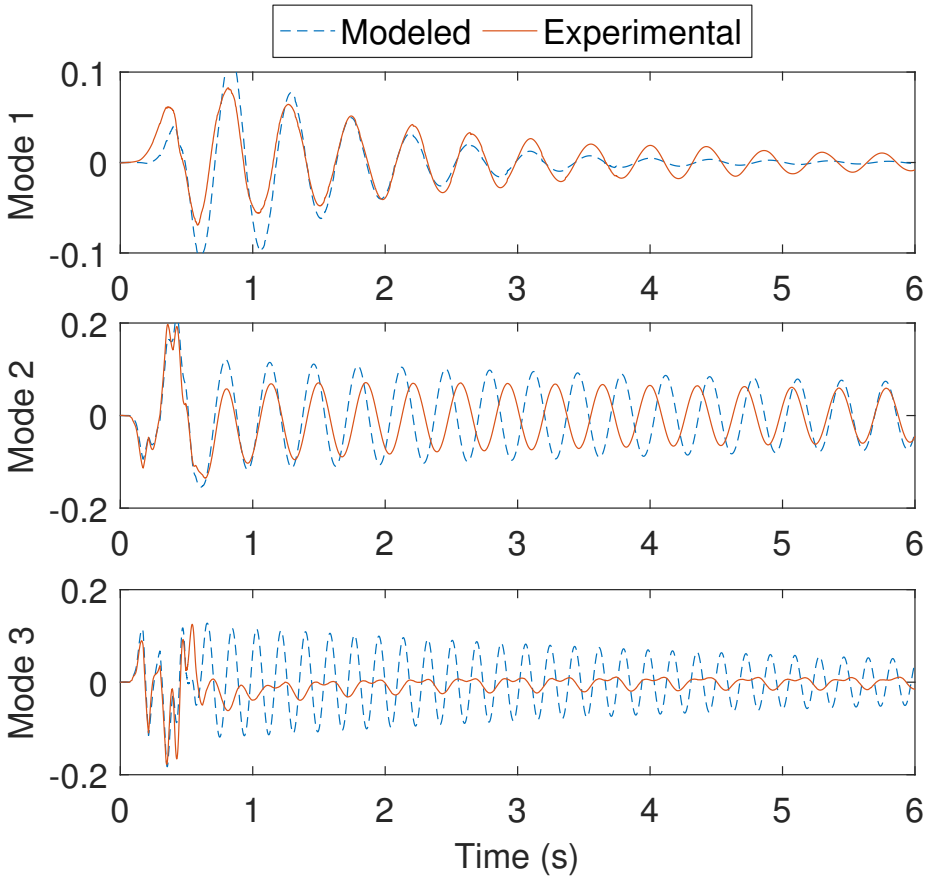


Figure 135: Modeled and Experimental Modal Velocities for Move 3

Appendix H

CONTROL EXTRA PLOTS

H.1 Inversion Based Controller Extra Plots

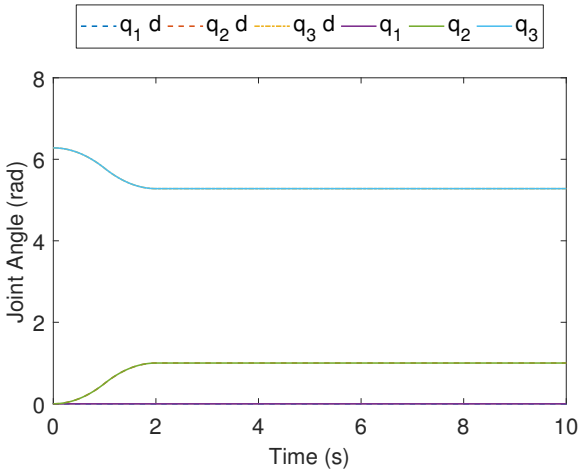


Figure 136: Joint Angles For Move 2

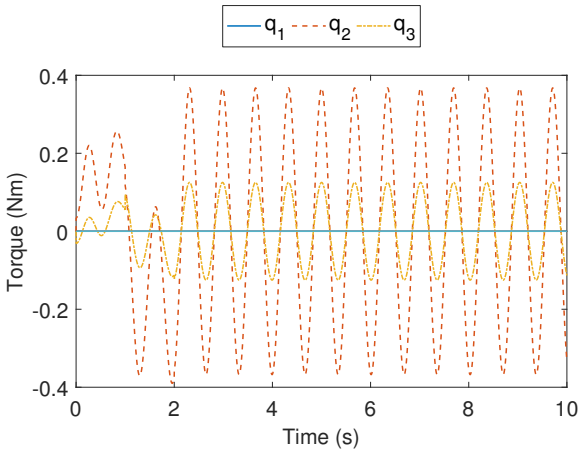
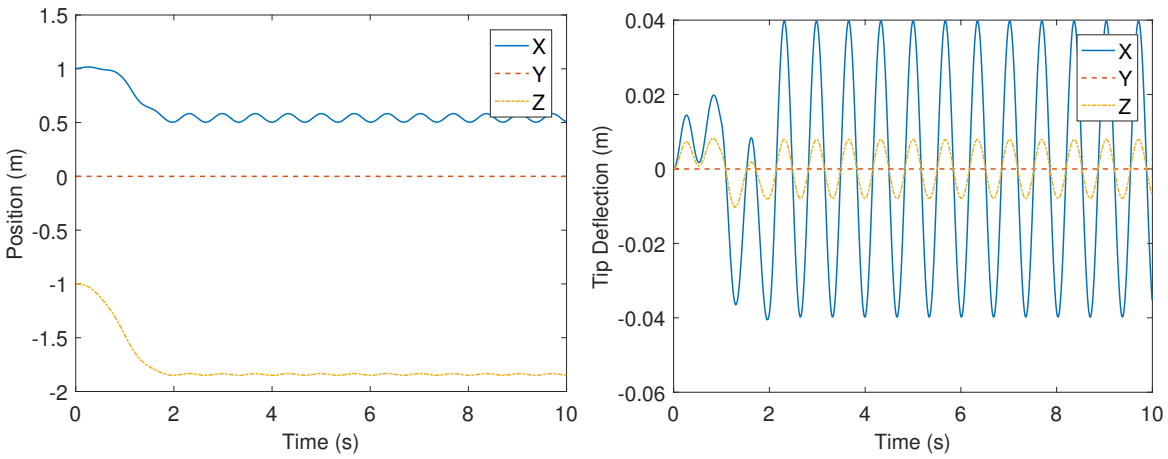


