NAG9-3

...ING, PROPERTY ... AND ADAPTIVE ...IGN OF SERIAL ...NIPULATORS ...D WITH ... ...TURAL ...ANCE

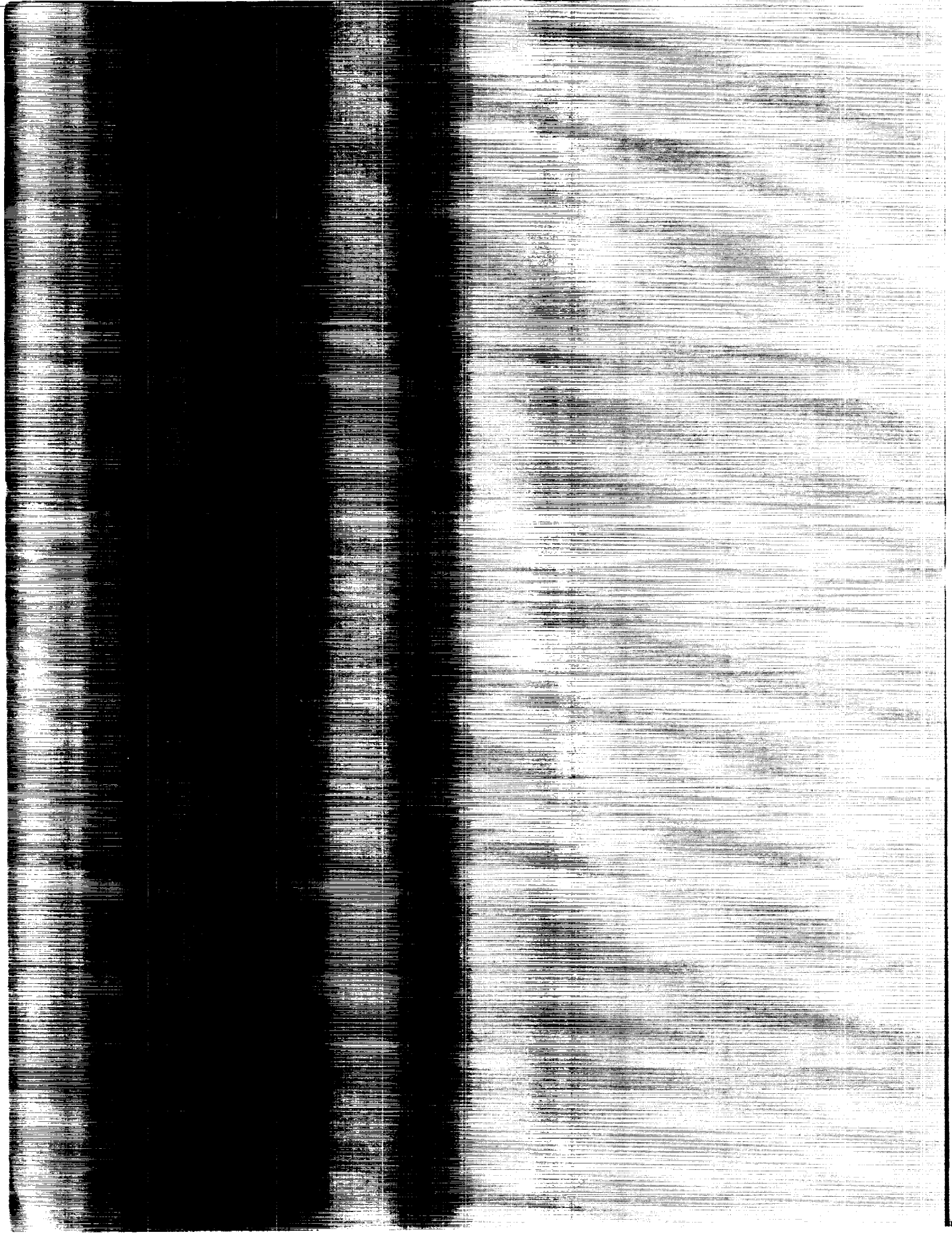
...Tesar
...Curran Chair
...Engineering

...inoglu
...ssociate

...ter Lin
...ssistant

...r 1990

...anical Engineering
...xas at Austin
...as 78712



DYNAMIC MODELING, PROPERTY INVESTIGATION, AND ADAPTIVE CONTROLLER DESIGN OF SERIAL ROBOTIC MANIPULATORS MODELED WITH STRUCTURAL COMPLIANCE

Delbert Tesar
Carol Cockrell Curran Chair
in Engineering

Sabri Tosunoglu
Research Associate

Shyng-Her Lin
Research Assistant

December 1990

Department of Mechanical Engineering
University of Texas at Austin
Austin, Texas 78712

Acknowledgments

This work was partially supported by the U.S. Department of Energy's program on Robotics for Advanced Reactors (Grant No. DE-FG-02-86NE37966). The universities of Florida, Michigan, Tennessee, Texas and Oak Ridge National Laboratory participate to this program.

Also, a grant from Cray Research, Inc. (Grant No. 0032188-00) allowed us to carry out the computer simulations on a Cray X-MP computer which considerably speeded up our progress. The authors acknowledge the funding agencies.

Abstract

This report presents research results on general serial robotic manipulators modeled with structural compliances. Two compliant manipulator modeling approaches, distributed and lumped parameter models, are used in this study. System dynamic equations for both compliant models are derived by using the first and second order influence coefficients. Also, the properties of compliant manipulator system dynamics are investigated. One of the properties, which is defined as inaccessibility of vibratory modes, is shown to display a distinct character associated with compliant manipulators. This property indicates the impact of robot geometry on the control of structural oscillations. Example studies are provided to illustrate the physical interpretation of inaccessibility of vibratory modes.

Two types of controllers are designed to control compliant manipulators. These controllers are designed for compliant manipulators modeled by either lumped or distributed parameter techniques. In order to maintain the generality of the results, neither linearization is introduced, nor any nonlinear term is neglected to simplify the controller design problem. The first type controller is built for N-degree-of-freedom robots with known system parameters, and several distinct control algorithms are introduced. Example simulations are given to demonstrate the controller performance. The second type controller is also built for general serial robot arms and is adaptive in nature which can estimate uncertain payload parameters on-line and simultaneously maintain trajectory tracking properties. The relation between manipulator

motion tracking capability and convergence of parameter estimation properties is discussed through example case studies. The effect of control input update delays on adaptive controller performance is also studied.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The goal of building efficient robots draws growing attention to the study of lightweight robotic manipulators. In order to maintain operational precision, traditional industrial robots are built relatively rigid at the cost of heavy weight and slow operational speed. In addition, the payload capacity is generally limited to as small as 1% of the manipulator weight to avoid deflections caused by inertial loading. Apparently, such robots consume large driving power and are inefficient to operate. As a remedy, the next generation robots tend to be lighter, faster, and have larger payload-to-weight ratios. However, a lightweight structure is subject to deformation and oscillations under high-speed inertial load. Therefore, building a high precision lightweight robot demands a thorough study of the inherent compliance problems from mechanism design to control issues. So far, lightweight manipulators are used mainly in outer-space exploration where inertial load and precision are not of concern; yet, development for high precision industrial applications is still at the infancy stage. This report will present research results on the dynamic modeling and controller design of serial robotic manipulators modeled with structural compliances. In this report, we will refer to this type of robotic arms as *compliant manipulators*.

This work covers several major topics. Literature survey on the study of compliant manipulators will be presented in this chapter. The

methodologies and algorithms used by researchers will be summarized. The second chapter will introduce some basic and handy tools for dynamic modeling of robotic manipulators. The third chapter will derive compliant manipulator dynamics by both distributed and lumped parameter models. Based on the derived system dynamics, the inherent dynamic properties will be investigated in the fourth chapter. The fifth chapter will design different control laws for compliant manipulators with known system parameters. In the sixth chapter, an adaptive algorithm will be introduced, which is capable of on-line motion control and payload estimation. In order to build the adaptive controller, system dynamics of uncertain parameters and an estimation method are also presented in the sixth chapter. The final chapter will summarize the effort of this work. Portions of this work have been reported in [Lin, Tosunoglu, and Tesar, 1990, a], [Lin, Tosunoglu, and Tesar, 1990, b], [Tosunoglu, Lin, and Tesar, 1990, a], [Tosunoglu, Lin, and Tesar, 1990, b], [Lin, Tosunoglu, and Tesar, 1989], and [Tosunoglu, Lin, and Tesar, 1989], which, due to their distinct approach, are not included in the following literature survey.

1.1 Dynamic Modeling Survey: Distributed Parameter Model

The structural flexibility modeling of a compliant manipulator concentrates on two elementary components: links and joints. Lack of structural rigidity in link design causes link compliance. On the other hand, flexibility of power transmission is usually the major contributor of joint compliance. Various models are proposed by researchers to describe link and joint compliances. Nevertheless, they could be categorized into two main disciplines: distributed and lumped parameter models. Both models have been applied

to model link compliance. However, joint compliance is always modeled by a lumped parameter model. In distributed parameter models, a link is modeled as a continuous beam which has infinite degree-of-freedom (DOF) oscillations. Generally, finite assumed modes are chosen to discretize the oscillations. The chosen mode shapes are mainly admissible functions satisfying given geometric boundary conditions. The assumed mode method not only reduces system dimension, but also separates the spatial and time variables of each vibratory mode, consequently, the system dynamics could be expressed as a function of generalized coordinates composed of nominal joint parameters and vibratory amplitudes. Some of the activities in distributed parameter model are reported in this section. [Hughes, 1979] models each compliant link as a continuous system and derives general compliant manipulator dynamics by the Newton-Euler approach. He computes the inertial dynamic force at each compliant link first, and then finds the compliant dynamics from flexibility kernel. In this work, the velocity coupling terms are neglected under slow motion assumption. [Sunada and Dubowsky, 1981] use a finite element model and the NASTRAN software package to study compliant mechanism motion. [Book, 1984] uses the Bernoulli-Euler Theory to model a compliant link under lateral bendings and longitudinal elongation. The rotatory inertial effect is neglected in the system dynamics. In this work, the author uses the same modal amplitude in all directional modal functions. The 4×4 homogeneous transformation matrix is applied to perform kinematic analysis. [Low, 1987] also uses Bernoulli-Euler Theory to build compliant system dynamics. In his recent work, [Low, 1989] presents solution schemes for inverse dynamics and kinematics. The author also discusses the assumed mode solution under different boundary conditions. [Naganathan and Soni, 1987] analyze lateral bendings and longitudinal elongation of a compliant link by the finite

element approach. Following the Timoshenko Beam Theory, the rotatory inertia and shear effects are included. The system dynamics are derived by Galerkin's method. [Kane, Ryan, and Banerjee, 1987] model a spatial manipulator with a compliant last link. Instead of using the Bernoulli-Euler Theory, the shear and centrifugal stiffening effects are included in the study. However, by applying the small deflection assumption, all vibratory coupling terms are neglected by the authors. In the above examples, multi-directional oscillations of the compliant link are considered. Interestingly, a great portion of modeling and control works of compliant manipulators focus on planar one- or two-link arms, and in these works only unilateral bending is investigated. For example, [Rakhsha, and Goldenberg, 1985] model a one-link arm in transverse bending. The compliant link is treated as a cantilever beam, and Newton-Euler approach is applied to derive system dynamics. [Usoro, Nadira, and Mahil, 1986] use the finite element approach to model a planar arm lateral bending. The Hermitian polynomials are chosen as mode shapes. The results are applied on a two-link planar arm example. [Nicosia, Tomei, and Tornambe, 1986] employ monomial mode shapes up to fourth orders to model a planar arm with lateral bending. [Benati and Morro, 1988] model a planar link as clamped-free beam with end-point mass whose lateral bending is depicted by two eigenmodes. Since two modes are suggested, the end-point deflection and tangent angle are used as the generalized coordinates. The model is simulated on a two-link planar arm with a compliant link followed by a rigid link. Instead of a one-link arm, [Yuh, Young, and Baek, 1989] model a cylindric planar arm composed of one pivot joint and one slider. The sliding link is compliant whose unilateral bending is modeled as a cantilever beam. Due to the sliding motion, the compliant link length changes continuously, therefore, the mode shapes are normalized for a unit link length, and

only the first two primary modes are modeled. To verify their analytic model, laboratory experiments are conducted by the authors, in which an accelerometer is added to the tip of the arm, whose double integration provide the tip displacement. The experimental results show good agreement with the analytic model. [Wang and Vidyasagar, 1988] study the dynamic model of a five DOF mechanism which is composed of a 4-bar linkage with a rotating base. The output link is flexible and bends in the direction perpendicular to the plane of the 4-bar. The link compliance is defined following the Bernoulli-Euler Theory. [Yang and Donath, 1988] model one-link arm dynamics by considering both joint and link compliances. The link deflection is described by the Bernoulli-Euler Theory; the modal eigenvalues and eigenfunctions are derived based on the given geometric and natural boundary conditions. The authors suggest that the first two modes give fair representation of link deflection. Their simulation results show that the first mode amplitude is about ten times the second mode magnitude. Since both clamped-free and pinned-free boundary conditions have been used in link compliance study, [Bellezza, Lanari, and Ulivi, 1990] derive the exact solutions of one-link arm vibratory dynamics by using both boundary descriptions. After comparing the solutions, the authors conclude that both results are equivalent after coordinate transformation. Another research topic in distributed parameter model is how many assumed modes should be used so that system dynamics have an acceptable level of accuracy and manageable size. Some works in this discipline are presented here. [Hastings, Dorsey, and Book, 1989] use balanced realization to identify the model order required for a compliant system. They linearize general compliant system dynamics, and by the assumption that the linearized system is controllable and observable, the number of dominant modes are identified from the solution of the linearized dynamic equations

by the singular values decomposition technique. [Tsujisawa and Book, 1989] apply the reduced order method to decide the dominant modes of a specific robot called RALF (Robotic Arm, Large and Flexible) which is a two DOF parallel mechanism with two ten-foot long links. The report suggests that two modes for each link is optimal from the control point of view. In their experimental study, it is observed that the spectrum ratio of the first mode to the second mode is ten to one. Also, [Krishnan and Vidyasagar, 1988] derive a reduced order model for a single-link arm by Hankel norm minimization. Finally, for on-line motion control, the vibratory states need to be identified. Regarding this subject, [Hastings and Book, 1986] use strain-gauge measurements and a reduced order observer to reconstruct the modal amplitudes and velocities, and [Hastings and Ravishankar, 1989] suggest several link deflection measurement methods and discuss the effect of the measurement technique on model order estimation. According to the works surveyed in this report, various deflection measurement equipments have been used, which include strain gauges, accelerometers, laser-interferometers, photodetectors, piezoelectric detectors, and vision systems.

1.2 Dynamic Modeling Survey: Lumped Parameter Model

Ideally link oscillations are composed of infinite modes, yet, the first fundamental mode generally dominates most of the elastic energy as reported by [Tsujisawa and Book, 1989] and [Yang and Donath, 1988]. Additionally, relatively high energy is required to bend a robotic link into high order mode shapes. Also, structural damping makes higher modes difficult to detect; therefore, a lumped parameter model is often used on compliant links to create an efficient dynamic description suitable for real-time motion control.

Besides modeling links, a lumped parameter model is frequently used to define joint compliance. In a lumped parameter model, each structural compliance is replaced by an equivalent spring. The two types of springs mostly used are translational and torsional springs. The former causes linear deflection and the later creates torsional deformation. The stiffness of the modeled spring is evaluated by either elemental stress-strain relations or by experimental identification. Some representative works are listed below.

[Tesar, 1978] uses lumped parameters to model an N DOF planar arm with compliant links. The loads at the distal end of each link are analyzed first, then the subjected deformations are derived from a static cantilever beam deflection relation. [Fresonke, Hernandez, and Tesar, 1988] further extend this approach to cover the deformations of a spatial mechanism with seven possible deflections at each link, i.e., one joint deformation and six distal end deflections and twistings. Since static load and deformation relations are used, this approach is termed quasi-static deflection analysis. Since quasi-static deflection is considered to be the major structural disturbance to end-point precession, it should be compensated for during motion control. Therefore, [Hernandez, 1989] develops real-time computation software to evaluate the end-effector deflection under inertial and external loads. In the lumped parameter model, the structural stiffnesses could be evaluated from elemental stress-strain relation. However, for an assembled robot, the elemental stiffness matrices are difficult to obtain analytically. Therefore, metrology approaches are applied to obtain the modeled spring stiffness values. [Behi, 1985] experimentally identifies the lumped parameters of a T3-776 robotic manipulator. Modal analysis is employed to identify the natural frequencies and modal amplitudes of the excited manipulator. The experimental data are fed into the lumped model to obtain system parameter values.

Instead of using frequency domain analysis, [Sklar, 1988] applies end-point loads on a static T3-776. A force sensor attached to the wrist reads the force and torque components along each orthogonal direction. The resultant hand deflections are measured by twin theodolite systems. From both force and deflection data, the global compliance matrix could be obtained, which in turn produces the modeled stiffness values after inverse kinematic operation. In order to obtain averaged stiffness values, the tests are repeated at different robot configurations and external loads. Similar metrology activities are reported by [Good, Sweet, and Strobel, 1984] and [Elmaraghy and Johns, 1988]. The lumped parameter model is also applied to the study of the Spatial Shuttle Remote Manipulator System (RMS). [Book, 1979] models the RMS as a massless chain connecting two end-point masses, i.e., the Orbiter and payload. The global hand compliance matrix is derived by using 4×4 homogeneous transformation matrices. Similarly, [Sellhorst, 1982] describes the compliances of the RMS as three linear torsional springs and studies payload motion response subjected to the thrust fired on Orbiter. More studies on lumped parameter model could be found. For example, [Huston, 1980] models multibody dynamics by assuming that each pair of bodies is connected by three translational and three torsional springs. The system dynamics are derived by Kane's partial velocity and partial angular velocity approaches along with the principle of virtual work. The result is applied in [Kelly and Huston, 1981] to derive manipulator dynamics for a six-link arm. In this work, the stiffness values are derived from elemental stiffness relations. The elemental stiffness matrix is also used by [Shahinpoor and Meghdari, 1988] to derive the global hand stiffness matrix. Instead of using one set of lumped parameters for each compliant link, [Huang and Lee, 1988] suggest to divide a compliant link into several lumped parameter segments to obtain a more

accurate model.

1.3 Survey on Compliant Manipulator Control

Besides dynamic modeling, a great portion of research on compliant manipulators is carried out on motion control. However, unlike modeling techniques, various controllers have been designed and tested numerically or experimentally. According to the control algorithm used, the following surveyed reports are classified into optimal control, singular perturbation method, external feedback linearization, inverse dynamics, quasi-static deflection compensation, resonance avoidance control, linearized system dynamics control, and adaptive control techniques.

1.3.1 Optimal Control

One special character of one-link flexible arms is that system dynamics contain no coupling terms. Therefore, after modal decomposition, the system dynamics are reduced to a linear time-invariant system. From that process, many researchers have discovered an interesting property that for one-link arms the transfer function between tip output and actuator input has non-minimum phase zeros. This means that for a given tip trajectory, the inverse dynamics could not be defined directly because the inverse transfer function is unstable. Hence, optimal control is used by some researchers to control the tip motion of one-link arms. One of the most famous studies in this area is reported by [Cannon and Schmitz, 1984] who model a one-link arm as a pinned-free beam. The Bernoulli-Euler Theory is used to derive system flexibility dynamics. Then the dynamic equations are decoupled by orthogonal modes. The decoupled system parameters are identified experimentally.

Finally, the tip motion is controlled by an optimal algorithm. Similar works on one-link arm are reported by [Sakawa, Matsuno, and Fukushima, 1985], [Chassiakos and Bekey, 1986], [Lee and Castelazo, 1987], [Pal, Stephanou, and Cook, 1988], and [Biswas and Klafter, 1988]. Optimal control is also applied on a multi-DOF planar mechanism containing a compliant beam as the last link, such as reported by [Matsuno, Fukushima, Kiyohara, and Sakawa, 1987] and [Schmitz, 1989]. However, in both works, flexibility dynamics are linearized around terminal static states.