Figure 137: Torque For Move 2



(a) Tip Position

(b) Tip Deflection

Figure 138: Tip Response to Move 2

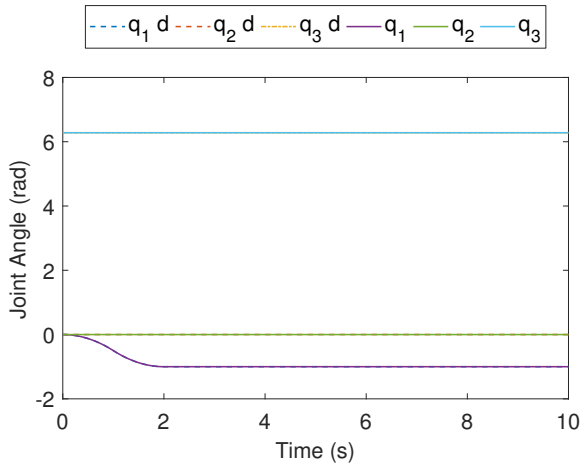


Figure 139: Joint Angles For Move 3

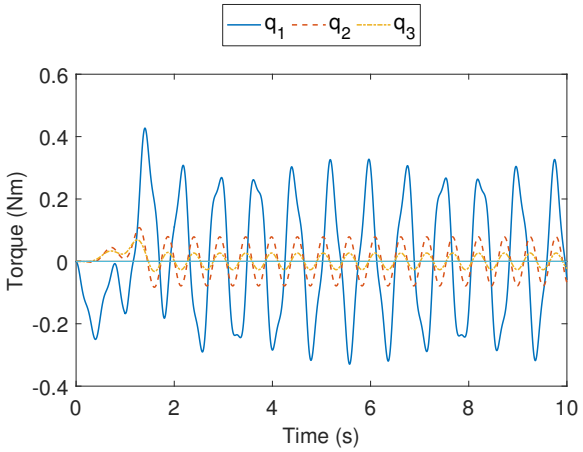
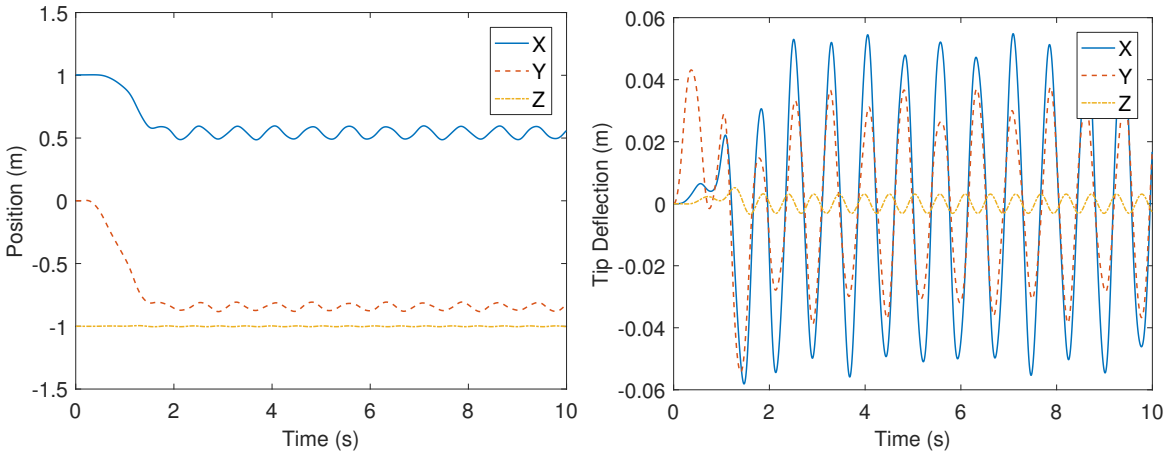