1.3.2 Singular Perturbation Method

Compliant manipulator dynamics are composed of two parts: one basically describes the dynamic balance of driving joints, called the rigid part dynamics, and the other depicts the dynamic interactions due to structural flexibility, or the vibratory part dynamics. Each part contains terms of nominal and vibratory parameters. In the singular perturbation method, it is assumed that the solutions of the vibratory part dynamics, a set of second order differential equations, are stable and called *integral manifolds*. Because of the nonlinear couplings and kinematic dependence, it is difficult to solve integral manifolds explicitly. But it is possible to find their approximations by expanding the vibratory dynamics around the rigid body mode by using the Taylor series method. If the structural compliances are small, this approximation will produce the major part of the integral manifolds in terms of rigid body parameters. A back substitution of these vibratory mode solutions into rigid part dynamics converts all vibratory parameters into rigid body dynamic parameters. At this point, any control algorithm developed for rigid manipulators could be applied on the rigid part dynamics. In this process, the

integral manifolds are assumed stable, therefore, they are not considered in the rigid part control design. However, the vibratory modes do not necessarily slide on the integral manifolds, hence, a small perturbed control input is applied to force the vibratory modes to track the integral manifolds. In doing so, it is assumed that the perturbed inputs are so small that it will not affect the rigid part's dynamic response. The drawback of singular perturbation is that in approximating the integral manifolds high order dynamic parameters are required, for example, fourth order dynamic parameters are used to produce the first order approximation. For a multi-DOF manipulator, it is highly demanding to derive such high order dynamic parameters. Therefore, this method is used mainly on simple systems such as one-link arms modeled with link or joint compliance. Some examples of this specific subject can be found in [Khorasani and Spong, 1985], [Marino and Spong, 1986] [Spong, Khorasani, and Kokotovic, 1987], and [Slotine and Hong, 1986] for one-link arm with a compliant joint; also [De Maria and Siciliano, 1987], [Siciliano and Book, 1988], and [Khorrami and Ozguner, 1988] for one-link arms with link compliance. In the last three reports, the authors model link compliances by using the Bernoulli-Euler Theory with clamped-free boundary conditions.

1.3.3 External Feedback Linearization

According to [Su, 1982], a nonlinear system could be converted into an equivalent linear system provided that system dynamics satisfy certain given conditions. The transformation process is called diffeomorphic coordinate transformation, and nonlinear system *rank* and *involutivity* conditions need to be checked over a special set of vectors derived by Lie brackets to ensure existence of a nonlinear transfer function. Provided that the transfor-

mation exits, then the manipulator dynamics could be expressed in term of a linear, time-invariant, controllable system whose controller is easy to design. Once a control law is developed for the equivalent linear system, the nonlinear system controller is obtained by an inverse transformation. Although, the conditions could be checked by tedious but straightforward differentiation, there is no simple rule guiding the selection of nonlinear transformation function, or diffeomorphism. The nonlinear compensation, or computed torque method, of rigid manipulator control is a special application of this technique. However, for compliant manipulators, this approach is still at a conceptual stage and is used only on one-link arms with link or joint compliance, such as reported by [Spong, 1987] on a one-link arm with compliant joint, [Nicosia, Tomei, and Tornambe, 1989] on one-line arm with flexible link, and [De Luca, Isidori, and Nicolo, 1985] on conceptual design.

1.3.4 Inverse Dynamics

Inverse dynamics basically is an open loop control scheme. For a given hand trajectory, the required driving torques are computed along the trajectory to ensure precise tracking. For rigid manipulators, inverse dynamics is mainly solving the inverse kinematics problem and is accomplished in one iteration. However, for compliant manipulators, structural compliance causes disturbances to hand motion, and due to the nonlinear interaction between nominal and vibratory modes, more than one iteration is generally required to find the final driving torques. For example, in the first iteration, rigid manipulator assumption is used to find approximated driving torques. The torques are applied on the compliant manipulator to solve the associated structural deformation. Then the resultant hand deflections are compensated

by adjusting nominal joint motion to maintain tracking precision. The driving torques of compliant manipulators are evaluated from compensation results. The process is repeated until an acceptable level of precision is reached. This technique is applied by [Asada, Ma, and Tokumaru, 1987] on a two-link planar arm. A similar approach is found in [Dado and Soni, 1986] for multi-link arms modeled by the finite element method. Instead of time domain approach, [Bayo and Moulin, 1989] find the transfer function between the tip and actuator input of a one-link flexible arm in the Laplace domain. Once the transfer function is defined, the actuator input is computed by convolution integral for a given tip trace. [De Luca, Lucibello, and Ulivi, 1989] use inverse dynamics to track various output points other than the end-effector tip. In their algorithm, the number of output states must equal to that of nominal joints. The tracking stability is also analyzed in this report.

1.3.5 Quasi-Static Deflection Compensation

Quasi-static deflection is considered to be the major contributor of structural deformation. Some reports suggest compensation of quasi-static deflection by either off-line trajectory planning or *control-in-the-small* algorithms. [Gupta, 1987] studies stationary compliant manipulator deflections under external forces. The hand deflections are compensated by adjusting the robot configuration. [Pfeiffer, 1989] divides the problem of compliant robot control into three stages. The first stage plans off-line optimal trajectory for rigid manipulators and computes the associated driving torques. The second stage finds the associated structural quasi-static deflection and adjusts joint variables to compensate the deflection. The last stage linearizes system dynamics around the terminal point of task trace and builds an on-line controller

for the linearized time-invariant system. This control algorithm is tested on a two-link planar arm modeled with link and joint compliances. In control-in-the-small technique, a mechanism performing fine-tuned motion is added to compliant manipulator design to compensate structural deflection. [Zalucky and Hardt, 1984] design a link composed of two parallel beams. The distal ends of the beams are jointed by a hydraulic servo. The rigidity of the link could be increased by regulating the hydraulic servo. [Oliver, Wysocki, and Thompson, 1985] replace one grounded pivot joint of a four-bar mechanism by a fast moving slider to actively compensate the output link deflection. [Tlusty and Wegerif, 1986] use a cam mechanism to compensate the deflection of a T3-776 robot during cutting process. The robot is modeled by a lumped parameter model. The global hand stiffness matrix is derived first, then hand deflections are computed form the stiffness matrix together with cutting forces measured at the wrist. The added cam then compensates the calculated deflection by providing the cutter with a fast but small sliding motion. In these reports, the fine-tuned mechanisms are limited to one degree of freedom. A micromanipulator capable of six-DOF fine-tuned motion is designed and analyzed by [Hudgin and Tesar, 1988] and [Han, Traver, and Tesar, 1989].

1.3.6 Resonance Avoidance Control

A group of activities on compliant manipulators are devoted to structural resonance study. [Cleghorn, Tabarrok, and Fenton, 1984] analyze the influence of running speed on the stability of a compliant four-bar mechanism. They solve the eigenvalues of system dynamics and find the associated running speeds. Through eigenvalue locations, the stable and unstable op-

eration speeds could be identified. In order to study the induced geometric stiffness effects on system stability, [Anderson, 1985] linearizes system dynamics around an equilibrium point and studies the effect of external forces on system stability. The manipulator used in this study is a three-link planar arm modeled with resilient joints. [Rivin, Zeid, and Rastgu-Ghamsari, 1985] model the compliances of a two-prismatic-joint planar arm by the lumped parameter approach. Two translational and one torsional springs are added to each joint of the arm. By neglecting vibratory velocity terms, the stable eigenvalues of the system dynamic equations are analyzed to examine the effect of nominal joint velocity on system stability. [Streit, Krousgrill, and Bajaj, 1986] linearize compliant manipulator dynamics and analyze system stability under repetitive operation from the eigenvalues of the linearized system. They apply this technique to a two-link planar arm modeled with lumped parameters. [Chiou and Shahinpoor, 1990] study the force control effect on system stability. They model two-link planar arm flexibility by the Bernoulli-Euler Theory. Hybrid force/position control is also adopted to form a closed loop control action. By slow motion assumption, the nonlinear terms are neglected, then the closed-loop system dynamics are linearized around an equilibrium position. To investigate system stability, the eigenvalues of the final linear dynamic equations are analyzed over various force feedback gains, force sensor stiffness, and structural flexibility. Other than stability study, [Singer and Seering, 1989] suggest using a counteractive oscillation to cancel existing structural vibration. They also suggest that for bang-bang type control the rectangular input commands could be preshaped to remove sensitive frequency contents. However, the authors limit their study to a simple mass-spring-damper model.

1.3.7 Linearized System Dynamics Control

Due to the nonlinear coupling between nominal and vibratory modes, building a controller for compliant manipulators is a difficult and challenging task. However, it is suggested that compliant manipulator control problem could be simplified by linearizing compliant system dynamics. [Chalhoub and Ulsoy, 1987] develop a controller for a spherical coordinate robot with two revolute joints and one prismatic joint. The prismatic joint is flexible and both principal lateral bendings are depicted by the first modes. The controller is built on linearized system dynamics, and pole placement technique is applied to locate the feedback gains. This algorithm is experimentally tested on a one-link planar arm. One special feature of this work is that observation and control spillover effects are examined. [Oosting and Dickerson, 1988] study the motion response of a two-link planar arm. Lumped parameters are employed to model both link and joint compliances. The system dynamics are linearized about the desired motion, and a control law for the linear system is designed with feedback and feedforward components. [Henrichfreise, 1988] studies the control of a manipulator modeled with three resilient joints and two compliant links. The system dynamics are linearized around an operating point. The controller feedforward and feedback gains of the linear system are selected from a special criterion reported in this work. [Nathan and Singh, 1989] divide the control of robotic arm with compliant links into two phases. In the first phase, system vibratory motion is neglected, and nominal joints are controlled by variable structure control law. The second phase starts when robot reaches the vicinity of terminal point. The system dynamics are then linearized around the terminal point. A vibration stabilizing controller is designed for the linear time-invariant system by pole placement technique.

At this stage both nominal and vibratory controllers are executed, and it is assumed that the vibratory controller is relatively small and does not affect nominal motion control.

1.3.8 Adaptive Control

The adaptive control techniques applied on compliant manipulators include self-tuning regulator, model reference, and gain scheduling methods. In self-tuning approach, [Yang and Gibson, 1989] adaptively control a two-link planar arm with a flexible beam as the second link. The authors assume that system dynamics could be described by an autoregressive moving-average (ARMA) model. Then, the coefficients of the ARMA model are estimated on-line. The control algorithm of the ARMA model is one-step ahead optimization. Similarly, [Yurkovich, Tzes, Lee, and Hillsley, 1990] define a two-link planar arm modeled with link compliances by an ARMA model, and both pole placement and one-step ahead optimization are employed to design control inputs. [Yuh and Tissue, 1990] control a two-link planar arm with joint compliances but rigid links. The continuous system dynamics are linearized around a given trajectory and then converted into discrete time form by Euler's method. The system parameters are assumed unknown and estimated on-line. The control inputs are defined by pole placement technique. [Cetinkunt and Wu, 1990] use a Lattice filter to estimate the coefficients of an autoregressive model which represents system dynamics of a one-link flexible arm. Two controllers are proposed: fixed-pole PD controller and one-step ahead optimization. Similar self-tuning control could be found in [Koivo and Lee, 1989] and [Chen and Menq, 1990]. A common assumption used in self-tuning regulator design is that the estimated system parameters vary slowly

in comparison with state variables. Also, since in the ARMA model nonlinear dynamic parameters are replaced by simple lumped coefficients, information such as payload mass, center of mass location, and moment of inertia are difficult to extract from on-line estimate data. Regarding other adaptive control applications, [Yuan, Book, and Siciliano, 1989] use model reference method to control a one-link flexible arm to behave like a decoupled, stable, linear, time-invariant system. In gain scheduling control, [Nelson and Mitra, 1986] compute off-line the optimal feedback gains of a single-link arm at different payload magnitudes. During on-line control, the uncertain payload is estimated by the gradient method, then the optimal feedback gains are adjusted accordingly with the estimated payload value. In this work, link compliance is modeled by the lumped parameter method. Similar work could be found in [Menq and Chen, 1988] and [Yurkovich, Pacheco, and Tzes, 1989]. Both works focus on control of single-link arms modeled as distributed systems. However, the former uses a gradient method to perform on-line payload estimation, while the latter identifies the payload by an accelerometer attached at the tip of the link. A detailed survey of adaptive control of rigid robotic manipulator is reported by [Tosunoglu and Tesar, 1988].

1.4 Other Approaches

Besides the efforts reported above in modeling and control areas, progress is also reported on strengthening a lightweight manipulator by structural design. For example, [Rivin, et al., 1987] use combinational links to increase structural rigidity and reduce inertial weight. In their design, a link is composed of two segments made of steel and aluminum separately. As suggested by the authors, the optimal design shows a reduction in link deflection

and driving torque. Instead of using traditional metal material, [Liao, Sung, and Thompson, 1987] design manipulator links by using composite materials. Link stiffness and damping are optimized by selecting proper design parameters for the fiber material, the type of matrix, fiber orientation in each ply of laminate, the fiber volume fraction, and the stacking sequence of the plies. A comparison between aluminum and composite laminate arm is conducted by [Choi, Thompson, and Gandhi, 1990]. The authors compare the performance of both arms through the motion control of single-link arms. According to the experimental results, the composite arm shows less settling time and smaller overshoot and consumes less torque.

Despite the extensive effort reported above to solve compliant manipulator problem, most of the approaches are still in early development stages and a great portion of them are limited to simple robotic structures such as one-link arms. For example, development of a simple, general robot controller suitable for both distributed and lumped parameter models have not been pursued. Additionally, the impact of robot geometry on vibratory mode control has seldom been investigated. Also, the majority of adaptive control of compliant manipulators focuses on one- and two-link planar arms. A more meaningful problem, adaptive control of spatial compliant manipulators, needs to be fully explored. In this report, we will develop controllers applicable to both lumped and distributed models. The effects of kinematics on vibratory mode control will be analyzed for a general compliant manipulator. Finally, an adaptive control law capable of on-line payload estimation and motion control will be designed. Case studies will be used to examine and illustrate our design and analysis throughout this report.

Chapter 2

Dynamic Equations of Rigid Robotic Manipulators

Robotic system dynamic equations will be derived in this chapter. In the following derivations, structural compliance is neglected to facilitate the introduction of several handy tools that are essential to this report. The first section will present coordinate transformation matrix that transforms between local and global coordinate frames. Then two properties, the first and second order influence coefficients, are defined in the second and third sections. These two properties have compact and transparent nature that makes system dynamics easy to derive and verify. Finally, robotic system dynamics are derived by using the Lagrange method and further verified by the Newton-Euler method.

2.1 Coordinate Transformation Matrices

One of the frequently used tools in robotic research is coordinate transformation. Generally, a local frame is assigned to each link of a given robot. This local frame is useful in defining physical properties such as the junction point to the next link, the center of mass location, moment of inertia, and so forth. But to find the gross motion of a robot and the required driving force or torque, those physical quantities eventually need to be expressed in a common frame. Coordinate transformation matrix is a tool developed for this purpose. Figure 2.1 shows a floating link and its local coordinate frame

$\{\hat{x}, \hat{y}, \hat{z}\}$, where $\hat{x}$, $\hat{y}$, and $\hat{z}$ are three unit $\mathcal{R}^3$ vectors defined in the global system $\{X, Y, Z\}$. A point p on the link is expressed by three local coordinates

$$r^l = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \in \mathcal{R}^3$$

where the superscript l indicates that the vector is defined in the local frame. The same vector in the global frame is defined by

$$\begin{aligned} r &= r_x\hat{x} + r_y\hat{y} + r_z\hat{z} \\ &= [\hat{x}\ \hat{y}\ \hat{z}] \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \\ &\stackrel{\text{def}}{=} T_l r^l \end{aligned} \tag{2.1}$$

In the above expression, $T_l \in \mathcal{R}^{3\times3}$ is the transformation matrix converting local coordinates to global values. Since the column vectors of T_l are unit and orthogonal, T_l is a orthogonal matrix, that is

$$\begin{aligned} T_l^T T_l &= \begin{bmatrix} \hat{x}^T\hat{x} & \hat{x}^T\hat{y} & \hat{x}^T\hat{z} \\ \hat{y}^T\hat{x} & \hat{y}^T\hat{y} & \hat{y}^T\hat{z} \\ \hat{z}^T\hat{x} & \hat{z}^T\hat{y} & \hat{z}^T\hat{z} \end{bmatrix} \\ &= \mathcal{I} \end{aligned} \tag{2.2}$$

where $\mathcal{I}$ is a 3×3 identity matrix, therefore $T_l^T = T_l^{-1}$. Apparently, to construct a coordinate transformation matrix, the unit vectors of a given local frame need to be identified first. Three basic rotational matrices are often used in defining the local unit vectors. They are derived as follows. In Figure 2.2, $\hat{S}_1$, $\hat{S}_2$, and $\hat{S}_3$ represent three orthogonal unit vectors, where $\hat{S}_1$ and $\hat{S}_2$ rotate an angle θ about $\hat{S}_3$ to the new orientations $\hat{S}_1'$ and $\hat{S}_2'$. By simple inner product operation, the new frame $\{\hat{S}_1', \hat{S}_2', \hat{S}_3'\}$ could be expressed in terms of the original frame $\{\hat{S}_1, \hat{S}_2, \hat{S}_3\}$ as

$$\hat{S}_1' = \cos\theta\hat{S}_1 + \sin\theta\hat{S}_2$$

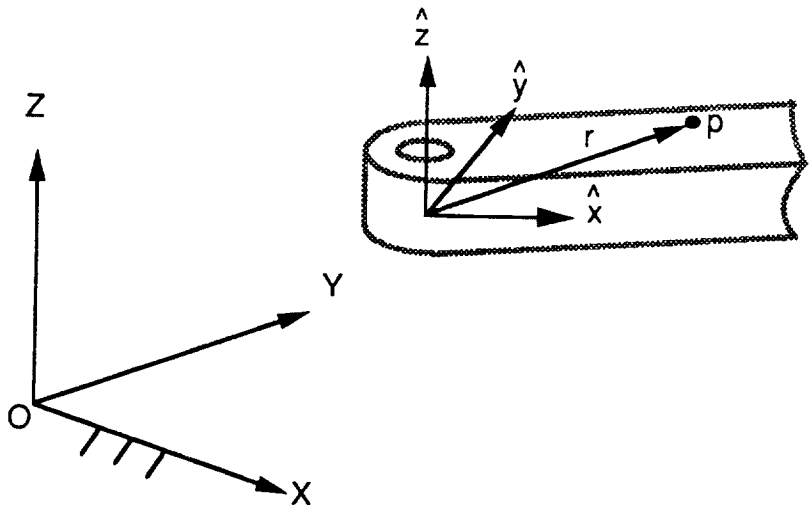


Figure 2.1: Local and Global Coordinates of a Robotic Link

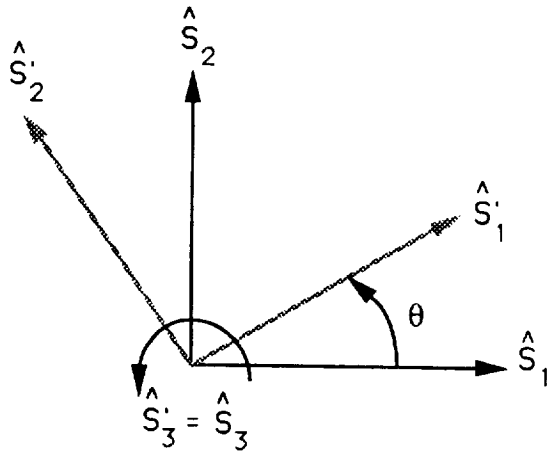


Figure 2.2: Coordinate Transformation Between Two Orthogonal Frames

$$\begin{aligned} \hat{S}_2' &= -\sin\theta\hat{S}_1 + \cos\theta\hat{S}_2 \\ \hat{S}_3' &= \hat{S}_3 \end{aligned} \tag{2.3}$$

Let $\hat{S}_3$ represent one of $\{X, Y, Z\}$, and $\hat{S}_1$ and $\hat{S}_2$ represent the other two by following the right-hand rule relationship that $\hat{S}_1 \times \hat{S}_2 = \hat{S}_3$, then the three rotational matrices, $\mathrm{Rot}(\hat{S}_3, \theta)$, converting $\{\hat{S}_1', \hat{S}_2', \hat{S}_3'\}$ back to $\{\hat{S}_1, \hat{S}_2, \hat{S}_3\}$ are

$$\mathrm{Rot}(X,\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} ; \mathrm{Rot}(Y,\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$\mathrm{Rot}(Z,\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

where the unit vector definitions $X = [1\ 0\ 0]^T$, $Y = [0\ 1\ 0]$, and $Z = [0\ 0\ 1]^T$ are used in the above derivations. In the above equation, $\mathrm{Rot}(\hat{S}_3, \theta)$ denotes that the new frame comes from a rotation of θ angles about an old frame axis $\hat{S}_3$ where $\hat{S}_3 \in \{X, Y, Z\}$. Notice that in each rotational matrix, the column vectors are exactly the unit vectors of the new frame defined in the last frame, so they are transformation matrix. The application of these rotational matrices is illustrated in the following example. The original frame $\{X, Y, Z\}$ rotates an angle α about X to a new frame $\{X', Y', Z'\}$. Then the new frame rotates an angle θ about Z' to the third frame $\{X'', Y'', Z''\}$. To find a matrix T transforming the third frame back to the original frame, the rotational matrices are multiplied together as

$$T = \mathrm{Rot}(X,\alpha)\mathrm{Rot}(Z',\theta) \tag{2.5}$$

Notice that the order of matrix multiplication must be obeyed in computing T. Another example is to find the transformation matrices of the three-link arm shown in Figure 2.3. In this model, each link is labeled with a local

frame $\{X_i, Y_i, Z_i\}$, $i \in \{1,2,3\}$, where the first joint is revolute and rotating about Z_1, the second joint is prismatic and sliding along Z_2, and the third joint has Z_3 as pivot axis. Let $\{X_o, Y_o, Z_o\}$ denote the global frame, and T_i be the transformation matrix converting the ith frame to the global frame. Then

$$\begin{aligned} T_1 &= \mathrm{Rot}(Z_o, \theta_1) \\ T_2 &= T_1 \mathrm{Rot}(Y_1, 90) \\ T_3 &= T_2 \mathrm{Rot}(X_2, 90)\mathrm{Rot}(Z_3, \theta_3) \end{aligned} \tag{2.6}$$