Figure 140: Torque For Move 3



(a) Tip Position

(b) Tip Deflection

Figure 141: Tip Response to Move 3

H.2 Singular Pertubation Based Controller Extra Plots

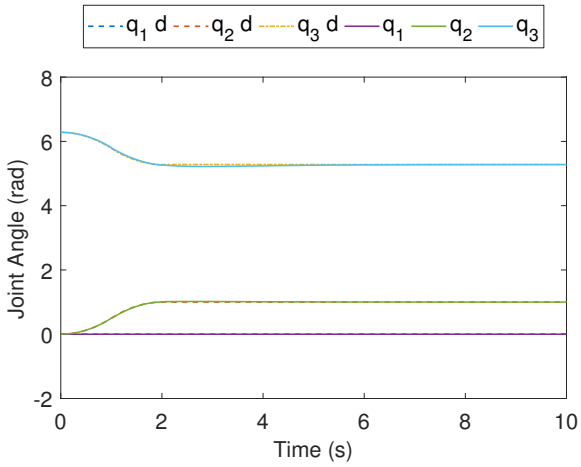


Figure 142: Joint Angles For Move 2

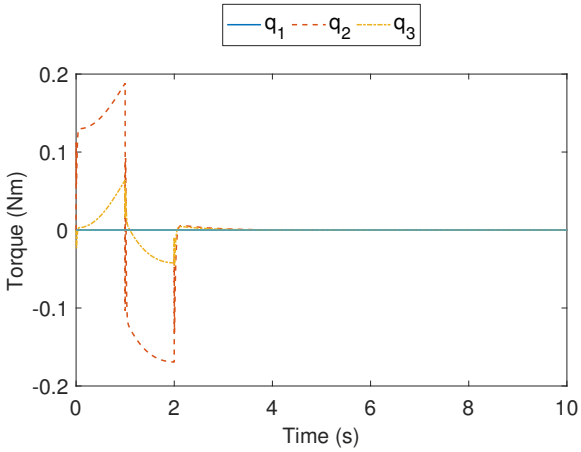
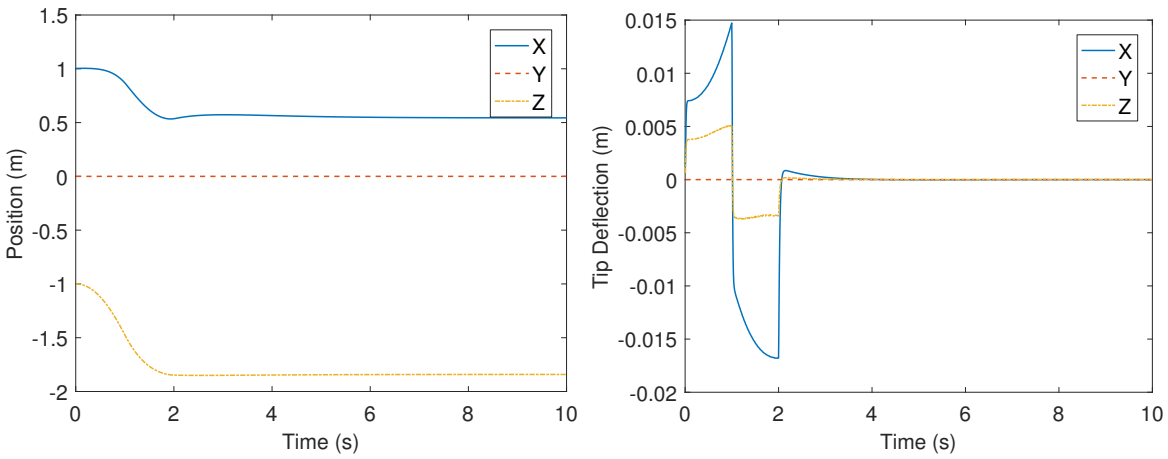


Figure 143: Torque For Move 2



(a) Tip Position

(b) Tip Deflection

Figure 144: Tip Response to Move 2

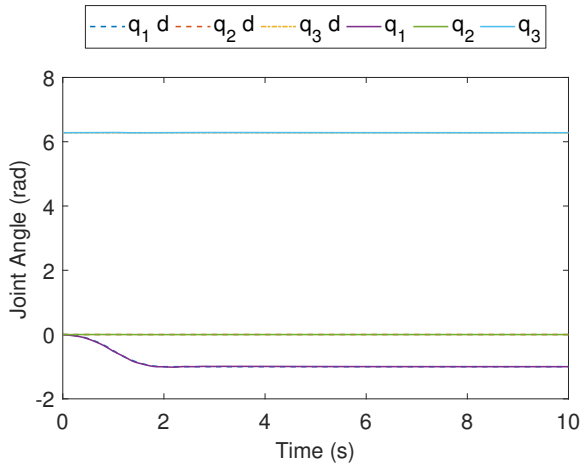


Figure 145: Joint Angles For Move 3

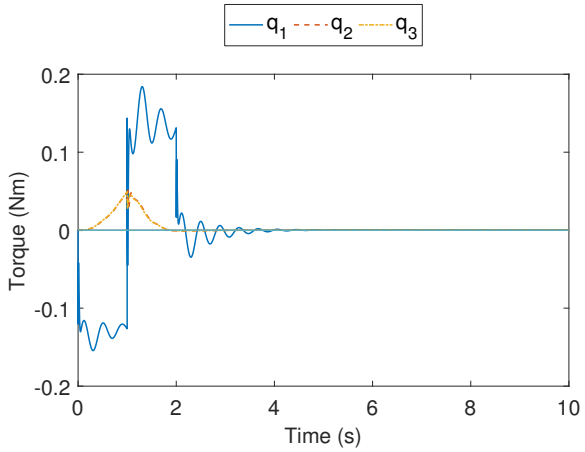
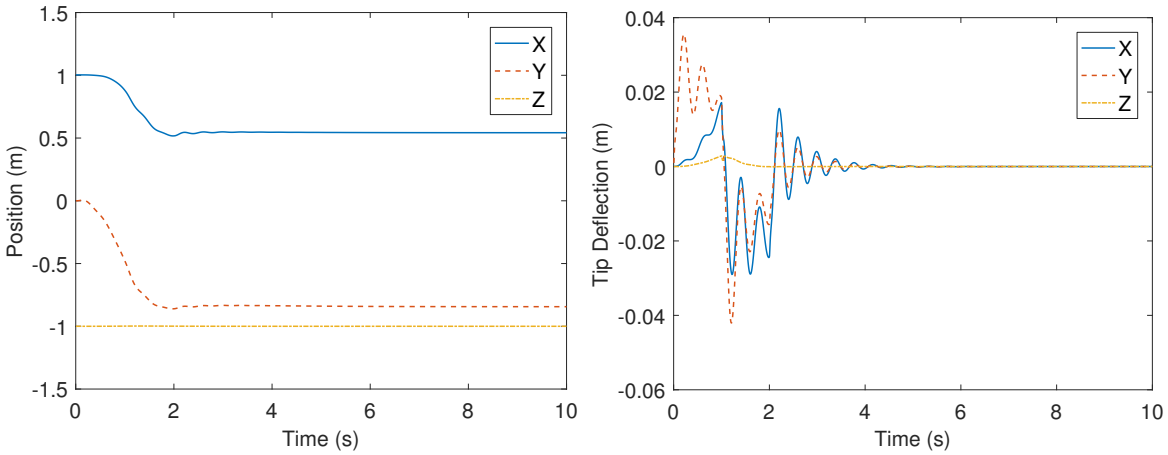


Figure 146: Torque For Move 3



(a) Tip Position

(b) Tip Deflection

Figure 147: Tip Response to Move 3

REFERENCES

[1] Abdollahi, F., Talebi, A. H., and Patel, V. R., "A stable neural network-based observer with application to flexible-joint manipulators," *Neural Networks, IEEE Transactions on*, vol. 17, pp. 118–129, 2004.

[2] Alberts, E. T., Dickerson, L. S., and Book, J. W., "Modeling and control of flexible manipulators," 1984.

[3] Alberts, E. T., Xia, H., and Chen, Y., "Dynamic analysis to evaluate viscoelastic passive damping augmentation for the space shuttle remote manipulator system," *Journal of Dynamic Systems, Measurement, and Control*, vol. 114, pp. 468–475, 1990.

[4] Albu-Schäffer, A., Ott, C., and Hirzinger, G., "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *The International Journal of Robotics Research*, vol. 26, pp. 23–39, 2005.

[5] Altintas, Y. and Khoshdarregi, R. M., "Contour error control of cnc machine tools with vibration avoidance," *CIRP annals-manufacturing technology*, vol. 61, pp. 335–338, 2011.

[6] Arteaga, A. M., "Tracking control of flexible robot arms with a nonlinear observer," *Automatica*, vol. 36, pp. 1329–1337, 1993.

[7] Arteaga, A. M., "On the properties of a dynamic model of flexible robot manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 120, pp. 8–14, Mar 1998. Many proofs and properties of the matrices.

[8] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T., "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[9] Asada, H. and Ma, Z.-D., "Inverse dynamics of flexible robots; modeling and recursive computation using virtual rigid link coordinate systems," *American Control Conference, 1989*, pp. 2352–2359.

[10] Balas, J. M., "Feedback control of flexible systems," *Automatic Control, IEEE Transactions on*, vol. 23, pp. 673–679, 1977.

[11] Banerjee, K. A. and Singhose, E. W., "Command shaping in tracking control of a two-link flexible robot," *Journal of Guidance, control, and Dynamics*, vol. 21, pp. 1012–1015, 1996.

[12] Beargie, M. A. and Book, J. W., "Sliding mode control of a non-collocated flexible system," 2002.

[13] Book, J. W., "Analysis of massless elastic chains with servo controlled joints," *Journal of Dynamic Systems*, vol. 101, pp. 178–192, 1977.

[14] Book, J. W., "Recursive lagrangian dynamics of flexible manipulator arms via transformation matrices," tech. rep., 1982.

[15] Book, J. W., "Controlled motion in an elastic world," *ASME Journal of Dynamic Systems Measurement and Control*, vol. 115, 1990.

[16] Book, J. W., Maizza-Neto, O., and Whitney, E. D., "Feedback control of two beam, two joint systems with distributed flexibility," *American Society of Mechanical Engineers, Winter Annual Meeting*.

[17] Book, J. W. 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, pp. 245–249, 1982.

[18] Book, W. J., *Modeling, design and control of flexible manipulator arms.* PhD thesis, Massachusetts Institute of Technology, 1974.

[19] Botsali, M. F., Kalyoncu, M., Tinkir, M., and Onen, U., "Fuzzy logic trajectory control of flexible robot manipulator with rotating prismatic joint," pp. 35–39.

[20] Brüls, O., Duysinx, P., and Golinval, J.-C., "The global modal parameterization for non-linear model-order reduction in flexible multibody dynamics," *International journal for numerical methods in engineering*, vol. 69, pp. 948–977, 2005.

[21] Cannon, H. R. and Schmitz, E., "Initial experiments on the end-point control of a flexible one link robot," *Int. J. of Robotics Research*, vol. 3, 1983.

[22] Carrera, E. and Serna, A. M., "Inverse dynamics of flexible robots," *Mathematics and Computers in Simulation*, vol. 41, pp. 485–508, 1994.

[23] Cetinkunt, S. and Book, J. W., "Performance of lightweight manipulators under joint variable feedback control: Analytical study of limitations," pp. 543–548, 1987.

[24] Cetinkunt, S. and Book, J. W., "Flexibility effects on the control system performance of large scale robotic manipulators," *Journal of Astronomical Sciences*, vol. 38, pp. 531–556, 1989.

[25] Chalhoub, G. N. and Kfoury, A. G., "Development of a robust nonlinear observer for a single-link flexible manipulator," *Nonlinear Dynamics*, vol. 39, pp. 217–233, 2003.

[26] Chaoui, H., Gueaieb, W., Yagoub, C. M., and Sicard, P., "Hybrid neural fuzzy sliding mode control of flexible-joint manipulators with unknown dynamics," *IEEE Industrial Electronics, IECON 2006-32nd Annual Conference on*, pp. 4082–4087, 2005.

[27] Chatterjee, P. and Bryant, M., "Transfer matrix modeling of a tensioned piezo-solar hybrid energy harvesting ribbon," *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, pp. 94310D–94310D, 2014.

[28] Chen, F. and Dunnigan, W. M., "Comparative study of a sliding-mode observer and kalman filters for full state estimation in an induction machine," pp. 12–12, 2000.

[29] Chen, W., "Dynamic modeling of multi-link flexible robotic manipulators," *Computers & Structures*, vol. 79, pp. 183–195, 1999.

[30] Chen, W., "Dynamic modeling of multi-link flexible robotic manipulators," *Computers & Structures*, vol. 79, Jan 2001.

[31] Cheong, J., Chung, K. W., and Youm, Y., "Two-step controller for 3-d flexible link manipulators: Bandwidth modulation and modal feedback approach," *Journal of dynamic systems, measurement, and control*, vol. 124, pp. 566–574, 2001. very useful.

[32] Cheong, J., Chung, W., and Youm, Y., "Bandwidth modulation of rigid subsystem for the class of flexible robots," *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2, pp. 1478–1483, 1997.