Let $\{\hat{x}_i, \hat{y}_i, \hat{z}_i\}$ denote the unit $\mathcal{R}^3$ vectors of $\{X_i, Y_i, Z_i\}$, $i \in \{1,2,3\}$, then the above transformation matrices give the unit vectors in terms of global coordinates by the relationship

$$T_i = [\hat{x}_i\ \hat{y}_i\ \hat{z}_i] \tag{2.7}$$

2.2 The First Order Influence Coefficients

In this section, two compact notations will be introduced to define the translational and angular velocities of a moving robot. In Figure 2.3, let $P \in \mathcal{R}^3$ be the global positional vector of a given point p within the payload, and also let $\Phi \in \mathcal{R}^3$ be the Euler angles of the payload. Then the velocities of point p are

$$\begin{aligned} \dot{P} &= \left[\frac{\partial P}{\partial \theta_1}\ \frac{\partial P}{\partial \theta_2}\ \frac{\partial P}{\partial \theta_3}\right] \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \\ &\stackrel{\text{def}}{=} {}_T G_p \dot{\theta} \end{aligned} \tag{2.8}$$

and

$$\begin{aligned} \dot{\Phi} &= \left[\frac{\partial \Phi}{\partial \theta_1}\ \frac{\partial \Phi}{\partial \theta_2}\ \frac{\partial \Phi}{\partial \theta_3}\right]\dot{\theta} \\ &\stackrel{\text{def}}{=} {}_R G_p \dot{\theta} \end{aligned} \tag{2.9}$$

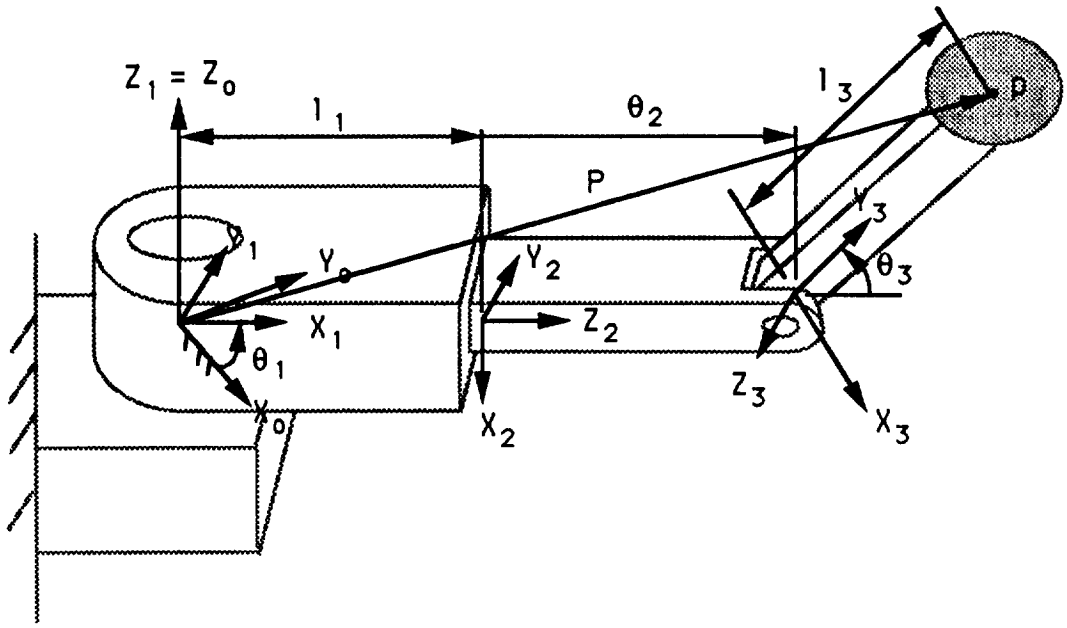


Figure 2.3: 3-Link Rigid Robot Modeled with Local Coordinate Frames

where $\theta = [\theta_1\ \theta_2\ \theta_3]^T \in \mathcal{R}^3$, and ${}_TG_p \in \mathcal{R}^{3\times3}$ is defined as the first order translational influence coefficient, or translational G function, of p. Similarly, ${}_RG_p \in \mathcal{R}^{3\times3}$ is the first order rotational influence coefficient, or rotational G function, of p. The details of ${}_TG_p$ are given next. According to Figure 2.3,

$$\begin{aligned}
\dot{P} &= \frac{d}{dt}(l_1\hat{x}_1 + \theta_2\hat{z}_2 + l_3\hat{y}_3) \\
&= l_1\dot{\hat{x}}_1 + \dot{\theta}_2\hat{z}_2 + \theta_2\dot{\hat{z}}_2 + l_3\dot{\hat{y}}_3 \\
&= l_1(\dot{\theta}_1\hat{z}_1 \times \hat{x}_1) + \dot{\theta}_2\hat{z}_2 + \theta_2(\dot{\theta}_1\hat{z}_1 \times \hat{z}_2) + l_3(\dot{\theta}_1\hat{z}_1 \times \hat{y}_3 + \dot{\theta}_3\hat{z}_3 \times \hat{y}_3) \\
&= [(\hat{z}_1 \times P)\ \hat{z}_2\ (\hat{z}_3 \times l_3\hat{y}_3)]\dot{\theta} \\
&= {}_TG_p\dot{\theta} \qquad (2.10)
\end{aligned}$$

A comparison between Equations 2.8 and 2.10 produces

$$\frac{\partial P}{\partial \theta_1} = \hat{z}_1 \times P\ ;\ \frac{\partial P}{\partial \theta_2} = \hat{z}_2\ ;\ \frac{\partial P}{\partial \theta_3} = \hat{z}_3 \times l_3\hat{y}_3 \qquad (2.11)$$

Apparently, the influence of a unit $\dot{\theta}_1$ on $\dot{P}$ is the cross product of the joint 1 pivot axis and the moment arm between joint 1 and p. Similarly, joint 3 is revolute, so every unit $\dot{\theta}_3$ adds a $\hat{z}_3 \times l_3\hat{y}_3$ vector to $\dot{P}$. Since joint 2 is prismatic, the contribution of unit $\dot{\theta}_2$ to $\dot{P}$ is simply $\hat{z}_2$. Such results could be explained by the definition of partial differentiation. Recalling that

$$\frac{\partial P}{\partial \theta_1} = \left(\frac{\partial P}{\partial \theta_1}\right)\Big|_{\theta_2=constant,\theta_3=constant}$$

which means that the above partial operation is taken at fixed θ_2 and θ_3. In Figure 2.2, the effect of unit variation of θ_1 on P with fixed θ_2 and θ_3 is exactly $\hat{z}_1 \times P$. Similar explanations could be given to the partial operations on θ_2 and θ_3. Following the interpretations and also example results, a general translational influence coefficient table is constructed below.

Table 2.1 Translational G Functions: ${}_TG_j$		
ith Joint Type	$({}_TG_j)_i$	Conditions
Revolute (R)	$\hat{S}_i \times r_{ij}$	$i \leq j$
R	0	$j < i$
Prismatic (P)	$\hat{S}_i$	$i \leq j$
P	0	$j < i$

In the above table, ${}_TG_j$ denotes the translational G function of any point on the jth link and $({}_TG_j)_i$ is its ith column vector. Also $\hat{S}_i$ is the unit vector of the ith joint axis, and r_{ij} is the positional vector from the ith joint to the particular point on the jth link. For an n-link robot, ${}_TG_j \in \mathcal{R}^{3\times n}$, and $i, j \in \{1, 2, \cdots, n\}$. Equation 2.11 results could be verified directly from Table 2.1. A similar table can be built for rotational G function, but the details of Equation 2.9 are worked out first.

$$\begin{aligned} \dot{\Phi} &= [\hat{z}_1\ 0\ \hat{z}_3]\dot{\theta} \\ &= {}_RG_p\dot{\theta} \end{aligned} \tag{2.12}$$

The above results could be checked easily. Since each revolute joint contributes to $\dot{\Phi}$ an angular velocity around the joint axis, $\hat{z}_1$ and $\hat{z}_3$ constitute separately the first and third column of ${}_RG_p$. However, joint 2 is prismatic which does not change the orientation of robot, so the second column of ${}_RG_p$ is a null vector. The content of a general rotational G function is tabulated below.

Table 2.2 Rotational G Functions: ${}_RG_j$		
ith Joint Type	$({}_RG_j)_i$	Conditions
R	$\hat{S}_i$	$i \leq j$
R	0	$j < i$
P	0	$i \leq j$
P	0	$j < i$

where $i, j \in \{1, 2, \cdots, n\}$ for an n-link robot. Obviously, only revolute joint contributes angular motion to a given robot. However, a revolute joint affects

only the part of robot located at the downstream of the particular joint. This means that if the ith joint is revolute then it only affects the orientation of the jth link with every $j \geq i$, otherwise $({}_R G_j)_i = 0$ for all $i > j$. This simple but important fact will be applied to the derivation of the second order influence coefficients introduced in the next section.

2.3 The Second Order Influence Coefficients

In the last section, G functions are introduced from velocity derivation. In this section, the second order influence coefficients, or H functions, will be defined from acceleration equations. First, the translational velocity of the three-link example, Equation 2.8, is differentiated with respect to time which produces the translational acceleration

$$\begin{aligned}
\ddot{P} &= {}_T G_p \ddot{\theta} + {}_T \dot{G}_p \dot{\theta} = {}_T G_p \ddot{\theta} + [\frac{d}{dt}(\frac{\partial P}{\partial \theta_1}) \; \frac{d}{dt}(\frac{\partial P}{\partial \theta_2}) \; \frac{d}{dt}(\frac{\partial P}{\partial \theta_3})]\dot{\theta} \\
&= {}_T G_p \ddot{\theta} + [\dot{\theta}_1 \; \dot{\theta}_2 \; \dot{\theta}_3] \begin{bmatrix} \frac{\partial^2 P}{\partial \theta_1 \partial \theta_1} & \frac{\partial^2 P}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 P}{\partial \theta_3 \partial \theta_1} \\ \frac{\partial^2 P}{\partial \theta_1 \partial \theta_2} & \frac{\partial^2 P}{\partial \theta_2 \partial \theta_2} & \frac{\partial^2 P}{\partial \theta_3 \partial \theta_2} \\ \frac{\partial^2 P}{\partial \theta_1 \partial \theta_3} & \frac{\partial^2 P}{\partial \theta_2 \partial \theta_3} & \frac{\partial^2 P}{\partial \theta_3 \partial \theta_3} \end{bmatrix} \dot{\theta} \\
&\stackrel{\text{def}}{=} {}_T G_p \ddot{\theta} + \dot{\theta}^T \, {}_T H_p \dot{\theta}
\end{aligned} \tag{2.13}$$

where ${}_T H_p$ is a $3 \times 3 \times 3$ matrix and is defined as the second order translational influence coefficient, or simply translational H function. For an n-link robot, ${}_T H_p \in \mathcal{R}^{3 \times n \times n}$; furthermore, let $({}_T H_p)_{ki}$ be the kth row and ith column element of ${}_T H_p$, then

$$({}_T H_p)_{ki} = \frac{\partial^2 P}{\partial \theta_k \partial \theta_i} \in \mathcal{R}^3$$

which means that ${}_T H_p$ is a Hessian matrix. The entries of ${}_T H_p$ in Equation 2.13 are developed in the following equations. The first column of ${}_T H_p$

is obtained by the relation

$$\begin{aligned}
\frac{d}{dt}(\frac{\partial P}{\partial \theta_1}) &= \frac{d}{dt}(\hat{z}_1 \times P) \\
&= \dot{\hat{z}}_1 \times P + \hat{z}_1 \times \dot{P} \\
&= \hat{z}_1 \times [\dot{\theta}_1 \hat{z}_1 \times P + \dot{\theta}_2 \hat{z}_2 + \dot{\theta}_3(\hat{z}_3 \times l_3 \hat{y}_3)] \\
&= [\dot{\theta}_1\ \dot{\theta}_2\ \dot{\theta}_3] \begin{bmatrix} \hat{z}_1 \times (\hat{z}_1 \times P) \\ \hat{z}_1 \times \hat{z}_2 \\ \hat{z}_1 \times (\hat{z}_3 \times l_3 \hat{y}_3) \end{bmatrix}
\end{aligned} \tag{2.14}$$

where $\dot{\hat{z}}_1 = 0$ and the second vector in the above equation is the first column of ${}_T H_p$. Similarly,

$$\begin{aligned}
\frac{d}{dt}(\frac{\partial P}{\partial \theta_2}) &= \dot{\hat{z}}_2 \\
&= \dot{\theta}_1 \hat{z}_1 \times \hat{z}_2 \\
&= \dot{\theta}^T \begin{bmatrix} \hat{z}_1 \times \hat{z}_2 \\ 0 \\ 0 \end{bmatrix}
\end{aligned} \tag{2.15}$$

and

$$\begin{aligned}
\frac{d}{dt}(\frac{\partial P}{\partial \theta_3}) &= \frac{d}{dt}(\hat{z}_3 \times l_3 \hat{y}_3) \\
&= \dot{\hat{z}}_3 \times l_3 \hat{y}_3 + \hat{z}_3 \times l_3 \dot{\hat{y}}_3 \\
&= \dot{\theta}_1(\hat{z}_1 \times \hat{z}_3) \times l_3 \hat{y}_3 + \hat{z}_3 \times l_3(\dot{\theta}_1 \hat{z}_1 \times \hat{y}_3 + \dot{\theta}_3 \hat{z}_3 \times \hat{y}_3) \\
&= \dot{\theta}_1 l_3[(\hat{z}_1 \times \hat{z}_3) \times \hat{y}_3 + \hat{z}_3 \times (\hat{z}_1 \times \hat{y}_3)] \\
&\quad + \dot{\theta}_3 l_3 \hat{z}_3 \times (\hat{z}_3 \times \hat{y}_3)
\end{aligned} \tag{2.16}$$

The last equation could be further simplified by using the vector triple product identity that $a \times (b \times c) = (a \cdot c)b - (a \cdot b)c$ for any $\mathcal{R}^3$ vectors a, b, c defined in an orthogonal frame. Therefore, from Equation 2.16,

$$\begin{aligned}
&(\hat{z}_1 \times \hat{z}_3) \times \hat{y}_3 + \hat{z}_3 \times (\hat{z}_1 \times \hat{y}_3) \\
=\ & -(\hat{y}_3 \cdot \hat{z}_3)\hat{z}_1 + (\hat{y}_3 \cdot \hat{z}_1)\hat{z}_3 + (\hat{z}_3 \cdot \hat{y}_3)\hat{z}_1 - (\hat{z}_3 \cdot \hat{z}_1)\hat{y}_3 \\
=\ & \hat{z}_1 \times (\hat{z}_3 \times \hat{y}_3)
\end{aligned} \tag{2.17}$$

and hence

$$\frac{d}{dt}(\frac{\partial P}{\partial \theta_3}) = \dot{\theta}^T \begin{bmatrix} \hat{z}_1 \times (\hat{z}_3 \times l_3\hat{y}_3) \\ 0 \\ \hat{z}_3 \times (\hat{z}_3 \times l_3\hat{y}_3) \end{bmatrix} \tag{2.18}$$

By combining the above results, ${}_TH_p$ is given by

$$\begin{aligned} {}_TH_p &= \begin{bmatrix} \frac{\partial^2 P}{\partial\theta_1\partial\theta_1} & \frac{\partial^2 P}{\partial\theta_2\partial\theta_1} & \frac{\partial^2 P}{\partial\theta_3\partial\theta_1} \\ \frac{\partial^2 P}{\partial\theta_1\partial\theta_2} & \frac{\partial^2 P}{\partial\theta_2\partial\theta_2} & \frac{\partial^2 P}{\partial\theta_3\partial\theta_2} \\ \frac{\partial^2 P}{\partial\theta_1\partial\theta_3} & \frac{\partial^2 P}{\partial\theta_2\partial\theta_3} & \frac{\partial^2 P}{\partial\theta_3\partial\theta_3} \end{bmatrix} \\ &= \begin{bmatrix} \hat{z}_1 \times (\hat{z}_1 \times P) & \hat{z}_1 \times \hat{z}_2 & \hat{z}_1 \times (\hat{z}_3 \times l_3\hat{y}_3) \\ \hat{z}_1 \times \hat{z}_2 & 0 & 0 \\ \hat{z}_1 \times (\hat{z}_3 \times l_3\hat{y}_3) & 0 & \hat{z}_3 \times (\hat{z}_3 \times l_3\hat{y}_3) \end{bmatrix} \end{aligned} \tag{2.19}$$

Actually, following the kinematic relation of a given robot, the entries of a general translational H function could be constructed directly from the next table.

Table 2.3 Translational H Functions: ${}_TH_j$

kth Joint Type	ith Joint Type	$({}_TG_j)_i$	$({}_TH_j)_{ki}$	Conditions
R	R	$\hat{S}_i \times r_{ij}$	$\hat{S}_k \times (\hat{S}_i \times r_{ij})$	$k \le i \le j$
R	R	$\hat{S}_i \times r_{ij}$	$\hat{S}_i \times (\hat{S}_k \times r_{kj})$	$i \le k \le j$
R	R	$\hat{S}_i \times r_{ij}$	0	$i \le j < k$
R	R	0	0	$j < i$, all k
P	R	$\hat{S}_i \times r_{ij}$	0	$k < i \le j$
P	R	$\hat{S}_i \times r_{ij}$	$\hat{S}_i \times \hat{S}_k$	$i < k \le j$
P	R	$\hat{S}_i \times r_{ij}$	0	$i \le j < k$
P	R	0	0	$j < i$, all k
R	P	$\hat{S}_i$	$\hat{S}_k \times \hat{S}_i$	$k < i \le j$
R	P	$\hat{S}_i$	0	$k \ge i$ and $j \ge i$
R	P	0	0	$j < i$, all k
P	P	$\hat{S}_i$	0	$i \le j$, all k
P	P	0	0	$j < i$, all k

In Table 2.3, ${}_TH_j$ represents the translational H function of a given point in the jth link, $\hat{S}_i$ and $\hat{S}_k$ are respectively the ith and kth joint axes, r_{tj} is the position vector between joint t and the particular point on link j, where $t = i, k$, and $({}_TH_j)_{ki} \in \mathcal{R}^3$ is the kth row and ith column element

of ${}_TH_j$. For an n-link robot, $k, i \in \{1, 2, \cdots, n\}$. In the third column of Table 2.3, the ${}_TG_j$ listed in Table 2.1 are repeated for clarity. Obviously, the values of $({}_TH_j)_{ki}$ rely on the geometric location of link j and joints i and k as well as the joint types of i and k. Notice that ${}_TH_j$ is symmetric, i.e., $({}_TH_j)_{ki} = ({}_TH_j)_{ik}$, which could be observed from Equation 2.19.

From Equation 2.12, the angular acceleration of point p is

$$\ddot{\Phi} = {}_RG_p\ddot{\theta} + \dot{\theta}^T\ {}_RH_p\dot{\theta} \tag{2.20}$$

where

$$\begin{aligned} {}_RH_p &= \begin{bmatrix} \frac{\partial^2\Phi}{\partial\theta_1\partial\theta_1} & \frac{\partial^2\Phi}{\partial\theta_2\partial\theta_1} & \frac{\partial^2\Phi}{\partial\theta_3\partial\theta_1} \\ \frac{\partial^2\Phi}{\partial\theta_1\partial\theta_2} & \frac{\partial^2\Phi}{\partial\theta_2\partial\theta_2} & \frac{\partial^2\Phi}{\partial\theta_3\partial\theta_2} \\ \frac{\partial^2\Phi}{\partial\theta_1\partial\theta_3} & \frac{\partial^2\Phi}{\partial\theta_2\partial\theta_3} & \frac{\partial^2\Phi}{\partial\theta_3\partial\theta_3} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & \hat{z}_1 \times \hat{z}_3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{2.21}$$

Similar result could be obtained from the following rotational H function table.