[33] Cheong, J., Youm, Y., and Chung, K. W., "Accessibility and identifiability of horizontal vibration in 3-d two-link flexible robots: system mode approach," *Journal of sound and vibration*, vol. 269, pp. 489–509, 2002.

[34] Chien, M.-C. and Huang, A.-C., "Adaptive control for flexible-joint electrically driven robot with time-varying uncertainties," *IEEE Transactions on Industrial Electronics*, vol. 54, pp. 1032–1038, 2005.

[35] Dado, M., "A generalized approach for forward and inverse dynamics of elastic manipulators," *IEEE International Conference on Robotics and Automation. Proceedings.*, vol. 3, pp. 359–364, 1983.

[36] De Luca, A. and Book, W., *Handbook of Robotics*, ch. Robots with flexible elements, pp. 287–319. Springer, 2006.

[37] De Luca, A. and Flacco, F., "A pd-type regulator with exact gravity cancellation for robots with flexible joints," 2011 IEEE International Conference on Robotics and Automation, pp. 317–323, IEEE.

[38] De Luca, A., Panzieri, S., and Ulivi, G., "Stable inversion control for flexible link manipulators," *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 1, pp. 799–805, 1996.

[39] De Luca, A. and Siciliano, B., "Explicit dynamic modeling of a planar two-link flexible manipulator," *Decision and Control, 1990., Proceedings of the 29th IEEE Conference on*, pp. 528–530, 1988.

[40] De Luca, A. and Siciliano, B., "Regulation of flexible arms under gravity," *Robotics and Automation, IEEE Transactions on*, vol. 9, pp. 463–467, 1992.

[41] Denavit, J., "A kinematic notation for lower-pair mechanisms based on matrices.," *Trans. of the ASME. Journal of Applied Mechanics*, vol. 22, pp. 215–221, 1954.

[42] di Castri, C. and Messina, A., "Exact modeling for control of flexible manipulators," *Journal of Vibration and Control*, vol. 18, pp. 1526–1551, 2007.

[43] Dogan, M. and Istefanopulos, Y., "Optimal nonlinear controller design for flexible robot manipulators with adaptive internal model," *Control Theory Applications, IET*, vol. 1, pp. 770–778, May 2007.

[44] Dong, X.-J., Meng, G., and Peng, J.-C., "Vibration control of piezoelectric smart structures based on system identification technique: Numerical simulation and experimental study," *Journal of Sound and Vibration*, vol. 297, pp. 680–693, 2004.

[45] Dubowsky, S., Gu, Y. P., and Deck, F. J., "The dynamic analysis of flexibility in mobile robotic manipulator systems," Proc. VIII World Congress on the Theory of Machines and Mechanisms, pp. 26–31.

[46] Dubus, G., David, O., and Measson, Y., "Vibration control of a flexible arm for the iter maintenance using unknown visual features from inside the vessel," *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 5697–5704, 2007.

[47] Dwivedy, K. S. and Eberhard, P., "Dynamic analysis of flexible manipulators, a literature review," *Mechanism and machine theory*, vol. 41, pp. 749–777, 2003.

[48] Etxebarria, V., Sanz, A., and Lizarraga, I., "Real-time experimental control of a flexible robotic manipulator using a composite approach," *Control Applications, 2004. Proceedings of the 2004 IEEE International Conference on*, vol. 2, pp. 955–960, 2002.

[49] Feliu, V., García, A., and Somolinos, A. J., "Gauge-based tip position control of a new three-degree-of-freedom flexible robot," *The International Journal of Robotics Research*, vol. 20, pp. 660–675, 1999.

[50] Feliu, V. and Ramos, F., "Strain gauge based control of single-link flexible very lightweight robots robust to payload changes," *Mechatronics*, vol. 15, pp. 547–571, 2003.

[51] Feliu, V., Rattan, S. K., and Brown, H. B. J., "Adaptive control of a single-link flexible manipulator," *Control Systems Magazine, IEEE*, vol. 10, pp. 29–33, 1988.

[52] Forbes, R. J. and Damaren, J. C., "Single-link flexible manipulator control accommodating passivity violations: Theory and experiments," *Control Systems Technology, IEEE Transactions on*, vol. 20, pp. 652–662, 2011.

[53] Franke, R., Malzahn, J., Nierobisch, T., Hoffmann, F., and Bertram, T., "Vibration control of a multi-link flexible robot arm with fiber-bragg-grating sensors," Robotics and Automation, 2009. ICRA '09. IEEE International Conference on, pp. 3365–3370, IEEE.

[54] FUKUDA, T., ARAI, F., HOSOGAI, H., and YAJIMA, N., "Flexibility control of flexible structures. modeling and control method of bending-torsion coupled vibrations.," *JSME international journal. Ser. 3, Vibration, control engineering, engineering for industry*, vol. 31, pp. 575–582, 1987.

[55] Ge, S. S., Lee, H. T., Gong, Q. J., and Wang, P. Z., "Model-free controller design for a single-link flexible smart materials robot," *International Journal of Control*, vol. 73, pp. 531–544, 1998.

[56] Ginsberg, H. J., *Mechanical and Structural Vibrations*. John Wiley and Sons, inc., 2000.

[57] Gouliaev, I. V. and Zavrazhina, V. T., "Dynamics of a flexible multi-link cosmic robot-manipulator," *Journal of Sound and vibration*, vol. 243, pp. 641–657, 1999.

[58] Green, A. and Sasiadek, Z. J., "Inverse dynamics and fuzzy repetitive learning flexible robot control," *IFAC Proceedings Volumes*, vol. 35, pp. 139–144, 2000.

[59] Gurses, K., Buckham, J. B., and Park, J. E., "Vibration control of a single-link flexible manipulator using an array of fiber optic curvature sensors and pzt actuators," *Mechatronics*, vol. 19, pp. 167–177, 2007.

[60] Haddab, Y., Chen, Q., and Lutz, P., "Improvement of strain gauges micro-forces measurement using kalman optimal filtering," *Mechatronics*, vol. 19, pp. 457–462, 2007.