Table 2.4 Rotational H Function: ${}_RH_j$

kth Joint Type	ith Joint Type	$({}_RG_j)_i$	$({}_RH_j)_{ki}$	Conditions
R	R	$\hat{S}_i$	$\hat{S}_k \times \hat{S}_i$	$k < i \leq j$
R	R	$\hat{S}_i$	0	$k \geq i$ and $j \geq i$
R	R	0	0	$j < i$, all k
P	R	$\hat{S}_i$	0	$i \leq j$, all k
P	R	0	0	$j < i$, all k
P	P	0	0	all i, j, k
R	P	0	0	all i, j, k

As mentioned before, a prismatic joint does not change robot orientation, so $({}_RH_j)_{ki} = 0$ when the kth or ith joint is prismatic. Also, a revolute joint only affects the motion of its downstream links, so $({}_RH_j)_{ki} = 0$ when the kth

joint is revolute and $k \geq i$. This simple relation makes ${}_RH_j$ asymmetric as shown in Equation 2.21, i.e., $({}_RH_j)_{ki} \neq ({}_RH_j)_{ik}$, although ${}_RH_j$ is a Hessian matrix.

2.4 The Lagrange Dynamic Equations

The G and H functions defined in the last sections will be very handy in formulating the robotic system dynamics that will be derived by the Lagrange method in this section. For an n-link robot, let v_i denote the generalized actuator force at the ith joint, that is, when the ith joint is prismatic then v_i represents the joint actuating force, or, if the ith joint is revolute then v_i is the joint driving torque. Also, let θ_i be the ith joint displacement, and θ be an n-dimensional vector with θ_i as the ith element. Then the Lagrange dynamic equations are given by

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{\theta}_i}\right) - \frac{\partial KE}{\partial \theta_i} + \frac{\partial PE}{\partial \theta_i} = v_i \tag{2.22}$$

where KE and PE are system kinetic and potential energy separately. In the following derivations, only inertial dynamics are considered, and actuator forces are the major forces driving the system. The dynamic equation derivations will be divided into kinetic and potential energy parts.

2.4.1 Lagrange Dynamics of Kinetic Energy Part

For a general serial robot, let P_j be the center of mass location of the jth link, and $\dot{\Phi}_j$ be the angular velocity of the same link. Both P_j and $\dot{\Phi}_j$ are $\mathcal{R}^3$ vectors measured with respect to a given global frame. Then the kinetic energy has an expression of

$$KE = \frac{1}{2}\sum_{j=1}^{N}[m_j \dot{P}_j^T \dot{P}_j + (\dot{\Phi}_j^j)^T I_j \dot{\Phi}_j^j] \tag{2.23}$$

where m_j is the mass of the jth link, I_j is the moment of inertia computed with respect to the jth link center of mass and defined by the jth link local frame, and the superscript j indicates that the physical quantity is defined by the jth local frame. In Equation 2.23, N denotes the total number of links, and payload could be treated as an additional fixed link attached to the robot gripper. Let T_j be the coordinate transformation matrix transforming the jth local frame back to the global frame, then

$$\begin{aligned} \dot{\Phi}_j &= {}_R G_j \dot{\theta} \\ &= T_j \dot{\Phi}_j^j \end{aligned} \tag{2.24}$$

and by the orthogonal matrix property that $T_j^T = T_j^{-1}$, $\dot{\Phi}_j^j$ is given by

$$\dot{\Phi}_j^j = T_j^T \, {}_R G_j \dot{\theta} \tag{2.25}$$

In the above equations, the angular velocity is expressed in terms of rotational G function. Similarly, $\dot{P}_j = {}_T G_j \dot{\theta}$ could be introduced into the KE definition in Equation 2.23 which becomes

$$\begin{aligned} KE &= \frac{1}{2} \sum_{j=1}^{N} (m_j \dot{\theta}^T \, {}_T G_j^T \, {}_T G_j \dot{\theta} + \dot{\theta}^T \, {}_R G_j^T T_j I_j T_j^T \, {}_R G_j \dot{\theta}) \\ &\stackrel{\text{def}}{=} \frac{1}{2} \dot{\theta}^T I^* \dot{\theta} \end{aligned} \tag{2.26}$$

where

$$I^* = \sum_{j=1}^{N} (m_j \, {}_T G_j^T \, {}_T G_j + \, {}_R G_j^T T_j I_j T_j^T \, {}_R G_j) \in \mathcal{R}^{n \times n} \tag{2.27}$$

Notice that I^* is function of θ only, i.e., $I^* = I^*(\theta)$, besides that since $KE > 0$, I^* is a positive definite matrix and $I^* = (I^*)^T$. The kinetic energy part of Equation 2.22, i.e.,

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \theta_i}\right) - \frac{\partial KE}{\partial \theta_i}$$

is further divided into a sum of mass and moment of inertia components, or

$$\frac{d}{dt}(\frac{\partial KE}{\partial \dot{\theta}_i}) - \frac{\partial KE}{\partial \theta_i} = \frac{d}{dt}(\frac{\partial KE_m}{\partial \dot{\theta}_i}) - \frac{\partial KE_m}{\partial \theta_i} + \frac{d}{dt}(\frac{\partial KE_I}{\partial \dot{\theta}_i}) - \frac{\partial KE_I}{\partial \theta_i}$$

where

$$KE_m = \frac{1}{2}\sum_{j=1}^{N} m_j \dot{\theta}^T \ {}_T G_j^T \ {}_T G_j \dot{\theta} \tag{2.28}$$

$$KE_I = \frac{1}{2}\sum_{j=1}^{N} \dot{\theta}^T \ {}_R G_j^T T_j I_j T_j^T \ {}_R G_j \dot{\theta} \tag{2.29}$$

.Derivation of $\frac{d}{dt}(\frac{\partial}{\partial \dot{\theta}_i} KE_m) - \frac{\partial}{\partial \theta_i} KE_m$

Let $({}_T G_j)_i$ denote the ith column vector of ${}_T G_j$ and $({}_T G_j)_i \in \mathcal{R}^3$, then a substitution of the KE_m expression defined by Equation 2.29 into $(\frac{\partial}{\partial \dot{\theta}_i} KE_m)$ produces

$$\frac{\partial KE_m}{\partial \dot{\theta}_i} = \sum_{j=1}^{N} m_j ({}_T G_j)_i^T \ {}_T G_j \dot{\theta} \tag{2.30}$$

then the time derivative of the above equation results in

$$\begin{aligned} \frac{d}{dt}(\frac{\partial KE_m}{\partial \dot{\theta}_i}) = & \sum_{j=1}^{N} m_j \left(\sum_{k=1}^{n} \frac{\partial^2 P_j}{\partial \theta_k \partial \theta_i} \dot{\theta}_k\right)^T \ {}_T G_j \dot{\theta} + \sum_{j=1}^{N} m_j ({}_T G_j)_i^T \ {}_T G_j \ddot{\theta} \\ & + \sum_{j=1}^{N} m_j ({}_T G_j)_i^T (\dot{\theta}^T \ {}_T H_j \dot{\theta}) \end{aligned} \tag{2.31}$$

where the second order partial $(\frac{\partial^2}{\partial \theta_k \partial \theta_i} P_j)$ is the second order property of P_j which is exactly the kth row and ith column entry of ${}_T H_j$ given in Table 2.3. Similar second order elements will show up in $(\frac{\partial}{\partial \theta_i} KE_m)$ which is

$$\frac{\partial KE_m}{\partial \theta_i} = \sum_{j=1}^{N} m_j \left(\sum_{k=1}^{n} \frac{\partial^2 P_j}{\partial \theta_i \partial \theta_k} \dot{\theta}_k\right)^T \ {}_T G_j \dot{\theta} \tag{2.32}$$

Since translational H function is symmetric, then

$$\frac{\partial^2 P_j}{\partial \theta_i \partial \theta_k} = \frac{\partial^2 P_j}{\partial \theta_k \partial \theta_i}$$

therefore, the second order terms in Equations 2.31 and 2.32 are canceled in the final dynamic expression of KE_m which is given below

$$\frac{d}{dt}(\frac{\partial KE_m}{\partial \dot{\theta}_i}) - \frac{\partial KE_m}{\partial \theta_i} = \sum_{j=1}^{N}[m_j({}_TG_j)_i^T\ {}_TG_j\ddot{\theta} + m_j({}_TG_j)_i^T(\dot{\theta}^T\ {}_TH_j\dot{\theta})] \quad (2.33)$$

.Derivation of $\frac{d}{dt}(\frac{\partial}{\partial \dot{\theta}_i}KE_I) - \frac{\partial}{\partial \theta_i}KE_I$

Set $({}_RG_j)_i$ be the ith column vector of ${}_RG_j$ and $({}_RG_j)_i \in \mathcal{R}^3$, then from Equation 2.29,

$$\begin{aligned}\frac{d}{dt}(\frac{\partial KE_I}{\partial \dot{\theta}_i}) &= \sum_{j=1}^{N}({}_R\dot{G}_j)_i^T T_j I_j T_j^T\ {}_RG_j\dot{\theta} + \sum_{j=1}^{N}({}_RG_j)_i^T \dot{T}_j I_j T_j^T\ {}_RG_j\dot{\theta} \\ &+ \sum_{j=1}^{N}({}_RG_j)_i^T T_j I_j \dot{T}_j^T\ {}_RG_j\dot{\theta} + \sum_{j=1}^{N}({}_RG_j)_i^T T_j I_j T_j^T\ {}_RG_j\ddot{\theta} \\ &+ \sum_{j=1}^{N}({}_RG_j)_i^T T_j I_j T_j^T(\dot{\theta}^T\ {}_RH_j\dot{\theta}) \end{aligned} \quad (2.34)$$

It will be shown that the third term in the right-hand side of the above equation is actually zero. According to Equation 2.7, the transformation matrix T_j is composed of the local unit vectors

$$T_j = [\hat{x}_j\ \hat{y}_j\ \hat{z}_j]$$

Since the unit vectors $\{\hat{x}_j, \hat{y}_j, \hat{z}_j\}$ change orientation due to the angular motion of link j, $\dot{T}_j$ is given by

$$\dot{T}_j = [(\dot{\Phi}_j \times \hat{x}_j)\,(\dot{\Phi}_j \times \hat{y}_j)\,(\dot{\Phi}_j \times \hat{z}_j)]$$

Also by the G function definition,

$$\dot{\Phi}_j = {}_RG_j\dot{\theta}$$

then the $(\dot{T}_j^T\ {}_RG_j\dot{\theta})$ part in the third term of Equation 2.34 could be expanded into

$$\dot{T}_j^T\ {}_RG_j\dot{\theta} = \dot{T}_j^T\dot{\Phi}_j = \begin{bmatrix} (\dot{\Phi}_j \times \hat{x}_j)^T\dot{\Phi}_j \\ (\dot{\Phi}_j \times \hat{y}_j)^T\dot{\Phi}_j \\ (\dot{\Phi}_j \times \hat{z}_j)^T\dot{\Phi}_j \end{bmatrix} = \begin{bmatrix} (\dot{\Phi}_j \times \hat{x}_j)\cdot\dot{\Phi}_j \\ (\dot{\Phi}_j \times \hat{y}_j)\cdot\dot{\Phi}_j \\ (\dot{\Phi}_j \times \hat{z}_j)\cdot\dot{\Phi}_j \end{bmatrix} \quad (2.35)$$

Following the vector product identities, $(a \times b) \cdot c = (c \times a) \cdot b$ and $\dot{\Phi}_j \times \dot{\Phi}_j = 0$, Equation 2.35 results in a null vector, so the third term in Equation 2.34 vanishes. Further reduction of Equation 2.34 is possible after introducing the following expansions. First $(\frac{\partial}{\partial \theta_i} KE_I)$ is expanded into two terms

$$\begin{aligned} \frac{\partial KE_I}{\partial \theta_i} &= \sum_{j=1}^{N} \dot{\theta}^T \left(\frac{\partial}{\partial \theta_i} {}_R G_j^T \right) T_j I_j T_j^T {}_R G_j \dot{\theta} \\ &\quad + \sum_{j=1}^{N} \dot{\theta}^T {}_R G_j^T \left(\frac{\partial T_j}{\partial \theta_i} \right) I_j T_j^T {}_R G_j \dot{\theta} \end{aligned} \tag{2.36}$$

In the second term of the above equation, it can be concluded from robot kinematic relation that

$$\frac{\partial T_j}{\partial \theta_i} = \begin{cases} 0 & \theta_i \text{ is prismatic or revolute but } i > j \\ [(\hat{S}_i \times \hat{x}_j)\,(\hat{S}_i \times \hat{y}_j)\,(\hat{S}_i \times \hat{z}_j)] & \theta_i \text{ revolute and } i \leq j \end{cases}$$

So the following analysis concentrates on the case where θ_i is revolute and $i \leq j$. Since $(I_j T_j^T {}_R G_j \dot{\theta}) \in \mathcal{R}^3$, let

$$I_j T_j^T {}_R G_j \dot{\theta} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \in \mathcal{R}^3$$

then

$$\begin{aligned} \frac{\partial T_j}{\partial \theta_i} (I_j T_j^T {}_R G_j \dot{\theta}) &= (\hat{S}_i \times \hat{x}_j) c_1 + (\hat{S}_i \times \hat{y}_j) c_2 + (\hat{S}_i \times \hat{z}_j) c_3 \\ &= \hat{S}_i \times \left([\hat{x}_j\ \hat{y}_j\ \hat{z}_j] \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \right) \\ &= \hat{S}_i \times (T_j I_j T_j^T {}_R G_j \dot{\theta}) \end{aligned} \tag{2.37}$$

therefore the second term of Equation 2.36 could be expressed as

$$\begin{aligned} &\sum_{j=1}^{N} \dot{\theta}^T {}_R G_j^T \left(\frac{\partial T_j}{\partial \theta_i} \right) I_j T_j {}_R G_j \dot{\theta} \\ &= \sum_{j=1}^{N} \dot{\Phi}_j \cdot [\hat{S}_i \times (T_j I_j T_j^T {}_R G_j \dot{\theta})] \\ &= \sum_{j=1}^{N} -\hat{S}_i \cdot [\dot{\Phi}_j \times (T_j I_j T_j^T {}_R G_j \dot{\theta})] \end{aligned} \tag{2.38}$$

It is now shown that the above equation has a negative counterpart and is neutralized in the final expression of

$$\frac{d}{dt}\left(\frac{\partial KE_I}{\partial \dot{\theta}_i}\right) - \frac{\partial KE_I}{\partial \theta_i}$$

Subtracting the first term of Equation 2.36 from the first term of Equation 2.34 produces

$$\begin{aligned} &\sum_{j=1}^{N} ({}_R\dot{G}_j)_i^T T_j I_j T_j^T \, {}_R G_j \dot{\theta} - \sum_{j=1}^{N} \dot{\theta}^T \left(\frac{\partial}{\partial \theta_i} {}_R G_j^T\right) T_j I_j T_j^T \, {}_R G_j \dot{\theta} \\ = &\sum_{j=1}^{N} \left(\sum_{k=1}^{n} \frac{\partial^2 \Phi_j}{\partial \theta_k \partial \theta_i} \dot{\theta}_k - \sum_{k=1}^{n} \frac{\partial^2 \Phi_j}{\partial \theta_i \partial \theta_k} \dot{\theta}_k\right) \cdot (T_j I_j T_j^T \, {}_R G_j \dot{\theta}) \end{aligned} \tag{2.39}$$

where each second order partial term represents an entry of rotational H function listed in Table 2.4. By Tables 2.4, it can be shown for revolute θ_i that

$$\begin{aligned} \frac{\partial^2 \Phi_j}{\partial \theta_k \partial \theta_i} &= \begin{cases} \delta_k \hat{S}_k \times \hat{S}_i & \text{if } j \geq i > k \\ 0 & \text{otherwise} \end{cases} \\ \frac{\partial^2 \Phi_j}{\partial \theta_i \partial \theta_k} &= \begin{cases} \hat{S}_i \times \delta_k \hat{S}_k & \text{if } j \geq k > i \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where

$$\delta_k = \begin{cases} 1 & \text{if } \theta_k \text{ is revolute} \\ 0 & \text{if } \theta_k \text{ is prismatic} \end{cases} \quad ; k \in \{1, 2, \cdots, j\} \tag{2.40}$$

For $j \geq i$, the second order terms in Equation 2.39 have the following compact results

$$\begin{aligned} &\sum_{k=1}^{n} \frac{\partial^2 \Phi_j}{\partial \theta_k \partial \theta_i} \dot{\theta}_k - \sum_{k=1}^{n} \frac{\partial^2 \Phi_j}{\partial \theta_i \partial \theta_k} \dot{\theta}_k \\ = &\left(\left[\frac{\partial^2 \Phi_j}{\partial \theta_1 \partial \theta_i} \; \frac{\partial^2 \Phi_j}{\partial \theta_2 \partial \theta_i} \; \cdots \frac{\partial^2 \Phi_j}{\partial \theta_n \partial \theta_i}\right] - \left[\frac{\partial^2 \Phi_j}{\partial \theta_i \partial \theta_1} \; \frac{\partial^2 \Phi_j}{\partial \theta_i \partial \theta_2} \; \cdots \frac{\partial^2 \Phi_j}{\partial \theta_i \partial \theta_n}\right]\right) \dot{\theta} \\ = &([(\delta_1 \hat{S}_1 \times \hat{S}_i)(\delta_2 \hat{S}_2 \times \hat{S}_i) \cdots (\delta_{i-1} \hat{S}_{i-1} \times \hat{S}_i) \, 0 \cdots 0] \\ &- [0 \cdots 0 \cdot (\hat{S}_i \times \delta_{i+1} \hat{S}_{i+1})(\hat{S}_i \times \delta_{i+2} \hat{S}_{i+2}) \cdots (\hat{S}_i \times \delta_j \hat{S}_j) \, 0 \cdots 0]) \dot{\theta} \\ = &[(\delta_1 \hat{S}_1 \times \hat{S}_i)(\delta_2 \hat{S}_2 \times \hat{S}_i) \cdots (\delta_j \hat{S}_j \times \hat{S}_i) \, 0 \cdots 0] \dot{\theta} \end{aligned}$$

$$
\begin{aligned}
&= -\hat{S}_i \times ([\delta_1 \hat{S}_1 \; \delta_2 \hat{S}_2 \; \cdots \; \delta_j \hat{S}_j \; 0 \; \cdots \; 0]\dot{\theta}) \\
&= -\hat{S}_i \times \dot{\Phi}_j
\end{aligned} \tag{2.41}
$$

where according to Table 2.2 and Equation 2.40

$$
\dot{\Phi}_j = [\delta_1 \hat{S}_1 \; \delta_2 \hat{S}_2 \; \cdots \; \delta_j \hat{S}_j \; 0 \; \cdots \; 0]\dot{\theta} \tag{2.42}
$$

In Equation 2.41, the fact that $\hat{S}_i \times \hat{S}_i = 0$ has been used to complete the formulation. Then Equation 2.39 becomes

$$
\begin{aligned}
&\sum_{j=1}^{N} (-\hat{S}_i \times \dot{\Phi}_j) \cdot (T_j I_j T_j^T \; {}_R G_j \dot{\theta}) \\
&= \sum_{j=1}^{N} -\hat{S}_i \cdot [\dot{\Phi}_j \times (T_j I_j T_j^T \; {}_R G_j \dot{\theta})]
\end{aligned} \tag{2.43}
$$

which is exactly the counterpart of Equation 2.38, and both will cancel each other in the final form. Notice that although the above derivations are mainly for revolute θ_i and $j \geq i$, the cancellation is still valid for prismatic θ_i or revolute θ_i when $j < i$, because both terms are identically zero in these cases. After eliminating all zero terms, the dynamic equation of KE_I part reduces to

$$
\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial KE_I}{\partial \dot{\theta}_i}\right) - \frac{\partial KE_I}{\partial \theta_i} \\
&= \sum_{j=1}^{N} ({}_R G_j)_i^T [\dot{\Phi}_j \times (T_j I_j T_j^T \; {}_R G_j \dot{\theta})] \\
&\quad + \sum_{j=1}^{N} ({}_R G_j)_i^T [(T_j I_j T_j^T)({}_R G_j \ddot{\theta} + \dot{\theta}^T \; {}_R H_j \dot{\theta})]
\end{aligned} \tag{2.44}
$$

The final KE part dynamic equations are the sum of Equations 2.33 and 2.44 which are

$$
\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{\theta}_i}\right) - \frac{\partial KE}{\partial \theta_i}
$$

$$
\begin{aligned}
= & \sum_{j=1}^{N} m_j({}_T G)_i^T({}_T G_j\ddot{\theta} + \dot{\theta}^T\,{}_T H_j\dot{\theta}) \\
& + \sum_{j=1}^{N}({}_R G_j)_i^T[\dot{\Phi}_j \times (T_j I_j T_j^T\,{}_R G_j\dot{\theta})] \\
& + \sum_{j=1}^{N}({}_R G_j)_i^T[(T_j I_j T_j^T)({}_R G_j\ddot{\theta} + \dot{\theta}^T\,{}_R H_j\dot{\theta})] \qquad (2.45)
\end{aligned}
$$

2.4.2 Lagrange Dynamics of Potential Energy Part

It is assumed that gravity force is the only external force on the system and the gravitational field is along the global Z direction. Then the gravitational acceleration vector is defined by $\hat{g} = [0\ 0\ g]^T \in \mathcal{R}^3$ where g is the accceleration constant of gravity. The gravitational potential energy of the jth link is given by

$$(PE)_j = m_j P_j^T \hat{g}$$

then

$$PE = \sum_{j=1}^{N} m_j P_j^T \hat{g} \qquad (2.46)$$

and

$$
\begin{aligned}
\frac{\partial PE}{\partial \theta_i} &= \sum_{j=1}^{N} m_j \frac{\partial P_j^T}{\partial \theta_i}\hat{g} \\
&= \sum_{j=1}^{N} m_j({}_T G_j)_i^T \hat{g} \qquad (2.47)
\end{aligned}
$$

With the above result and Equation 2.45, the dynamic equation for the ith actuator is

$$
\begin{aligned}
v_i &= \frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{\theta}_i}\right) - \frac{\partial KE}{\partial \theta_i} + \frac{\partial PE}{\partial \theta_i} \\
&= \sum_{j=1}^{N} m_j({}_T G)_i^T({}_T G_j\ddot{\theta} + \dot{\theta}^T\,{}_T H_j\dot{\theta})
\end{aligned}
$$

$$+\sum_{j=1}^{N}({}_{R}G_j)_i^T[\dot{\Phi}_j \times (T_j I_j T_j^T\ {}_{R}G_j\dot{\theta})]$$
$$+\sum_{j=1}^{N}({}_{R}G_j)_i^T[(T_j I_j T_j^T)({}_{R}G_j\ddot{\theta}+\dot{\theta}^T\ {}_{R}H_j\dot{\theta})]$$
$$+\sum_{j=1}^{N} m_j({}_{T}G_j)_i^T\hat{g} \tag{2.48}$$

Let v be an n-dimensional vector with v_i as the ith element, and recalling that $({}_{T}G_j)_i$ and $({}_{R}G_j)_i$ are the ith column of ${}_{T}G_j$ and ${}_{R}G_j$ respectively, then for an n-link robot the total inertial dynamic equations are

$$\begin{aligned} v &= \frac{d}{dt}\left(\frac{\partial KE}{\partial\dot{\theta}}\right)-\frac{\partial KE}{\partial\theta}+\frac{\partial PE}{\partial\theta} \\ &= I^*\ddot{\theta}+\sum_{j=1}^{N} m_j\ {}_{T}G_j^T(\dot{\theta}^T\ {}_{T}H_j\dot{\theta})+\sum_{j=1}^{N}\ {}_{R}G_j^T[\dot{\Phi}_j\times(T_jI_jT_j^T\ {}_{R}G_j\dot{\theta})] \\ &\quad +\sum_{j=1}^{N}\ {}_{R}G_j^T[(T_jI_jT_j^T)(\dot{\theta}^T\ {}_{R}H_j\dot{\theta})]+\sum_{j=1}^{N} m_j({}_{T}G_j)^T\hat{g} \end{aligned} \tag{2.49}$$

where I^* is the generalized inertial matrix defined by Equation 2.27.