[61] Haddadin, S., Albu-Schäffer, A., and Hirzinger, G., "Safe physical human-robot interaction: measurements, analysis and new insights," pp. 395–407, 2009.

[62] Hastings, G. G. and Book, J. W., "A linear dynamic model for flexible robotic manipulators," *Control Systems Magazine, IEEE*, vol. 7, pp. 61–64, 1986.

[63] Hastings, G. G. and Book, J. W., "Reconstruction and robust reduced-order observation of flexible variables," Nov 1986.

[64] Henriksson, R., Norrlöf, M., Moberg, S., Wernholt, E., and Schön, B. T., "Experimental comparison of observers for tool position estimation of industrial robots," *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pp. 8065–8070, 2007.

[65] Henriksson, R., Norrlöf, M., Moberg, S., Wernholt, E., and Schön, T., "Experimental comparison of observers for tool position estimation of industrial robots," *Linköping University Electronic Press*, 2007.

[66] Hermann, R. and Krener, A., "Nonlinear controllability and observability," *IEEE Transactions on automatic control*, vol. 22, pp. 728–740, 1975.

[67] Huang, A.-C. and Chen, Y.-C., "Adaptive sliding control for single-link flexible-joint robot with mismatched uncertainties," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 5, pp. 770–775, 2004.

[68] Husain, R. A., Khairudin, M., and Mohamed, Z., "Dynamic model and robust control of flexible link robot manipulator," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 9, Apr 2013.

[69] IEEE, *Comparison of Input Shaping Techniques for Speed-Critical Multi-Mode Flexible Systems*, IEEE Conf. on Control Applications, 1996.

[70] Jiang, X., Yabe, Y., Konno, A., and Uchiyama, M., "Vibration suppression control of a flexible arm using image features of unknown objects," *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp. 3783–3788, 2006.

[71] Jonker, J. and Aarts, R., "A perturbation method for dynamic analysis and simulation of flexible manipulators," *Multibody System Dynamics*, vol. 6, no. 3, pp. 245–266, 2001.

[72] Jung, J., Huh, K., Fathy, K. H., and Stein, L. J., "Optimal robust adaptive observer design for a class of nonlinear systems via an h-infinity approach," *American Control Conference, 2006.*

[73] Karandikar, D. and Bandyopadhyay, B., "Sliding mode control of single link flexible manipulator," in *Industrial Technology 2000. Proceedings of IEEE International Conference on*, vol. 1, pp. 712–717, IEEE, 2000.

[74] Khalil, H. K., *Noninear Systems.* Prentice-Hall, New Jersey, 1996.

[75] Kokotović, P., Khalil, K. H., and O'reilly, J., *Singular perturbation methods in control: analysis and design.* SIAM, 1997.

[76] Konno, A. and Uchiyama, M., "Vibration suppression control of spatial flexible manipulators," *Control Engineering Practice*, vol. 3, pp. 1315–1321, 1993.

[77] Konno, A., Uchiyama, M., and Murakami, M., "Configuration-dependent vibration controllability of flexible-link manipulators," *The International Journal of Robotics Research*, vol. 16, pp. 567–576, 1995.

[78] Krauss, R., *An Improved Technique for Modeling and Control of Flexible Structures.* PhD thesis, Georgia Institute of Technology, 2005.

[79] Kuo, Y. K. and Lin, J., "Fuzzy logic control for flexible link robot arm by singular perturbation approach," *Applied Soft Computing*, vol. 2, pp. 24–38, 2000.

[80] Lee, D. J. and Wang, B.-L., "Optimal control of a flexible robot arm," *Computers & structures*, vol. 29, pp. 459–467, 1987.

[81] Lee, W. J. and Book, J. W., "Efficient dynamic models for flexible robots," pp. 8–10, 1988.

[82] Lertpiriyasuwat, V., Berg, C. M., and Buffinton, W. K., "Extended kalman filtering applied to a two-axis robotic arm with flexible links," *The International Journal of Robotics Research*, vol. 19, pp. 254–270, Mar 2000.

[83] Lew, Y. J. and Book, J. W., "Hybrid control of flexible manipulators with multiple contacts," pp. 242–247, 1992.

[84] Lewis, L. F., Xie, L., and Popa, D., *Optimal and Robust Estimation With an Introduction to Stochastic Control Theory.* 2007.

[85] Lin, J. and Lewis, F. L., "Two-time scale fuzzy logic controller of flexible link robot arm," *Fuzzy sets and systems*, vol. 139, no. 1, pp. 125–149, 2003.

[86] Lin, S.-H., Tosunoglu, S., and Tesar, D., "Control of a six-degree-of-freedom flexible industrial manipulator," *IEEE Control Systems*, vol. 11, pp. 24–30, 1989.

[87] Luca, D. A. and Siciliano, B., "Closed-form dynamic model of planar multilink lightweight robots," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 21, pp. 826–839, 1989.

[88] Magee, P. D. and Book, J. W., "Eliminating multiple modes of vibration in a flexible manipulator," *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pp. 474–479, 1991.

[89] Malzahn, D.-I. J., *Modeling and Control of Multi-Elastic-Link Robots under Gravity*. PhD thesis, TU Dortmund, 2014.

[90] Malzahn, J., Phung, S. A., and Bertram, T., "A multi-link-flexible robot arm catching thrown balls," Proceedings of ROBOTIK 2012; 7th German Conference on Robotics;, pp. 1–6.

[91] 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, pp. 539–544, 2008.

[92] Maybeck, S. P., *Stochastic models, estimation, and control*, vol. 141 of *Mathematics in Science and Engineering*. 1977.

[93] Meggiolaro, A. M. and Dubowsky, S., "Improving the positioning accuracy of powerful manipulators with application in nuclear maintenance," vol. 15 of *Proceedings of the 16th Brazilian Congress of Mechanical Engineering on Robotics and Control*, pp. 210–219.