2.5 The Analogous Newton-Euler Dynamics

The dynamic equations derived by the Lagrange method will be reconstructed by the analogous Newton-Euler approach in this section. The Newton-Euler method will verify the Lagrange results and also provide the dynamic equations with a physical interpretation. The linear momentum of the jth link is $m_j\dot{P}$, then the associated inertial force is

$$\begin{aligned} F_j &= \frac{d}{dt}(m_j\dot{P}_j) \\ &= m_j\ddot{P}_j \\ &= m_j({}_{T}G_j\ddot{\theta}+\dot{\theta}^T\ {}_{T}H_j\dot{\theta})\ \in\mathcal{R}^3 \end{aligned} \tag{2.50}$$

where $\ddot{P}_j$ is replaced by the G and H function expressions defined in Equation 2.13. To support this inertial force, the ith joint needs to generate a

generalized actuator force $(v_i)_{F_j}$ whose magnitude is given in the following three cases:

Case 1. If $i > j$, then the ith joint is located downstream of the jth link, so the inertial force creates no effect on $(v_i)_{F_j}$, therefore, $(v_i)_{F_j} = 0$. However, from Table 2.1, $({}_TG_j)_i = 0$ at $i > j$, hence $(v_i)_{F_j}$ could be expressed as

$$(v_i)_{F_j} = m_j({}_TG_j)_i^T({}_TG_j\ddot{\theta} + \dot{\theta}^T\,{}_TH_j\dot{\theta})$$

Case 2. The ith joint is prismatic and $i \leq j$, then $(v_i)_{F_j}$ is a force given by the projection of F_j on the ith joint axis $\hat{S}_i$, or

$$\begin{aligned}(v_i)_{F_j} &= \hat{S}_i^T F_j \\ &= m_j({}_TG_j)_i^T({}_TG_j\ddot{\theta} + \dot{\theta}^T\,{}_TH_j\dot{\theta})\end{aligned}$$

where $({}_TG_j)_i = \hat{S}_i$ is listed in Table 2.1 under the condition that $i \leq j$.

Case 3. The ith joint is revolute and $i \leq j$, then $(v_i)_{F_j}$ is a torque created by

$$\begin{aligned}(v_i)_{F_j} &= \hat{S}_i \cdot (r_{ij} \times F_j) \\ &= (\hat{S}_i \times r_{ij})^T F_j \\ &= m_j({}_TG_j)_i^T({}_TG_j\ddot{\theta} + \dot{\theta}^T\,{}_TH_j\dot{\theta}) \qquad (2.51)\end{aligned}$$

where r_{ij} is the positional vector between the ith joint and the center of mass of the jth link. The vector triple product identity $a \cdot (b \times c) = (a \times b) \cdot c$ is applied in the above equation. For any of the three possible cases, $(v_i)_{F_j}$ could be expressed by the unique form in Equation 2.51.

Let h_j^j be the angular momentum of the jth link evaluated around the center of mass, where $h_j^j \in \mathcal{R}^3$ and is defined in the jth local frame, then

$$\begin{aligned}h_j^j &= I_j\dot{\Phi}_j^j \\ &= I_jT_j^T\,{}_RG_j\dot{\theta} \qquad (2.52)\end{aligned}$$

which could be converted into a global frame description h_j by

$$\begin{aligned} h_j &= T_j h_j^j \\ &= T_j I_j T_j^T \, {}_R G_j \dot{\theta} \end{aligned} \tag{2.53}$$

Set τ_j be the inertial torque associated with h_j and $\tau_j \in \mathcal{R}^3$, then

$$\begin{aligned} \tau_j &= \frac{d}{dt} h_j \\ &= \frac{d}{dt}(T_j I_j T_j^T \, {}_R G_j \dot{\theta}) \\ &= \dot{T}_j I_j T_j^T \, {}_R G_j \dot{\theta} + T_j I_j \dot{T}_j^T \, {}_R G_j \dot{\theta} \\ &\quad + T_j I_j T_j^T (\, {}_R G_j \ddot{\theta} + \dot{\theta}^T \, {}_R H_j \dot{\theta}) \end{aligned} \tag{2.54}$$

According to Equation 2.35 results, $\dot{T}_j^T \, {}_R G_j \dot{\theta} = 0$, so its accompanied term in Equation 2.54 vanishes. Also,

$$\dot{T}_j I_j T_j^T \, {}_R G_j \dot{\theta} = \dot{\Phi}_j \times (T_j I_j T_j^T \, {}_R G_j \dot{\theta}) \tag{2.55}$$

then Equation 2.54 becomes

$$\begin{aligned} \tau_j &= \dot{\Phi}_j \times (T_j I_j T_j^T \, {}_R G_j \dot{\theta}) \\ &\quad + T_j I_j T_j^T (\, {}_R G_j \ddot{\theta} + \dot{\theta}^T \, {}_R H_j \dot{\theta}) \end{aligned} \tag{2.56}$$

Let $(v_i)_{\tau_j}$ be the generalized actuating force at the ith joint to support τ_j, then its value is decided in the following two cases:

Case 1. If the ith joint is prismatic or revolute but $i > j$, then τ_j has no effect on $(v_i)_{\tau_j}$, so $(v_i)_{\tau_j} = 0$. Since by Table 2.2, $(\, {}_R G_j)_i = 0$ in this case, $(v_i)_{\tau_j}$ could be written as

$$\begin{aligned} (v_i)_{\tau_j} &= 0 \cdot \tau_j \\ &= (\, {}_R G_j)_i^T \tau_j \end{aligned}$$

Case 2. The ith joint is revolute and $i \leq j$, then $(v_i)_{\tau_j}$ is the projection of τ_j on the ith joint axis $\hat{S}_i$, or

$$\begin{aligned}(v_i)_{\tau_j} &= \hat{S}_i^T \tau_j \\ &= ({}_R G_j)_i^T \tau_j \end{aligned} \tag{2.57}$$

where the fact that $({}_R G_j)_i = \hat{S}_i$ is used in the above equation. Since in both cases, $(v_i)_{\tau_j}$ has the same expression as in Equation 2.57, so the τ_j given in Equation 2.56 is substituted into Equation 2.57 to produce the final form

$$\begin{aligned}(v_i)_{\tau_j} &= ({}_R G_j)_i^T [\dot{\Phi}_j \times (T_j I_j T_j^T \, {}_R G_j \dot{\theta})] \\ &\quad + ({}_R G_j)_i^T [(T_j I_j T_j^T)({}_R G_j \ddot{\theta} + \dot{\theta}^T \, {}_R H_j \dot{\theta})] \end{aligned} \tag{2.58}$$

Finally, for the gravity force, let $(v_i)_{g_j}$ denote the ith joint force supporting the gravity force of the jth link, then

Case 1. If $j < i$ and i is either revolute or prismatic joint, then

$$\begin{aligned}(v_i)_{g_j} &= 0 \cdot m_j \hat{g} \\ &= m_j ({}_T G_j)_i^T \hat{g} \end{aligned}$$

Case 2. If $j \geq i$ and the ith joint is prismatic, then

$$\begin{aligned}(v_i)_{g_j} &= m_j \hat{S}_i^T \hat{g} \\ &= m_j ({}_T G_j)_i^T \hat{g} \end{aligned}$$

Case 3. If $j \geq i$ and the ith joint is revolute, then

$$\begin{aligned}(v_i)_{g_j} &= m_j \hat{S}_i^T (r_{ij} \times \hat{g}) \\ &= m_j (\hat{S}_i \times r_{ij})^T \hat{g} \\ &= m_j ({}_T G_j)_i^T \hat{g} \end{aligned} \tag{2.59}$$

All three conditions result in the similar form as Equation 2.59. Let v_i be the sum of $(v_i)_{F_j}$, $(v_i)_{\tau_j}$, and $(v_i)_{g_j}$ for all $j \in \{1, 2, 3, \cdots, N\}$, then from Equations 2.51, 2.58, and 2.59,

$$\begin{aligned} v_i &= \sum_{j=1}^{N}[(v_i)_{F_j} + (v_i)_{\tau_j} + (v_i)_{g_j}] \\ &= \sum_{j=1}^{N} m_j({}_T G_j)_i^T({}_T G_j \ddot{\theta} + \dot{\theta}^T {}_T H_j \dot{\theta}) \\ &\quad + \sum_{j=1}^{N}({}_R G_j)_i^T[\dot{\Phi}_j \times (T_j I_j T_j^T {}_R G_j \dot{\theta}) + (T_j I_j T_j^T)({}_R G_j \ddot{\theta} + \dot{\theta}^T {}_R H_j \dot{\theta})] \\ &\quad + \sum_{j=1}^{N} m_j({}_T G_j)_i^T \hat{g} \end{aligned} \tag{2.60}$$

which is exactly the same result derived by the Lagrange equation in Equation 2.48. Notice that in both results, $\dot{\theta}$ and $\ddot{\theta}$ are decoupled from position dependent parameters in the dynamic equations. A physical interpretation of Equation 2.60 is given here. In the first term of Equation 2.60,

$$m_j({}_T G_j \ddot{\theta} + \dot{\theta}^T {}_T H_j \dot{\theta})$$

represents the inertial force on the jth link, where

$$({}_T G_j \ddot{\theta} + \dot{\theta}^T {}_T H_j \dot{\theta})$$

denotes link translational acceleration, and $({}_T G_j)_i$ is the projection vector projecting the inertial force on the ith joint. Similarly, in the second term

$$\dot{\Phi}_j \times (T_j I_j T_j^T {}_R G_j \dot{\theta}) + (T_j I_j T_j^T)({}_R G_j \ddot{\theta} + \dot{\theta}^T {}_R H_j \dot{\theta})$$

is the inertial torque on the jth link due to angular acceleration of the link, i.e.,

$$(T_j I_j T_j^T)({}_R G_j \ddot{\theta} + \dot{\theta}^T {}_R H_j \dot{\theta})$$

and also due to the change of moment of inertia magnitudes in the global frame, which is

$$\dot{\Phi}_j \times (T_j I_j T_j^T \ {}_R G_j \dot{\theta})$$

and $({}_R G_j)_i$ is the projection vector projecting the inertial torque to the ith joint. A similar interpretation could also be given to the gravity force term. Although the geometric location order between the ith joint and the jth link and the joint type will affect the final projected value, such problems are solved automatically by the contents of projection vectors $({}_R G_j)_i$ and $({}_T G_j)_i$ as shown in the previous case discussions. Therefore, the above dynamic equations are general for any i and j due to compactness of G and H functions.

2.6 Summary

In this chapter, rigid robot dynamic equations are derived by both Lagrange and Newton-Euler methods. The purpose of presenting rigid system dynamic equations is to help the reader to get acquainted with the notation and tools used in this report. One of the most useful tools, the coordinate transformation matrix, is introduced first to assist the identification of local coordinate frame. Then, the first and second order influence coefficients, G and H functions, are defined. Simple kinematic relations are used in constructing and interpreting the entries of both functions. Finally G and H functions are applied to system dynamics derivation. One advantage of introducing the G and H functions is the compactness of these notations. Besides that the G and H functions carry the kinematic relations required to define robot motion and actuator driving force. The physical interpretations of the final dynamic equations are given in the last section. Finally, it should be pointed out that the G functions also produce the "partial velocities" and

"partial angular velocities" defined by [Kane and Levinson, 1985], which can be proved directly by using the d'Alembert's principle and the principle of virtual work.

Chapter 3

Dynamic Equations of Compliant Robotic Manipulators

In the previous chapter, we developed system dynamics for rigid robotic manipulators. Physically, no system is absolutely rigid under load. The assumption of rigid body is to simplify system model so that an efficient dynamic description could be obtained with acceptable level of accuracy. One of such occasions that a rigid body model becomes a practical approach is when the system has negligible deformation. For example, the rigid body model is suitable to most current industrial robots which are built with rugged arms but carry small payload and operate in low speed. However, the rigid body assumption is not realistic when structural deformation becomes prominent and consequently affects operation precision. An obvious case where rigid body assumption becomes impractical is in the modeling of a lightweight robot. To increase productivity and economic value, the next generation robots tend to be lighter, faster, and have better precision while carrying larger payloads. Due to the lightweight nature, a heavy payload plus high speed will cause structural deformation and vibration. To maintain precision, it is essential to have a compliant model to help system and control designers solve the structural deformation and residual oscillation problems. The aim of this chapter is to present system dynamics of robotic manipulator modeled with structural compliance.

3.1 Distributed Parameter Model of Compliant Manipulators

There are two popular methods of describing manipulator compliance. The first method is the distributed parameter model, and the second method is the lumped parameter model. This section will introduce the distributed model and save the second method for a later section. Although the distributed model is used mainly on link compliance, for the completeness of the derivation, joint compliance is also included in the following analysis where lumped massless spring is employed to represent joint compliance as suggested by [Good, Sweet, and Strobel, 1984]. The distributed parameter approach defines each robotic link as a continuous beam. Link deformation is analyzed at every point along the link. Then an integration of all points in the link gives the gross vibratory motion. A detailed derivation of compliant manipulator dynamics by the distributed model is reported by [Graves, 1988]. This report will present a simplified version of compliant manipulator dynamics based on the following assumptions. It is assumed that (1) each compliant link has a large slenderness ratio so that the Euler-Bernoulli theory is applicable, (2) small deflections, and (3) deflections are decoupled in all directions. According to the first assumption, the rotatory inertia and shearing effects are negligible. And by the third assumption, coupling effects like the centrifugal stiffening is neglected in the analysis. The link deflections modeled in this section include lateral bendings in principal planes, longitudinal elongation, and twisting along axial direction. Also, each joint compliance is modeled by a lumped spring. The Lagrange method will be applied to derive the compliant system dynamics. As a representative analysis, the kinetic and potential energy are derived for the compliant link j shown in Figure 3.1. In this example, a given point p deflects from its rigid link position to

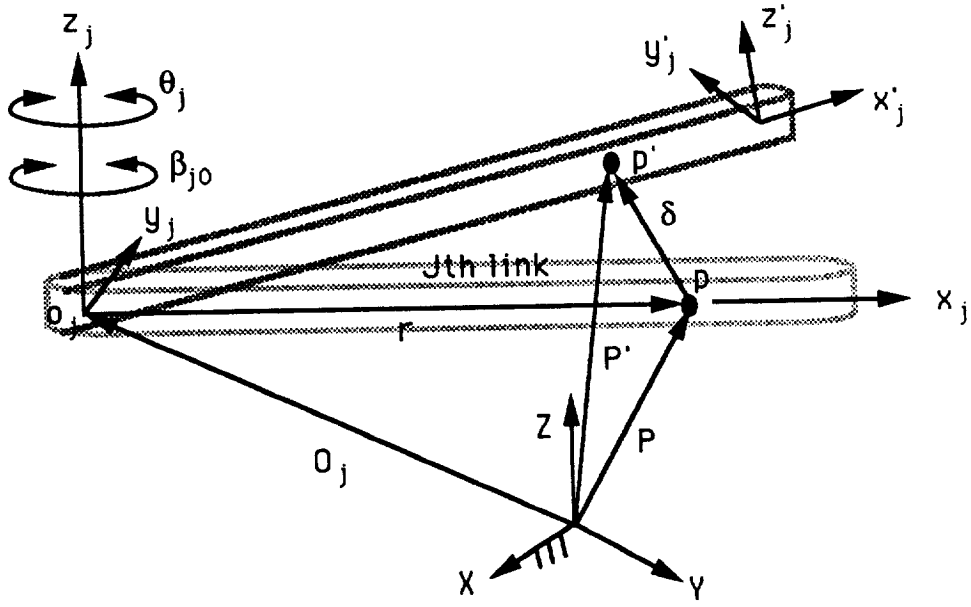


Figure 3.1: Distributed Model of a Compliant link

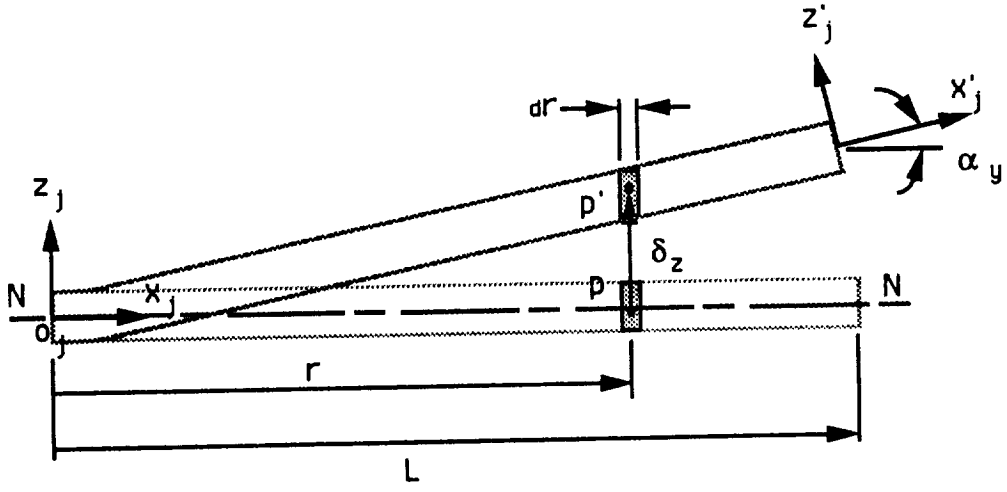


Figure 3.2: Side View of a Compliant Link

a deformed position p'. $\{x_j, y_j, z_j\}$ is the fixed local frame before deformation, and $\{x'_j, y'_j, z'_j\}$ is another local frame attached to the distal end of the deformed link. Both frames are parallel when the link is absolutely rigid. Figure 3.2 shows the $x_j - z_j$ plane view of Figure 3.1. According to the third assumption, Figure 3.2 represents the link bending along z_j direction. In the same figure, $N - N$ is the neutral surface, and p is a point on the neutral axis located at a distance r from the fixed local frame. Let δ denote the deflection vector from p to p', and $\{\hat{x}_j, \hat{y}_j, \hat{z}_j\}$ be the unit $\mathcal{R}^3$ vectors of $\{x_j, y_j, z_j\}$, all unit vectors are defined in the global frame, then δ is composed by

$$\delta(r,t) = \delta_x(r,t)\hat{x}_j + \delta_y(r,t)\hat{y}_j + \delta_z(r,t)\hat{z}_j \in \mathcal{R}^3 \tag{3.1}$$

which indicates that δ is a function of position and time. In the above equation, $\{\delta_x, \delta_y, \delta_z\}$ are the deflections of p along $\{\hat{x}_j, \hat{y}_j, \hat{z}_j\}$, where δ_y and δ_z represent the lateral bendings and δ_x is longitudinal elongation. Now, let $\alpha_x(r,t)$ be the twisting angle of the cross section at point p along $\hat{x}_j$. Then the linear and angular velocities of point p' are derived in the followings. Let O_j be the global position vector of the jth frame, and P and P' be the global position vector of p and p' separately, then

$$P' = O_j + r\hat{x}_j + \delta \in \mathcal{R}^3 \tag{3.2}$$

and

$$\begin{aligned} \dot{P}' &= \dot{O}_j + \dot{r}\hat{x}_j + r\dot{\hat{x}}_j + \dot{\delta} \\ &= \dot{P} + \dot{\delta} \end{aligned} \tag{3.3}$$