[94] Mehra, R., "On the identification of variances and adaptive kalman filtering," *IEEE Transactions on automatic control*, vol. 15, pp. 175–184, 1968.

[95] Mehrez, W. M. and El-Badawy, A. A., "Effect of the joint inertia on selection of under-actuated control algorithm for flexible-link manipulators," *Mechanism and Machine Theory*, vol. 45, pp. 967–980, 2008.

[96] Meirovitch, L., *Elements of Vibration Analysis*. McGraw-Hill, Inc., 1985.

[97] Milford, I. R. and Asokanthan, F. S., "Configuration dependent eigenfrequencies for a two-link flexible manipulator: experimental verification," *Journal of Sound and Vibration*, vol. 222, pp. 191–207, 1998.

[98] Mohamed, Z. and Tokhi, O. M., "Command shaping techniques for vibration control of a flexible robot manipulator," *Mechatronics*, vol. 14, pp. 69–90, 2002.

[99] Naets, F., Tamarozzi, T., Heirman, G. H., and Desmet, W., "Real-time flexible multibody simulation with global modal parameterization," *Multibody System Dynamics*, vol. 27, no. 3, pp. 267–284, 2012.

[100] Nagarajan, S. and Turcic, A. D., "Lagrangian formulation of the equations of motion for elastic mechanisms with mutual dependence between rigid body and elastic motions: Part i—element level equations," *Journal of Dynamic Systems, Measurement, and Control*, vol. 112, pp. 203–214, 1989.

[101] Nissing, D., "A vibration damped flexible robot: Identification and parameter optimization," *American Control Conference, 2000. Proceedings of the 2000*, vol. 3, pp. 1715–1719, 1998.

[102] Obergfell, K. and Book, J. W., "End-point position measurement of long-reach flexible manipulators," 1993.

[103] Obergfell, K. and Book, J. W., "Control of flexible manipulators using vision and modal feedback," pp. 71–86, 1998.

[104] Pastor, M., Binda, M., and Harčarik, T., "Modal assurance criterion," *Procedia Engineering*, vol. 48, pp. 543–548, 2012.

[105] 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, 2009.

[106] Pereira, E., Trapero, R. J., D?az, M. I., and Feliu, V., "Adaptive input shaping for manoeuvring flexible structures using an algebraic identification technique," *Automatica*, vol. 45, pp. 1046–1051, 2008.

[107] Perrot, Y., Gargiulo, L., Houry, M., Kammerer, N., Keller, D., Measson, Y., Piolain, G., and Verney, A., "Long-reach articulated robots for inspection and mini-invasive interventions in hazardous environments: Recent robotics research, qualification testing, and tool developments," *Journal of Field Robotics*, vol. 29, pp. 175–185, 2010.

[108] Post, K. B. and Book, J. W., "A robust observation strategy for the control of flexible manipulators," Apr 2011.

[109] Post, K. B., *Robust state estimation for the control of flexible robotic manipulators.* PhD thesis, 2011.

[110] Pradhan, K. S. and Subudhi, B., "Real-time adaptive control of a flexible manipulator using reinforcement learning," *IEEE Transactions on Automation Science and Engineering*, vol. 9, pp. 237–249, 2010.

[111] Rhim, S. and Book, J. W., "Adaptive time-delay command shaping filter for flexible manipulator control," *IEEE/ASME Transactions on Mechatronics*, vol. 9, pp. 619–626, 2002.

[112] Saravanos, A. D. and Lamancusa, S. J., "Optimum structural design of robotic manipulators with fiber reinforced composite materials," *Computers & structures*, vol. 36, pp. 119–132, 1988.

[113] Sasaki, M., Asai, A., Shimizu, T., and Ito, S., "Self-tuning control of a two-link flexible manipulator using neural networks," *ICCAS-SICE, 2009*, pp. 2468–2473.

[114] Sciavicco, L. and Siciliano, B., *Modelling and control of robot manipulators.* Springer Verlag, 1998.

[115] Scott, A. M., Gilbert, G. M., and Demeo, E. M., "Journal of guidance, control, and dynamics," vol. 16, pp. 275–280, 1992.

[116] Shawky, A., Ordys, A., and Grimble, J. M., "End-point control of a flexible-link manipulator using hinfinity nonlinear control via a state-dependent riccati equation," pp. 501–506, 2001.

[117] Siciliano, B., Prasad, R. J. V., and Calise, J. A., "Output feedback two-time scale control of multilink flexible arms," *ASME J. Dyn. Syst., Meas., Control*, vol. 114, pp. 70–77, 1990.

[118] Siciliano, B. and Book, W. J., "A singular perturbation approach to control of lightweight flexible manipulators," *The International Journal of Robotics Research*, vol. 7, no. 4, pp. 79–90, 1988.

[119] Simon, D., *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2004.

[120] Singer, C. N. and Seering, P. W., "Preshaping command inputs to reduce system vibration," *Journal of Dynamic Systems, Measurement, and Control*, vol. 112, pp. 76–82, 1989.

[121] Singer, C. N. and Seering, P. W., "Preshaping command inputs to reduce system vibration," *J. of Dynamic Sys., Measurement, and Control*, vol. 112, pp. 76–82, 1989.

[122] Singhose, W., Singer, N., and Seering, W., "Comparison of command shaping methods for reducing residual vibration," *Proceedings of European control conference*, pp. 1126–1131, 1993.

[123] Singhose, W. and Vaughan, J., "Reducing vibration by digital filtering and input shaping," *IEEE Transactions on Control Systems Technology*, vol. 19, pp. 1410–1420, 2009.

[124] Singhose, W. E., Seering, W. P., and Singer, N. C., "Input shaping for vibration reduction with specified insensitivity to modeling errors," vol. 1, pp. 307–13, 1996.

[125] Smith, M. O. J., "Posicast control of damped oscillatory systems," *Proceedings of the IRE*, vol. 45, pp. 1249–1255, 1956.

[126] Spong, W. M., Hutchinson, S., and Vidyasagar, M., *Robot modeling and control*. John Wiley & Sons New York, 2004.