Note that if the jth joint is revolute then $\dot{r} = 0$, if the jth joint is prismatic then $\dot{r} = \dot{\theta}_j + \dot{\beta}_{j0}$, where θ_j is the actual joint displacement, and β_{j0} is the joint deflection. Let ω_j and $\dot{\Phi}_j$ be the angular velocity of point p' and p

respectively. Since the rotatory effect is ignored, the angular velocity of p' has the form

$$\omega_j(r,t) = \dot{\Phi}_j + \dot{\alpha}_x(r,t)\hat{x}_j \in \mathcal{R}^3 \tag{3.4}$$

Notice that the joint compliance effect is included in $\dot{\Phi}_j$, i.e, a $\dot{\beta}_{j0}\hat{z}_j$ term is contained in $\dot{\Phi}_j$ if the jth joint is revolute. For the incremental segment surrounding p' in Figure 3.2, let $dm(r)$ be the mass of the segment and $dI(r)$ be the associated mass moment of inertia with $dI(r) \in \mathcal{R}^{3\times 3}$, then the kinematic energy of the segment is

$$2dKE_j = \dot{P}'^T\dot{P}'dm(r) + \omega_j^T T_j(dI(r))T_j^T\omega_j \tag{3.5}$$

The above derivation is general for any given point p at $0 \le r \le L$, so an integration of all such segments over the whole link length L gives the kinetic energy of compliant link j, which is

$$\begin{aligned} 2KE_j &= \int \dot{P}'^T\dot{P}'dm + \int \{\omega_j^T T_j(dI)T^T\omega_j\} \\ &= \int \dot{P}^T\dot{P}dm + \int \dot{P}^T\dot{\delta}dm + \int \dot{\delta}^T\dot{P}dm + \int \dot{\delta}^T\dot{\delta}dm \\ &\quad + \int \dot{\Phi}_j^T T_j(dI)T_j^T\dot{\Phi}_j + \int (\dot{\alpha}_x\hat{x}_j)^T T_j(dI)T_j^T\dot{\Phi}_j \\ &\quad + \int \dot{\Phi}_j^T T_j(dI)T_j^T(\dot{\alpha}_x\hat{x}_j) + \int (\dot{\alpha}_x\hat{x}_j)^T T_j(dI)T_j^T(\dot{\alpha}_x\hat{x}_j) \end{aligned} \tag{3.6}$$

where the part

$$\int \dot{P}^T\dot{P}dm + \int \dot{\Phi}_j^T T_j(dI)T_j^T\dot{\Phi}_j$$

is the rigid link kinetic energy. According to the first and third assumptions, it is shown by [Low, 1987] that the gravitational and elastic potential energy are given by

$$\begin{aligned} PE_j &= \int P'^T\hat{g}dm + \frac{1}{2}\int EI_y(\frac{\partial\alpha_y}{\partial r})^2dr + \frac{1}{2}\int EI_z(\frac{\partial\alpha_z}{\partial r})^2dr \\ &\quad + \frac{1}{2}\int EA(\frac{\partial\delta_x}{\partial r})^2dr + \frac{1}{2}\int GJ_x(\frac{\partial\alpha_x}{\partial r})^2dr + \frac{1}{2}K_{j0}\beta_{j0}^2 \end{aligned} \tag{3.7}$$

where E and G are separately the moduli of elasticity and rigidity, A is the cross section area, I_y and I_z are the area moments of inertia about $\hat{y}_j$ and $\hat{z}_j$ respectively, J_x is the polar moment of inertia along $\hat{x}_j$, K_{j0} is the stiffness of modeled joint spring, and $\hat{g} \in \mathcal{R}^3$ is the gravitational acceleration vector. In Equation 3.7, the rotational angles α_y and α_z are given by

$$\alpha_y = -\frac{\partial \delta_z}{\partial r} \; ; \; \alpha_z = \frac{\partial \delta_y}{\partial r} \tag{3.8}$$

which represent the rotation angles around $\hat{y}_j$ and $\hat{z}_j$ respectively. The gravitational term in Equation 3.7 could be further divided into

$$\int P'^T \hat{g} dm = \int P^T \hat{g} dm + \int \delta^T \hat{g} dm$$

where the first term is the gravitational potential energy of rigid link. To include rotatory inertia effect into the kinetic energy expression, Equation 3.4 is modified by

$$\omega_j = \dot{\Phi}_j + \dot{\alpha}_x \hat{x}_j + \dot{\alpha}_y \hat{y}_j + \dot{\alpha}_z \hat{z}_j \in \mathcal{R}^3 \tag{3.9}$$

then the ω_j in Equations 3.5 and 3.6 are replaced by the above new definition. In order to simplify flexibility dynamics of a continuous system, a finite number of assumed modes are generally employed to discretize the deflections which actually contain infinite degrees of freedom. Therefore, it is assumed that the deflections could be expressed by decoupled forms

$$\begin{aligned}
\delta_x &= \sum_{i=1}^{nx} \beta_{jxi}(t)\phi_{jxi}(r) \\
\delta_y &= \sum_{i=1}^{ny} \beta_{jyi}(t)\phi_{jyi}(r) \\
\delta_z &= \sum_{i=1}^{nz} \beta_{jzi}(t)\phi_{jzi}(r) \\
\alpha_x &= \sum_{i=1}^{n\alpha} \beta_{j\alpha i}(t)\phi_{j\alpha i}(r) \qquad .
\end{aligned} \tag{3.10}$$

where $\{\phi_{jxi}, \phi_{jyi}, \phi_{jzi}, \phi_{j\alpha i}\}$ are the ith mode shapes of $\{\delta_x, \delta_y, \delta_z, \alpha_x\}$ and are functions of r, and $\{\beta_{jxi}, \beta_{jyi}, \beta_{jzi}, \beta_{j\alpha i}\}$ are the associated amplitudes which are time functions. Then kinetic and potential energy expressions in Equations 3.6 and 3.7 could be integrated for all modes shape, and system dynamics are defined by a set of generalized coordinates composed by nominal joint parameters and modal amplitudes. Several often used mode shapes are suggested by [Meirovitch, 1980] for the general continuous beam. For example, polynomial mode shapes with orthogonal nature are used by [Graves, 1988]. Of course, selection of mode shape needs to meet geometric boundary conditions. Since a compliant link is generally modeled as a fixed-free or pinned-free beam, selected mode shapes must satisfy both boundary constraints. Usually simple mode shapes are employed to facilitate integration. Once the mode shapes are defined, the modal amplitudes could be reconstructed from on-line measurements as reported by [Hastings and Book, 1986].

For an n_θ-link robot, the total system kinetic and potential energy are

$$\begin{aligned} KE &= \sum_{j=1}^{N} KE_j \\ PE &= \sum_{j=1}^{N} PE_j \end{aligned} \tag{3.11}$$

Here, payload is treated as a fixed link, and N is the total number of links including payload. By collecting all state variables into a vector q and defining that

$$\begin{aligned} q &= [\theta^T \ \beta^T]^T \\ \theta &= [\theta_1 \ \cdots \theta_{n_\theta}]^T \\ \beta &= [\beta_{10} \, \beta_{1x1} \cdots \beta_{1xnx} \, \beta_{1y1} \cdots \beta_{1yny} \cdots \beta_{1z1} \cdots \beta_{1znz} \, \beta_{1\alpha 1} \cdots \beta_{1\alpha n\alpha} \cdots]^T \end{aligned} \tag{3.12}$$

where θ is the vector of all nominal joint variables, and β is the collection of all joint and link vibratory amplitudes. Then the Lagrange equation gives

$$\frac{d}{dt}(\frac{\partial KE}{\partial \dot{q}_i}) - \frac{\partial KE}{\partial q_i} + \frac{\partial PE}{\partial q_i} = \begin{cases} v_i & q_i \in \theta \\ 0 & q_i \in \beta \end{cases} \tag{3.13}$$

where q_i is the ith element of the vector q defined in Equation 3.12, and v_i is the generalized actuator force at the ith joint. In the above dynamic equations, it is assumed that no external force is applied on the system. Now, some remarks regarding the above derivation are given here. In Figure 3.1, the $(j+1)$th link is connected to the end of the deformed jth link, so coordinate transformation of the $(j+1)$th frame should be with respect to the distal end frame $\{\hat{x}'_j, \hat{y}'_j, \hat{z}'_j\}$ instead of to the rigid link frame. Since change of $\{\hat{x}'_j, \hat{y}'_j, \hat{z}'_j\}$ orientations are due to link twistings $\alpha_x, \alpha_y, \alpha_z$ at $r = L$, so

$$\begin{aligned} T'_j &= [\hat{x}'_j\ \hat{y}'_j\ \hat{z}'_j] = [\hat{x}_j, \hat{y}_j, \hat{z}_j]W_j(L) \\ &= T_j W_j(L) \end{aligned} \tag{3.14}$$

where $W_j(L)$ is a 3×3 transformation matrix needed to be defined. To find $W_j(L)$, the rotational matrices of $\alpha_x, \alpha_y, \alpha_z$ at $r = L$ are derived first. For small deformations, it can be shown that

$$\begin{aligned} \mathrm{Rot}(\hat{x}, \alpha_x) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_x(L) & -\sin\alpha_x(L) \\ 0 & \sin\alpha_x(L) & \cos\alpha_x(L) \end{bmatrix} \approx \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -\alpha_x(L) \\ 0 & \alpha_x(L) & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\alpha_x(L) \\ 0 & \alpha_x(L) & 0 \end{bmatrix} \\ &\stackrel{\text{def}}{=} \mathcal{I} + \Delta_x \end{aligned} \tag{3.15}$$

$$\begin{aligned} \mathrm{Rot}(\hat{y}, \alpha_y) &= \begin{bmatrix} \cos\alpha_y(L) & 0 & \sin\alpha_y(L) \\ 0 & 1 & 0 \\ -\sin\alpha_y(L) & 0 & \cos\alpha_y(L) \end{bmatrix} \approx \begin{bmatrix} 1 & 0 & \alpha_y(L) \\ 0 & 1 & 0 \\ -\alpha_y(L) & 0 & 1 \end{bmatrix} \\ &\stackrel{\text{def}}{=} \mathcal{I} + \Delta_y \end{aligned} \tag{3.16}$$

$$\begin{aligned} \mathrm{Rot}(\hat{z},\alpha_z) &= \begin{bmatrix} \cos\alpha_z(L) & -\sin\alpha_z(L) & 0 \\ \sin\alpha_z(L) & \cos\alpha_z(L) & 0 \\ 0 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} 1 & -\alpha_z(L) & 0 \\ \alpha_z(L) & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &\stackrel{\text{def}}{=} \mathcal{I} + \Delta_z \end{aligned} \tag{3.17}$$

where $\mathcal{I}$ is a 3×3 identity matrix. By neglecting the high order terms, a multiplication of the above rotational matrices produces

$$(\mathcal{I} + \Delta_x)(\mathcal{I} + \Delta_y)(\mathcal{I} + \Delta_z) \approx \mathcal{I} + \Delta_x + \Delta_y + \Delta_z \tag{3.18}$$

Since the order of multiplication does not affect the above result, W_j is given by

$$W_j(L) = \mathcal{I} + \Delta_x + \Delta_y + \Delta_z = \begin{bmatrix} 1 & -\alpha_z(L) & \alpha_y(L) \\ \alpha_z(L) & 1 & -\alpha_x(L) \\ -\alpha_y(L) & \alpha_x(L) & 1 \end{bmatrix} \tag{3.19}$$

A substitution of the above W_j into Equation 3.14 finishes the derivation of T'_j. Notice that the orthogonality property of the transformation matrix holds for T'_j, i.e., $(T'_j)^{-1} = T'^T_j$. After establishing local frame for each compliant link, the position vector of each local frame vertex could be generated recursively. The procedure of finding the local frame position of a compliant link is similar to that of rigid link except that the distal end-point deflections are added to each rigid link length. The jth link origin O_j in Equation 3.2 could be computed by this procedure. Another comment is that the G and H function definitions are also applicable to compliant system dynamics. For example, in Figure 3.1, the velocities and accelerations of p' could be expressed by

$$\begin{aligned} \dot{P}' &= [\frac{\partial P'}{\partial q_1} \frac{\partial P'}{\partial q_2} \cdots]\dot{q} \\ &= {}_T G_{p'}\dot{q} \\ \ddot{P}' &= {}_T G_{p'}\ddot{q} + \dot{q}^T\ {}_T H_{p'}\dot{q} \\ \omega_j &= {}_R G_{p'}\dot{q} \\ \dot{\omega}_j &= {}_R G_{p'}\ddot{q} + \dot{q}^T\ {}_R H_{p'}\dot{q} \end{aligned} \tag{3.20}$$

where link deflections and twistings and joint deformations are considered in forming the G and H functions. With these compact notations, the kinetic energy formulation in Equation 3.6 could be expressed by

$$\begin{aligned} 2KE_j &= \int \dot{P}'^T \dot{P}' dm + \int \{\omega_j^T T_j (dI) T^T \omega_j\} \\ &= \dot{q}^T (\int {}_T G_{p'}^T \, {}_T G_{p'} dm + \int {}_R G_{p'}^T T_j (dI) T_j^T \, {}_R G_{p'}) \dot{q} \end{aligned} \tag{3.21}$$

and the total kinetic energy is

$$\begin{aligned} 2KE &= \dot{q}^T \sum_{j=1}^{N} (\int {}_T G_{p'}^T \, {}_T G_{p'} dm + \int {}_R G_{p'}^T T_j (dI) T_j^T \, {}_R G_{p'}) \dot{q} \\ &\stackrel{\text{def}}{=} \dot{q}^T I^* \dot{q} \end{aligned} \tag{3.22}$$

where I^* is the generalized inertia matrix which is generally symmetric and positive definite. Another important result of assumed mode method is that resilient energy could be defined by constant stiffness matrices. For example, in Equation 3.7, the resilient energy of α_y is given by

$$\frac{1}{2} \int E I_y (\frac{\partial \alpha_y}{\partial r})^2 dr \tag{3.23}$$

since the assumed mode expression of δ_z in Equation 3.10 could be written as

$$\begin{aligned} \delta_z &= \sum_{i=1}^{nz} \beta_{jzi}(t) \phi_{jzi}(r) \\ &= \beta_{jz}^T(t) \phi_{jz}(r) \end{aligned} \tag{3.24}$$

with β_{hz} and ϕ_{jz} defined by

$$\begin{aligned} \beta_{jz}(t) &= [\beta_{jz1}(t) \cdots \beta_{jznz}(t)]^T \in \mathcal{R}^{nz} \\ \phi_{jz}(r) &= [\phi_{jz1}(r) \cdots \phi_{jznz}(r)]^T \in \mathcal{R}^{nz} \end{aligned} \tag{3.25}$$

then the α_y in Equation 3.8 has an inner product form of

$$\alpha_y = -\beta_{jz}^T (\frac{\partial \phi_{jz}}{\partial r}) \tag{3.26}$$

A substitution of the above α_y into resilient energy expression in Equation 3.23 produces

$$\begin{aligned} \frac{1}{2}\int EI_y(\frac{\partial \alpha_y}{\partial r})^2 dr &= \frac{1}{2}\beta_{jz}^T[\int EI_y(\frac{\partial^2 \phi_{jz}}{\partial r^2})(\frac{\partial^2 \phi_{jz}}{\partial r^2})^T dr]\beta_{jz} \\ &\stackrel{\text{def}}{=} \frac{1}{2}\beta_{jz}^T K_{jz}\beta_{jz} \end{aligned} \tag{3.27}$$

in which

$$K_{jz} = \int EI_y(\frac{\partial^2 \phi_{jz}}{\partial r^2})(\frac{\partial^2 \phi_{jz}}{\partial r^2})^T dr \in \mathcal{R}^{nz \times nz} \tag{3.28}$$

is a constant stiffness matrix after integration. Similarly, the other resilient energy in Equation 3.7 could be defined by

$$\begin{aligned} \frac{1}{2}\int EI_z(\frac{\partial \alpha_z}{\partial r})^2 dr &= \frac{1}{2}\beta_{jy}^T K_{jy}\beta_{jy} \\ \frac{1}{2}\int EA(\frac{\partial \delta_x}{\partial r})^2 dr &= \frac{1}{2}\beta_{jx}^T K_{jx}\beta_{jx} \\ \frac{1}{2}\int GJ_x(\frac{\partial \alpha_x}{\partial r})^2 dr &= \frac{1}{2}\beta_{j\alpha}^T K_{j\alpha}\beta_{j\alpha} \end{aligned} \tag{3.29}$$

In the above equations, the vibratory amplitude vector and stiffness matrix dimensions are defined accordingly by the number of used assumed modes. With these constant stiffness matrices, it is possible to express the total potential energy in Equation 3.11 by

$$PE = \sum_{j=1}^{N}(\int P'^T \hat{g} dr) + \frac{1}{2}\beta^T K \beta \tag{3.30}$$

where

$$K = \text{diag}[K_{10}\ K_{1x}\ K_{1y}\ K_{1z}\ K_{1\alpha}\ K_{20}\ K_{2x}\ \cdots] \tag{3.31}$$

and vector β is defined in Equation 3.12. Let n_β denote the total number of assumed modes, and n_θ be the number of nominal joints, then from Equation 3.12, the dimensions of vectors θ and β are given which are $\theta \in \mathcal{R}^{n_\theta}$ and $\beta \in \mathcal{R}^{n_\beta}$. By the kinetic energy expression in Equation 3.22 and the potential

energy formulation in Equation 3.30, the system dynamics could be defined by

$$\begin{aligned} \frac{d}{dt}(\frac{\partial KE}{\partial \dot{q}}) - \frac{\partial KE}{\partial q} + \frac{\partial PE}{\partial q} &= I^*\ddot{q} + \begin{bmatrix} f_1' \\ f_2 \end{bmatrix} + \begin{bmatrix} 0 \\ K\beta \end{bmatrix} \\ &= \begin{bmatrix} v \\ 0 \end{bmatrix} \end{aligned} \tag{3.32}$$

where $v \in \mathcal{R}^{n_\theta}$ is the vector of generalized actuator forces, and f_1' and f_2 are the coupling force terms defined by

$$\begin{bmatrix} f_1' \\ f_2 \end{bmatrix} = \frac{d}{dt}(I^*)\dot{q} - \frac{\partial}{\partial q}(\frac{1}{2}\dot{q}^T I^* \dot{q}) + \frac{\partial}{\partial q}[\sum_{j=1}^{N}(\int P'^T \hat{g} dr)] \tag{3.33}$$

both f_1' and f_2 are nonlinear functions of θ, $\dot{\theta}$, β, and $\dot{\beta}$, and their dimensions are: $f_1' \in \mathcal{R}^{n_\theta}$ and $f_2 \in \mathcal{R}^{n_\beta}$. For a simple system, a symbolic program like MACSYMA is generally suggested to perform analytical integrations of the above kinetic and potential energy expressions.

3.2 Lumped Parameter Model of Compliant Manipulators

Theoretically link deflections are composed of infinite modes, but resilient energy actually concentrates in few primary modes. Also, due to the damping effect and the demand of large energy to bend a link into high order mode shapes, high frequency modes are seldom excited in regular operation. Based on these facts, usually only primary modes are considered in the study of continuous link vibration. Furthermore, as the rigidity of a compliant link increases, the link deflection reduces and the number of dominant modes decreases. Therefore for small deflection, a simple but efficient method of studying link deformation is to focus on the largest contributor: the first mode. This first mode approximation method is called the lumped parameter

model. In the rigid body model, system inertia is modeled by a point mass. The same approach is extended to the lumped parameter model except that lumped springs are added to the compliant system to simulate structural deformation. These springs are imaginary and occupy no physical space or weight. They are defined by their locations, motion nature (translational or torsional), and stiffness values. In the lumped parameter model, each spring represents one degree of freedom, and a generalized coordinate is assigned to each vibratory displacement. It should be noticed that to preserve physical reality, the lumped spring stiffnesses are generally obtained experimentally such as reported by [Behi, 1985] and [Sklar, 1988].