[127] Staufer, P. and Gattringer, H., "State estimation on flexible robots using accelerometers and angular rate sensors," *Mechatronics*, 2010.

[128] Subudhi, B. and Morris, A. S., "Dynamic modelling, simulation and control of a manipulator with flexible links and joints," *Robotics and Autonomous Systems*, vol. 41, no. 4, pp. 257–270, 2002.

[129] Subudhi, B. and Morris, S. A., "Dynamic modelling, simulation and control of a manipulator with flexible links and joints," *Robotics and Autonomous Systems*, vol. 41, pp. 257–270, 2001.

[130] Subudhi, B. and Morris, S. A., "Soft computing methods applied to the control of a flexible robot manipulator," *Applied Soft Computing*, vol. 9, pp. 149–158, 2007.

[131] Sun, D., Mills, K. J., Shan, J., and Tso, K. S., "A pzt actuator control of a single-link flexible manipulator based on linear velocity feedback and actuator placement," *Mechatronics*, vol. 14, pp. 381–401, 2002.

[132] Sung, Y.-G. and Lee, K.-T., "An adaptive tracking controller for vibration reduction of flexible manipulator," *International Journal of Precision Engineering and Manufacturing*, vol. 7, pp. 51–55, 2004.

[133] Thenozhi, S. and Yu, W., "Advances in modeling and vibration control of building structures," *Annual Reviews in Control*, vol. 37, pp. 346–364, 2012.

[134] Theodore, J. R. and Ghosal, A., "Modeling of flexible-link manipulators with prismatic joints.," *IEEE Transactions on Systems Man and Cybernetics Part B (Cybernetics)*, vol. 27, pp. 296–305, Jan 1997.

[135] Theodore, J. R. and Ghosal, A., "Comparison of the assumed modes and finite element models for flexible multilink manipulators," *The International Journal of Robotics Research*, vol. 14, pp. 91–111, Apr 1995.

[136] Theodore, J. R. and Ghosal, A., "Robust control of multilink flexible manipulators," *Mechanism and machine theory*, vol. 38, pp. 367–377, 2001.

[137] Tomei, P. and Tornambe, A., "Approximate modeling of robots having elastic links," *Systems, Man and Cybernetics, IEEE Transactions on*, pp. 831–840, Aug 1988.

[138] Tosunoglu, S., Lin, S.-H., and Tesar, D., "Accessibility and controllability of flexible robotic manipulators," in *American Control Conference, 1990*, pp. 704–711, IEEE, 1990.

[139] Tumari, M. M. Z., Ahmad, A. M., Saealal, S. M., Zawawi, A. M., Mohamed, Z., and Yusop, M. N., "The direct strain feedback with pid control approach for a flexible manipulator: Experimental results," *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*, pp. 7–12.

[140] Uchiyama, M. and Konno, A., "Modeling, controllability and vibration suppression of 3d flexible robots," in *Robotics Research*, pp. 90–99, Springer, 1996.

[141] Uchiyama, M., Konno, A., Uchiyama, T., and Kanda, S., "Development of a flexible dual-arm manipulator testbed for space robotics," *Intelligent Robots and Systems' 90.'Towards a New Frontier of Applications', Proceedings. IROS'90. IEEE International Workshop on*, pp. 375–381, 1988.

[142] Usoro, B. P., Nadira, R., and Mahil, S. S., "A finite element/lagrange approach to modeling lightweight flexible manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 108, 1984.

[143] Van Der Smagt, P., Groen, F., and Schulten, K., "Analysis and control of a rubbertuator arm," *Biological Cybernetics*, vol. 75, 1994. Biological Cybernetics Source Information: 1996, Vol. 75 Issue 5, p433; Subject Term: ROBOTICS; Subject Term: ELECTRONIC control; Number of Pages: 8p; Document Type: Article.

[144] Vaughan, J., Yano, A., and Singhose, W., "Comparison of robust input shapers," *Journal of Sound and Vibration*, vol. 315, pp. 797–815, 2006.

[145] Wang, Z., Zeng, H., Ho, W. D., and Unbehauen, H., "Multiobjective control of a four-link flexible manipulator: a robust h? approach," *Control Systems Technology, IEEE Transactions on*, vol. 10, pp. 866–875, 2001.

[146] Xie, L., Yeng Chai, S., and de Souza, E. C., "Robust kalman filtering for uncertain discrete-time systems," *Automatic Control, IEEE Transactions on*, vol. 39, pp. 1310–1314, 1992.

[147] Xu, Y. and Ritz, E., "Vision based flexible beam tip point control," *Control Systems Technology, IEEE Transactions on*, vol. 17, pp. 1220–1227, 2007.

[148] Yang, Z. and Sadler, P. J., "Large-displacement finite element analysis of flexible linkages," *Journal of Mechanical Design*, vol. 112, 1988.

[149] Yeganefar, N. and Dambrine, M., "Relation between exponential stability and input-to-state stability of time-delay systems," *American Control Conference, 2007. ACC'07*, pp. 4919–4920, 2005.

[150] Yigit, S. A., "Dynamics of a radially rotating beam with impact: implications for robotics," 1987.

[151] Yin, H., Kobayashi, Y., Hoshino, Y., and Emaru, T., "Hybrid sliding mode control with optimization for flexible manipulator under fast motion," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 458–463, IEEE, 2011.

[152] Yuan, B.-S., Book, J. W., and Huggins, D. J., "Dynamics of flexible manipulator arms: Alternative derivation, verification, and characteristics for control," *Journal of dynamic systems, measurement, and control*, vol. 115, pp. 394–404, 1992.

[153] Zhou, T., Zu, J. W., and Goldenberg, A. A., "Vibration controllability of flexible robot-payload systems," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2, pp. 1484–1489, IEEE, 2000.

[154] Zhu, X., Soh, C. Y., and Xie, L., "Robust kalman filter design," *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, vol. 4, pp. 3813–3818.