Figure 3.3 sketches the lumped spring model of a compliant link. In that figure, the jth link is connected to its preceding link by a compliant revolute joint which is modeled by a torsional spring with vibratory amplitude β_{j0}. In case that the jth joint is prismatic then joint compliance is replaced by a translational spring. The jth link is assumed rigid and the actual link compliances are undertaken by three translational and three torsional springs attached to the distal end of the link. The six springs are assumed decoupled. The translational springs duplicate the end-point deflection of the link along each local coordinate direction whose amplitudes are given by $\{\beta_{j1}, \beta_{j2}, \beta_{j3}\}$. The torsional springs produce the distal-end twistings of the link in three orthogonal directions which are defined by $\{\beta_{j4}, \beta_{j5}, \beta_{j6}\}$. Therefore in the lumped parameter model, each link deformations could be defined by seven vibratory coordinates which are

$$\{\beta_{j0}, \beta_{j1}, \beta_{j2}, \beta_{j3}, \beta_{j4}, \beta_{j5}, \beta_{j6}\}$$

and plus the actual joint displacement there are eight degrees of freedom to each compliant link. Hence, an n_θ-link compliant manipulator could be

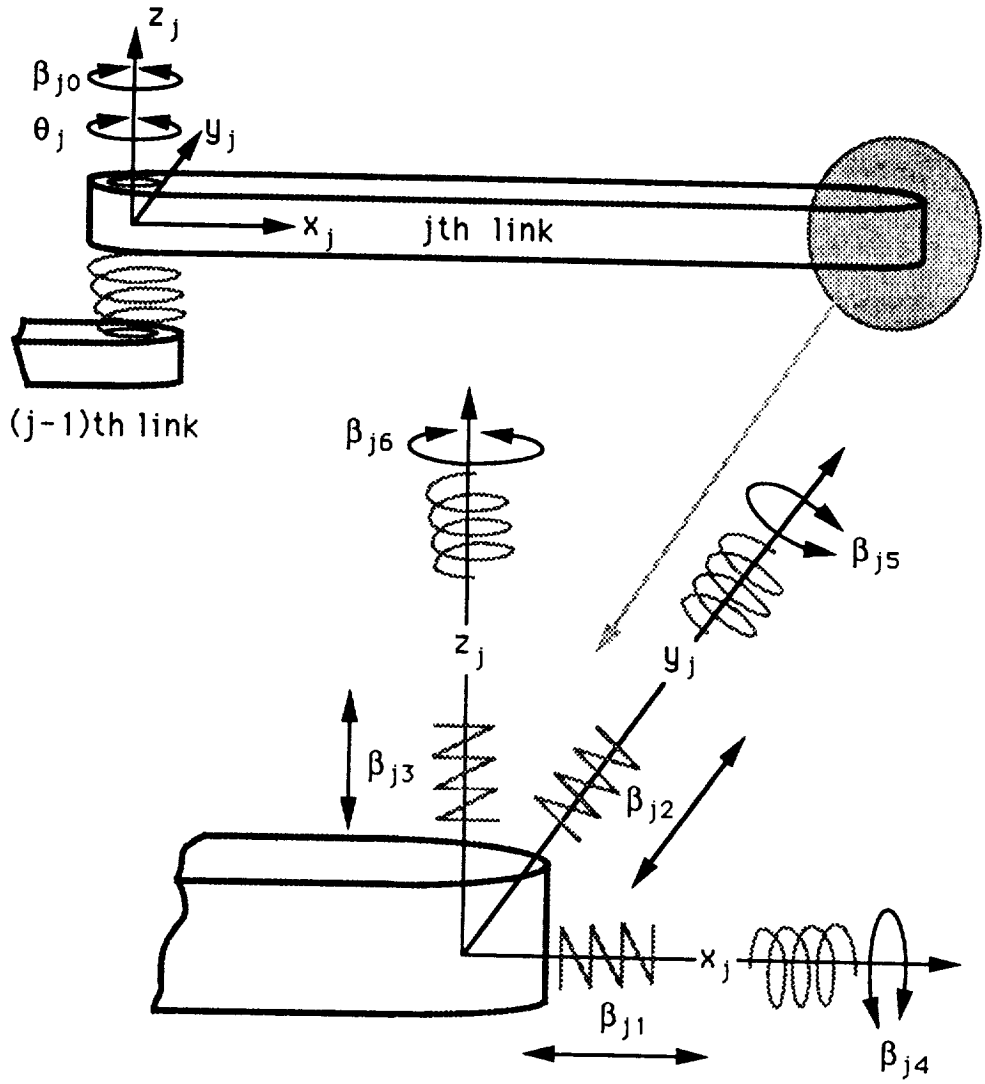


Figure 3.3: Lumped Parameter Model of a Compliant Link

modeled by $8n_\theta$ degrees of freedom in the lumped parameter model.

In the lumped parameter method, the modeled springs are assumed decoupled, hence the pseudo-joint concept could be introduced to define the vibratory motion. In the pseudo-joint technique, a modeled spring is treated like an actuator of an imaginary joint, and the resultant deflection is duplicated by the movement of that imaginary joint. For example, a translational deflection is replaced by a pseudo prismatic joint, and a torsional deformation is modeled by a pseudo revolute joint. These pseudo joints are driven by the force or torque stored in the deformed springs. After replacing all springs by pseudo joints, a compliant manipulator kinematically behaves like a rigid manipulator, and the rigid model dynamics derived before could be extended directly to the lumped parameter system.

As a demonstration, the lumped parameter model of the three-link arm shown in Figure 2.3 is given below. Figure 3.4 shows a breakdown of the three-link arm. Deformation of each link is approximated by the fundamental mode in each direction. For example, the first link motion is defined by the nominal joint displacement θ_1, joint deflection β_1, three linear deflections $\{\beta_2, \beta_3, \beta_4\}$ at the distal end, and three end-point twistings $\{\beta_5, \beta_6, \beta_7\}$. The vibratory directions of the above amplitudes are indicated in the figure. In this example case, there are eight degrees of freedom in each link, hence totally twenty-four generalized coordinates are used to define the arm motion. In order to describe the arm motion, the transformation matrices are computed first to define the local coordinate frames of the three-link arm, which are

$$
\begin{aligned}
T_1 &= [\hat{x}_1\ \hat{y}_1\ \hat{z}_1] \\
&= \begin{bmatrix} \cos(\theta_1+\beta_1) & -\sin(\theta_1+\beta_1) & 0 \\ \sin(\theta_1+\beta_1) & \cos(\theta_1+\beta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

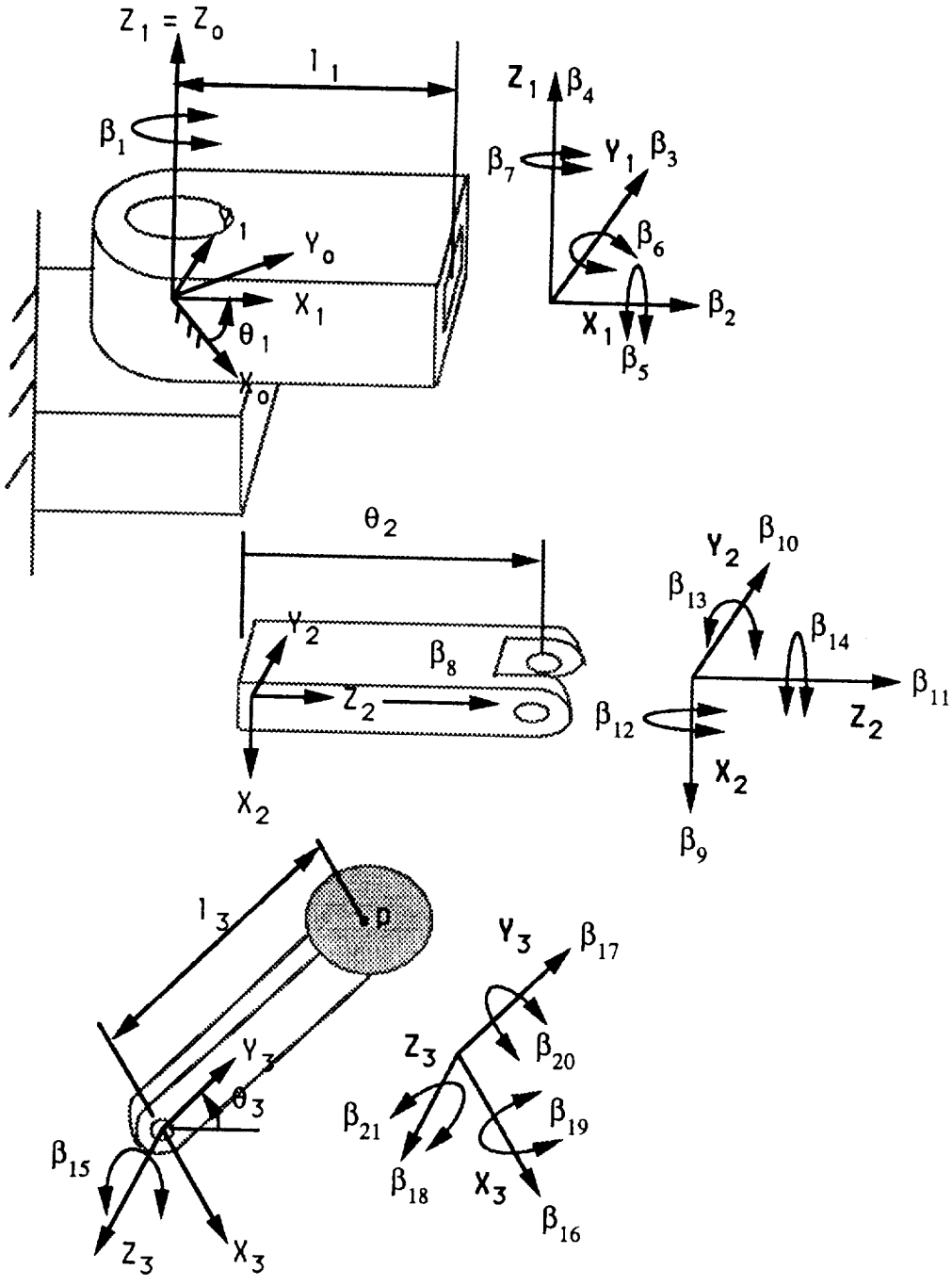


Figure 3.4: Lumped Parameter Model of a Three-Link Arm

$$
\begin{aligned}
T_2 &= [\hat{x}_2\ \hat{y}_2\ \hat{z}_2] \\
&= T_1 \begin{bmatrix} 1 & -\beta_7 & \beta_6 \\ \beta_7 & 1 & -\beta_5 \\ -\beta_6 & \beta_5 & 1 \end{bmatrix} \begin{bmatrix} \cos 90 & 0 & \sin 90 \\ 0 & 1 & 0 \\ -\sin 90 & 0 & \cos 90 \end{bmatrix} \\
T_3 &= [\hat{x}_3\ \hat{y}_3\ \hat{z}_3] \\
&= T_2 \begin{bmatrix} 1 & -\beta_{14} & \beta_{13} \\ \beta_{14} & 1 & -\beta_{12} \\ -\beta_{13} & \beta_{12} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 90 & -\sin 90 \\ 0 & \sin 90 & \cos 90 \end{bmatrix} \\
&\quad \begin{bmatrix} \cos(\theta_3+\beta_{15}) & -\sin(\theta_3+\beta_{15}) & 0 \\ \sin(\theta_3+\beta_{15}) & \cos(\theta_3+\beta_{15}) & 0 \\ 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{3.34}
$$

where small twisting angle assumption and Equation 3.19 results are applied in the above equations. As an example, the linear and angular velocities of the point p in Figure 3.4 are derived in terms of the G functions. The positional vector of p is given by

$$P = P_{12} + P_{23} + P_{3p} \in \mathcal{R}^3 \tag{3.35}$$

where P_{12} is the positional vector from the origin of the first frame to that of the second frame, similar definition is used on P_{23}, and P_{3p} is the positional vector of p with respect to the third frame. All vectors are defined in the fixed global frame $\{X_o, Y_o, Z_o\}$. Details of the above positional vectors are shown in the following equations:

$$
\begin{aligned}
P_{12} &= l_1\hat{x}_1 + \beta_2\hat{x}_1 + \beta_3\hat{y}_1 + \beta_4\hat{z}_1 \\
P_{23} &= \theta_2\hat{z}_2 + \beta_8\hat{z}_2 + \beta_9\hat{x}_2 + \beta_{10}\hat{y}_2 + \beta_{11}\hat{z}_2 \\
P_{3p} &= l_3\hat{y}_3 + \beta_{16}\hat{x}_3 + \beta_{17}\hat{y}_3 + \beta_{18}\hat{z}_3
\end{aligned} \tag{3.36}
$$

By the pseudo-joint technique, the arm is defined by the generalized coordinates

$$q = [\theta_1\ \theta_2\ \theta_3\ \beta_1\ \beta_2\ \cdots\ \beta_{21}]^T \in \mathcal{R}^{24}$$

Now, each link of the arm is assumed rigid and is connected by decoupled actual and pseudo joints. This model allows us to define point p velocities by using G functions, which are

$$\begin{aligned} \dot{P} &= {}_{T}G_{p}\dot{q} \\ \dot{\Phi} &= {}_{R}G_{p}\dot{q} \end{aligned} \tag{3.37}$$

where $\dot{\Phi}$ is angular velocity of p, and the 3×24 matrices, ${}_{T}G_{p}$ and ${}_{R}G_{p}$, are the translational and rotational G functions respectively whose elements are given as follows

$${}_{T}G_{p} = [\frac{\partial p}{\partial \theta_1} \; \frac{\partial p}{\partial \theta_2} \; \cdots \; \frac{\partial p}{\partial \beta_{21}}] \tag{3.38}$$

with

$$\begin{array}{llll} \frac{\partial P}{\partial \theta_1} = \hat{z}_1 \times P & \frac{\partial P}{\partial \theta_2} = \hat{z}_2 & \frac{\partial P}{\partial \theta_3} = \hat{z}_3 \times P_{3p} & \frac{\partial P}{\partial \beta_1} = \hat{z}_1 \times P \\ \frac{\partial P}{\partial \beta_2} = \hat{x}_1 & \frac{\partial P}{\partial \beta_3} = \hat{y}_1 & \frac{\partial P}{\partial \beta_4} = \hat{z}_1 & \frac{\partial P}{\partial \beta_5} = \hat{x}_1 \times P_{23} \\ \frac{\partial P}{\partial \beta_6} = \hat{y}_1 \times P_{23} & \frac{\partial P}{\partial \beta_7} = \hat{z}_1 \times P_{23} & \frac{\partial P}{\partial \beta_8} = \hat{z}_2 & \frac{\partial P}{\partial \beta_9} = \hat{x}_2 \\ \frac{\partial P}{\partial \beta_{10}} = \hat{y}_2 & \frac{\partial P}{\partial \beta_{11}} = \hat{z}_2 & \frac{\partial P}{\partial \beta_{12}} = \hat{x}_2 \times P_{3p} & \frac{\partial P}{\partial \beta_{13}} = \hat{y}_2 \times P_{3p} \\ \frac{\partial P}{\partial \beta_{14}} = \hat{z}_2 \times P_{3p} & \frac{\partial P}{\partial \beta_{15}} = \hat{z}_3 \times P_{3p} & \frac{\partial P}{\partial \beta_{16}} = \hat{x}_3 & \frac{\partial P}{\partial \beta_{17}} = \hat{y}_3 \\ \frac{\partial P}{\partial \beta_{18}} = \hat{z}_3 & \frac{\partial P}{\partial \beta_{19}} = 0 & \frac{\partial P}{\partial \beta_{20}} = 0 & \frac{\partial P}{\partial \beta_{21}} = 0 \end{array} \tag{3.39}$$

and

$${}_{R}G_{p} = [\hat{z}_1 \; 0 \hat{z}_3 \; \hat{z}_1 \; 0 \; 0 \; 0 \hat{x}_1 \; \hat{y}_1 \; \hat{z}_1 0 \; 0 \; 0 \; 0 \hat{x}_2 \; \hat{y}_2 \; \hat{z}_2 \; \hat{z}_3 \; 0 \; 0 \; 0 \hat{x}_3 \; \hat{y}_3 \; \hat{z}_3] \tag{3.40}$$

The above results are created by a direct application of the G function definitions listed in Tables 2.1 and 2.2. Similarly, accelerations of p are

$$\begin{aligned} \ddot{P} &= {}_{T}G_{p}\ddot{q} + \dot{q}^{T}\,{}_{T}H_{p}\dot{q} \\ \ddot{\Phi} &= {}_{R}G_{p}\ddot{q} + \dot{q}^{T}\,{}_{R}H_{p}\dot{q} \end{aligned} \tag{3.41}$$

where ${}_{T}H_{p}$ and ${}_{R}H_{p} \in \mathcal{R}^{3\times 24\times 24}$ are the associated H functions. The above H functions could be derived immediately from Tables 2.3 and 2.4. However, due to the prohibitively large dimension, they are not shown in this report. The

above results are applicable to any point in the arm, so the lumped parameter model possesses the efficient nature of rigid body model. Therefore for an n_θ-link compliant manipulator modeled by the lumped parameter technique, the system kinetic energy is

$$\begin{aligned} KE &= \frac{1}{2}\sum_{j=1}^{N}(m_j\dot{q}^T\ {}_TG_j^T\ {}_TG_j\dot{q} + \dot{q}^T\ {}_RG_j^T T_j I_j T_j^T\ {}_RG_j\dot{q}) \\ &\stackrel{\text{def}}{=} \frac{1}{2}\dot{q}^T I^*(\theta,\beta)\dot{q} \end{aligned} \tag{3.42}$$

where

$$I^*(\theta,\beta) = \sum_{j=1}^{N}(m_j\ {}_TG_j^T\ {}_TG_j + \ {}_RG_j^T T_j I_j T_j^T\ {}_RG_j) \in \mathcal{R}^{8n_\theta \times 8n_\theta} \tag{3.43}$$

is the generalized inertia matrix which still maintains the symmetric and positive definite character. In the above equations, q is a generalized coordinate vector composed by two parts: $q = [\theta^T, \beta^T]^T$, where $\theta \in \mathcal{R}^{n_\theta}$ is the vector of all nominal joint variables, and $\beta \in \mathcal{R}^{n_\beta}$ contains all pseudo joint displacements with $n_\beta = 7n_\theta$. Also, m_j is mass of the jth link, I_j is the associated moment of inertia, and N is the total number of links including payload. Following the above notations, the system potential energy is

$$PE = \sum_{j=1}^{N} m_j P_j^T \hat{g} + \frac{1}{2}\beta^T K\beta \tag{3.44}$$

where K is a diagonal stiffness matrix whose diagonal values correspond to the modeled spring stiffnesses. The Lagrange equation is used directly to formulate the system dynamics which are

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{q}_i}\right) - \frac{\partial KE}{\partial q_i} + \frac{\partial PE}{\partial q_i} = \begin{cases} v_i & q_i \in \theta \\ 0 & q_i \in \beta \end{cases} \tag{3.45}$$

Then by using the approach introduced in Chapter 2 and the lumped parameter model approximation, the dynamic equations of a compliant manipulator

are given by

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{q}}\right) - \frac{\partial KE}{\partial q} + \frac{\partial PE}{\partial q} \\
&= I^*\ddot{q} + \sum_{j=1}^{N} m_j \, {}_T G_j^T ({}_T \dot{q}^T \, {}_T H_j \dot{q}) + \sum_{j=1}^{N} {}_R G_j^T [\dot{\Phi}_j \times (T_j I_j T_j^T \, {}_R G_j \dot{q})] \\
&\quad + \sum_{j=1}^{N} {}_R G_j^T [(T_j I_j T_j^T)({}_R \dot{q}^T \, {}_R H_j \dot{q})] + \sum_{j=1}^{N} m_j \, {}_T G_j^T \hat{g} + \begin{bmatrix} 0 \\ K\beta \end{bmatrix} \\
&\stackrel{\text{def}}{=} I^*\ddot{q} + \begin{bmatrix} f_1' \\ f_2 \end{bmatrix} + \begin{bmatrix} 0 \\ K\beta \end{bmatrix} \\
&= \begin{bmatrix} v \\ 0 \end{bmatrix} \in \mathcal{R}^{n_\theta + n_\beta}
\end{aligned} \tag{3.46}$$

where $f_1' \in \mathcal{R}^{n_\theta}$ and $f_2 \in \mathcal{R}^{n_\beta}$ contain the summation terms in the above equations, and both are nonlinear functions of θ, $\dot{\theta}$, β, and $\dot{\beta}$. In the above equation, v is an n_θ vector of generalized actuator forces.

3.3 Compliant Manipulator Dynamics Including Actuator Parameters

In both distributed and lumped parameter models, the compliant manipulator dynamics have the common symbolic form of

$$I^*\ddot{q} + \begin{bmatrix} f_1' \\ f_2 \end{bmatrix} + \begin{bmatrix} 0 \\ K\beta \end{bmatrix} = \begin{bmatrix} v \\ 0 \end{bmatrix} \tag{3.47}$$

where q is the vector of generalized coordinates composed of nominal joint displacements θ and vibratory deflections β, I^* represents system generalized inertia and is a function of θ and β, f_1' and f_2 contain the nonlinear Coriolis, centrifugal, and gravitational forces, $K\beta$ is the vector of spring forces, and v is the vector of generalized actuator forces. To complete the system dynamics, actuator dynamics is added to the above dynamic equations in this section. First, since $\theta \in \mathcal{R}^{n_\theta}$, $\beta \in \mathcal{R}^{n_\beta}$ and $I^* = I^{*T}$, I^* could be divided into following

submatrices

$$I^* = \begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \in \mathcal{R}^{(n_\theta+n_\beta)\times(n_\theta+n_\beta)} \tag{3.48}$$

where $\Lambda_1' \in \mathcal{R}^{n_\theta \times n_\theta}$, $\Sigma \in \mathcal{R}^{n_\beta \times n_\theta}$, and $\Lambda_2 \in \mathcal{R}^{n_\beta \times n_\beta}$. Then Equation 3.47 becomes

$$\begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1' \\ f_2 \end{bmatrix} + \begin{bmatrix} 0 \\ K\beta \end{bmatrix} = \begin{bmatrix} v \\ 0 \end{bmatrix} \tag{3.49}$$

where the first part, i.e.,

$$\Lambda_1'\ddot{\theta} + \Sigma^T\ddot{\beta} + f_1' = v \tag{3.50}$$

represents the dynamic balance between actuating forces and system response, and the second part

$$\Sigma\ddot{\theta} + \Lambda_2\ddot{\beta} + f_2 + K\beta = 0 \tag{3.51}$$

shows the interaction between nominal and oscillatory motions through spring forces. It is assumed that the compliant manipulator is driven by DC servo motors, and the actuator dynamics of the jth joint DC motor are

$$L_j\dot{i}_j + R_j i_j + K_{vj}N_j\dot{\theta}_j = u_j' \tag{3.52}$$

$$N_j J_j\ddot{\theta}_j + N_j D_j\dot{\theta}_j + N_j S_j\theta_j + v_j/N_j = K_{tj}i_j \tag{3.53}$$

where i_j is the armature current, L_j is the circuit inductance, R_j is the circuit resistance, K_{vj} is the actuator voltage gain, u_j' is the voltage input, J_i is the armature inertia, D_j is the actuator damping, S_j is the motor shaft stiffness, K_{tj} is the actuator torque gain, v_j is the external load, and N_j is the gear reduction ratio. The subscript j indicates the parameter of the jth joint actuator. These actuator parameters except v_j are constant values. Generally, the inductance L_j is negligible, so Equations 3.52 and 3.53 could be combined into a second order form

$$N_j^2 J_j\ddot{\theta}_j + \left(N_j^2 D_j + \frac{N_j^2 K_{vj}K_{tj}}{R_j}\right)\dot{\theta}_j + N_j^2 S_j\theta_j + v_j = \frac{K_{tj}N_j}{R_j}u_j' \tag{3.54}$$

Then for an n_θ-link manipulator, the actuator dynamics can be grouped into

$$J\ddot{\theta} + D\dot{\theta} + K_c\theta + v = K_t u' \tag{3.55}$$

where

$$\begin{aligned}
J &= \text{diag}[(N_1^2 J_1)\,(N_2^2 J_2)\,\cdots\,(N_{n_\theta}^2 J_{n_\theta})] \\
D &= \text{diag}[(N_1^2 D_1 + \frac{N_1^2 K_{v1} K_{t1}}{R_1})\,\cdots\,(N_{n_\theta}^2 D_{n_\theta} + \frac{N_{n_\theta}^2 K_{vn_\theta} K_{tn_\theta}}{R_{n_\theta}})] \\
K_c &= \text{diag}[(N_1^2 S_1)\,(N_2^2 S_2)\,\cdots\,(N_{n_\theta}^2 S_{n_\theta})] \\
K_t &= \text{diag}[(\frac{K_{t1} N_1}{R_1})\,(\frac{K_{t2} N_2}{R_2})\,\cdots\,(\frac{K_{tn_\theta} N_{n_\theta}}{R_{n_\theta}})]
\end{aligned} \tag{3.56}$$

and J, D, K_c, and K_t are $n_\theta \times n_\theta$ constant diagonal matrices. Also, u' is the vector of input voltages with u'_j as the jth element. By substituting Equation 3.50 into Equation 3.55, the combined dynamic equations become

$$(\Lambda'_1 + J)\ddot{\theta} + \Sigma^T \ddot{\beta} + f'_1 + D\dot{\theta} + K_c\theta = K_t u' \tag{3.57}$$

the above equations have actuator voltages as the control inputs. By defining

$$\Lambda_1 = \Lambda'_1 + J\ ;\ f_1 = f'_1 + D\dot{\theta} + K_c\theta\ ;\ u = K_t u' \tag{3.58}$$

the final system dynamic equations are

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 + K\beta \end{bmatrix} = \begin{bmatrix} u \\ 0 \end{bmatrix} \tag{3.59}$$

In the later controller designs, control algorithm will be built based on the above symbolic dynamic equations which are general for distributed and lumped parameter models. The designed input u will be divided by K_t to obtain the actual input voltage vector u'.

3.4 Comparison Between Distributed and Lumped Parameter Models

In the previous sections, we have presented the model and dynamic equations for compliant manipulators by both distributed and lumped parameter methods. By comparing the assumption and techniques used in both approaches, each one possesses its unique characters that are not shared by the other. Obviously, the advantage of distributed parameter model is its precision. Since the distributed parameter model defines vibratory motion based on the dynamic balance of each differential segment along a compliant link. Fidelity is the strong point of this method. The disadvantage of the distributed parameter model is computational inefficiency. Although the resilient energy of a continuous link is contained primarily in few basic modes, and finite number of assumed modes are often applied to simplify the dimensions and integrations of a given distributed model, the final system dimensions are still beyond the real-time computation capacity of general processors. Therefore, the distributed parameter model is used mainly in off-line design study.

By contrast, the lumped parameter technique models link deflections by the first mode approximation and neglects high order mode effects, so exactness is not the focus of this method. But due to its smaller dimension, the lumped parameter method is a candidate for real-time control model. Of course, selection of a compliant manipulator model still relies on the nature of a given compliant manipulator and conditions of application. Several factors affecting selection of model are listed as follows: (1) dimension of compliant link, (2) material of compliant link, (3) robot operation speed, (4) payload size and external force, and (5) operation precision requirement. The first

two factors define link rigidity, the third and fourth factors determine the load supported by the link, and the last factor decides the degree of accuracy needed in a compliant link model. For example, according to [Ussher and Doetsch, 1983] and [Taylor, 1985], Space Shuttle Remote Manipulator System (RMS) has two tubular *booms* (links) of diameter 13.5 in and length 23 ft and 20.9 ft separately, and the booms are made from graphite-epoxy composites. Due to the lightweight nature, the stiffness of RMS at its fully extended position is 7.5 lb/in, plus the maximum payload mass is 65,000 lb, apparently high order mode effects are important in analyzing link deformation. However, RMS operates in a zero-g environment, so the inertial load is caused mainly by the motion of the payload. Therefore, when RMS operates in a low speed, the inertial load is very small in which case the first mode dominates link deflection, then the lumped parameter model is considered an efficient approach. Additionally, RMS is so light that its inertia is almost negligible in comparison with payload and Orbiter inertia. Hence, [Book, 1979] and [Sellhorst, 1982] treat RMS as massless chain and model its compliances by lumped springs. This example points out that selection of compliant manipulator model is basically oriented by a given task. The tradeoff between accuracy and computation effort should be decided by user based on the final purpose of application. Currently, the distributed parameter model is used in off-line design or on-line motion control of simple structures like one-link arms, e.g., [Hasting and Book, 1986], and the lumped parameters model is used in on-line control study of high degree-of-freedom robotic manipulators as reported by [Hernandez, 1989] and [Lin, Tosunoglu, and Tesar, 1990, b].

Chapter 4

Dynamic Property Investigation of Compliant Manipulators

In the last chapter, we derived compliant manipulator dynamic equations by both distributed and lumped parameter models. Unlike rigid body dynamics, compliant dynamics have additional n_β equations describing the vibratory system behavior. These additional equations carry some properties that are distinct from rigid body dynamics. Four major aspects of these properties are investigated here: reduction to rigid system dynamics, the system natural frequency, accessibility of vibratory mode, and controllability of vibratory mode.

4.1 Reduction to Rigid System Dynamics

The compliant manipulator system dynamics are derived in Equation 3.59 which are common for both distributed and lumped parameter models. This section will study the physical meaning of vibratory dynamics when the structure becomes infinitely rigid. First the dynamic equations in Equation 3.59 are repeated here, which are

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 + K\beta \end{bmatrix} = \begin{bmatrix} u \\ 0 \end{bmatrix} \in \mathcal{R}^{n_\theta + n_\beta} \tag{4.1}$$

where the upper part represents the dynamic equations of nominal joints, and the lower part describes the vibratory mode dynamics. When manipula-

tor rigidity increases and structural deflection becomes negligible, the vibratory amplitudes vanish from the dynamic expressions, and the upper part of Equation 4.1 reduces to the rigid body dynamic expression. Consequently, the vibratory dynamics which occupy the lower part of Equation 4.1 take on a different physical meaning. Recalling that in deriving the rigid body dynamics in the second chapter, we have used the Newton-Euler approach to show that the dynamic equation at each joint is actually the projection of all inertial loads on that particular joint, and that the actuator force is the counteracting force to the projected inertial load. Following that relation, the final system dynamic equations balance joint actuating forces to system inertial load. However, in doing so, it is assumed that each link is of infinite strength to support and transmit these forces, therefore, structural internal reaction forces are neglected because they are cancelled out in the final dynamic expressions. In a rigid link system, the spring force $K\beta$ is exactly the internal reaction force that is generally neglected in rigid body dynamics. This result could be explained from pseudo-joint point of view. In a lumped parameter model, each link has six decoupled pseudo joints located at the distal end: three orthogonal prismatic joints and three orthogonal revolute joints. Then the lower part of Equation 4.1 indicates that $K\beta$ is the force counteracting inertial projection at each pseudo joint, or direction, hence when link rigidity becomes infinitely large, $K\beta$ becomes the internal reaction force or torque at the distal end of a given link. Therefore, in the rigid link case $K\beta$ represents internal structural forces, and the lower part of Equation 4.1 represents the action and reaction balance at each pseudo joint location and direction. It should be noted that although internal forces usually do not show up in the final system dynamics of a rigid body model, it is important to analyze these forces for structural strength design.

4.2 Structural Natural Frequency

Another important information observed in the vibratory dynamic equations is the structural system natural frequencies. For a stationary manipulator, the actual joints are motionless, that is, $\dot{\theta} = \ddot{\theta} = 0$. Then a sudden blow on the manipulator will cause structural oscillations. The oscillatory dynamic equations are exactly the lower part of Equation 4.1 with $\dot{\theta} = \ddot{\theta} = 0$, that is,

$$\Lambda_2 \ddot{\beta} + f_2(\dot{\theta} = 0) + K\beta = 0 \tag{4.2}$$

The above equation is a simple second order system with Λ_2 as the generalized mass matrix, f_2 as nonlinear damping force vector, and K as the spring matrix. For small oscillations, f_2 is negligible according to [Behi, 1985], then Equation 4.2 is reduced to

$$\ddot{\beta} + \Lambda_2^{-1} K\beta = 0 \tag{4.3}$$

Let $\ddot{\beta} = \lambda\beta$, then λ is the eigenvalue of

$$(\lambda \mathcal{I} - \Lambda_2^{-1} K)\beta = 0 \tag{4.4}$$

and β is the associated eigenvector. In the above equation, $\mathcal{I}$ is an $n_\beta \times n_\beta$ identity matrix. For a compliant manipulator modeled by n_β vibratory modes, there are n_β natural frequencies given by the square root of λ. For each natural frequency, the associated eigenvector gives information on the relative magnitudes of modeled modal amplitudes. These natural frequencies and eigenvectors could assist users to identify system inertial and stiffness parameters. For example, [Behi, 1985] has experimentally measured Cincinnati Milacron T3-776 robot oscillatory frequencies and vibratory modes and then used modal analysis methods to determine the inertial and stiffness

values. Of course, since Λ_2 is position dependent, the eigenvalues and eigenvectors change with manipulator position. Therefore, a least-square curve fitting approach may be used to average the experimentally obtained system parameters.

Another usage of the vibratory dynamic equations is to decide the quasi-static deflection of a compliant manipulator. A stationary manipulator is subjected to gravitational loads and hence deflection. Similarly, a working manipulator deforms under dynamic inertial loads. An approximation of analyzing the structural deformation of a moving robot is by quasi-static approach. In quasi-static analysis, robot is assumed stationary at every instance of motion, then the static deflection caused by the inertial and gravitational loads are computed for that position. This means that by neglecting the $\ddot{\beta}$ and $\dot{\beta}$ terms in Equation 4.1, the quasi-static deflection is given by

$$\beta = -K^{-1}(\Sigma\ddot{\theta} + f_2) \tag{4.5}$$

In the above equation, it is assumed that all β terms in Σ and f_2 are negligible, therefore, $\Sigma = \Sigma(\theta)$ and $f_2 = f_2(\theta, \dot{\theta})$. Since the quasi-static deflection is considered to be the major contributor of structural deformation, it is suggested by many researchers to compensate quasi-static deflection by either off-line trajectory planning or on-line control-in-the-small method as reported in the first chapter. Detailed analysis of quasi-static deflection and global stiffness matrix derivations are presented by [Fresonke, Hernandez, and Tesar, 1988], and real-time computation of manipulator quasi-static deformation is studied by [Hernandez, 1989].

4.3 Accessibility of Vibratory Modes

Another dynamic property to be examined is the effect of kinematics on vibratory mode motion control. In the dynamic equations

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 + K\beta \end{bmatrix} = \begin{bmatrix} u \\ 0 \end{bmatrix} \tag{4.6}$$

the generalized inertial matrix associated with acceleration term is positive definite therefore invertible. Then the inverse of the inertial matrix is defined as

$$\begin{bmatrix} A_1 & C^T \\ C & A_2 \end{bmatrix} \stackrel{\text{def}}{=} \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}^{-1} \tag{4.7}$$

$$= \begin{bmatrix} \left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} & -\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1}\Sigma^T\Lambda_2^{-1} \\ -\Lambda_2^{-1}\Sigma\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} & \Lambda_2^{-1} + \Lambda_2^{-1}\Sigma\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1}\Sigma^T\Lambda_2^{-1} \end{bmatrix}$$

where the second equation gives the inverse identity in terms of the submatrices of the original inertial matrix. In the above equations, the dimensions of each element are: $\Lambda_1 \in \mathcal{R}^{n_\theta \times n_\theta}$, $\Sigma \in \mathcal{R}^{n_\beta \times n_\theta}$, $\Lambda_2 \in \mathcal{R}^{n_\beta \times n_\beta}$, $f_1 \in \mathcal{R}^{n_\theta}$, $f_2 \in \mathcal{R}^{n_\beta}$, $K \in \mathcal{R}^{n_\beta \times n_\beta}$, $\theta \in \mathcal{R}^{n_\theta}$, and $\beta \in \mathcal{R}^{n_\beta}$. Also, $A_1 \in \mathcal{R}^{n_\theta \times n_\theta}$, $C \in \mathcal{R}^{n_\beta \times n_\theta}$, and $A_2 \in \mathcal{R}^{n_\beta \times n_\beta}$. By these definitions, the dynamic equations could be written in a form

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} A_1 & C^T \\ C & A_2 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 + K\beta \end{bmatrix} = \begin{bmatrix} A_1 \\ C \end{bmatrix} u \tag{4.8}$$

in which the column vectors of matrix $[A_1^T\, C^T]^T$ constitute the control space of u. Ultimately, these column vectors affect the control of u on $[\ddot{\theta}^T, \ddot{\beta}^T]^T$ values. Although the second nonlinear term in the left-hand side of the above equation will alter acceleration response for a given input u, the following analysis focuses on the direct relation between input u and system acceleration response, especially $\ddot{\beta}$, by investigating the properties of the gain matrix

$[A_1^T\ C^T]^T$. Equation 4.8 could be rewritten as

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} A_1 \\ C \end{bmatrix} u - \begin{bmatrix} A_1 \\ C \end{bmatrix} f_1 + \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \tag{4.9}$$

Since A_1 is a diagonal submatrix of a positive definite matrix, it has full rank, hence $\ddot{\theta}$ resides in the column space of A_1, and u can affect $\ddot{\theta}$ directly. But C is an off-diagonal submatrix whose rank is indecisive, so u might not be able to affect some vibrational accelerations directly. One situation which we define as *inaccessibility problem* is that C contains one or more null row vectors. In that case, control u loses direct access to the corresponding vibratory mode, therefore, that particular mode is dominated by the nonlinear term $f_2 + K\beta$ in the above equation. Notice that f_1 has the same gain matrix as u, so the inaccessible problem also occurs to f_1. It will be shown later that this accessibility problem generally does not imply a controllability problem, but without the direct influence of control input u, active damping on structural oscillation is impeded. Also, since f_2 and $K\beta$ are nonlinear terms governing the inaccessible mode, they must have a specific structure to dampen that inaccessible vibration. Since f_2 is a nonlinear coupling term, constructing a specific f_2 by u is neither transparent nor an easy task. Additionally, $K\beta$ represents structural resilient force, using this term to remove inaccessible oscillation will create unwanted deformations, which apparently is not an effective strategy. So, inaccessibility becomes a control problem for compliant manipulators. Another important feature of vibrational accessibility is its kinematic dependence. Since the value of C varies with manipulator posture, a vibrational mode could change from being accessible to inaccessible as manipulator changes its configuration. This kinematic dependency makes the study of inaccessibility of vibrational modes an important and valuable work in the control of compliant manipulators.

4.3.1 The Algebraic Interpretation of Vibratory Mode Accessibility

To investigate the occurrence of one or more null row vectors in C, the constituent elements of C are examined. From Equation 4.7, the inverse identity of C is expressed by

$$\begin{aligned} C &= -\Lambda_2^{-1}\Sigma\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} \\ &= \begin{bmatrix} -(\Lambda_2^{-1}\Sigma)_{1r}\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} \\ -(\Lambda_2^{-1}\Sigma)_{2r}\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} \\ \vdots \\ -(\Lambda_2^{-1}\Sigma)_{n_\beta r}\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} \end{bmatrix} \end{aligned} \tag{4.10}$$

where $(\Lambda_2^{-1}\Sigma)_{ir}$ is the ith row vector of $(\Lambda_2^{-1}\Sigma)$ with $i \in \{1, \cdots, n_\beta\}$ and $(\Lambda_2^{-1}\Sigma)_{ir}^T \in \mathcal{R}^{n_\theta}$. Notice that the second subscript r indicates a row vector, later on, another subscript c will be used to denote the column vector. Now, for example, if the jth row of C is a null vector, then

$$(\Lambda_2^{-1}\Sigma)_{jr}\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} = 0$$

which means that the jth row of C is zero when $(\Lambda_2^{-1}\Sigma)_{jr}$ is orthogonal to the matrix $\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1}$. Since $\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} \in \mathcal{R}^{n_\theta \times n_\theta}$ is an invertible matrix whose column vectors are linearly independent and span $\mathcal{R}^{n_\theta}$ space, $(\Lambda_2^{-1}\Sigma)_{jr}^T$ could be composed linearly by these column vectors. This implies that the orthogonality relation exists if and only if the jth row of $(\Lambda_2^{-1}\Sigma)$, i.e., $(\Lambda_2^{-1}\Sigma)_{jr}$, is a null vector. This could also be shown by postmultiplying $\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)$ to both sides of Equation 4.10 which produces

$$\begin{aligned} -\Lambda_2^{-1}\Sigma &= C\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right) \\ &= \begin{bmatrix} C_{1r}\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right) \\ \vdots \\ C_{n_\beta r}\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right) \end{bmatrix} \end{aligned} \tag{4.11}$$

where C_{ir} is the ith row of C, then if the jth row of C is zero, so does the jth row of $\Lambda_2^{-1}\Sigma$. This result reduces the accessibility examination to the properties of $(\Lambda_2^{-1}\Sigma)$. Defining

$$\begin{aligned} D &= (\Lambda_2^{-1}\Sigma) \in \mathcal{R}^{n_\beta \times n_\theta} \\ &= [D_{1c}\, D_{2c} \cdots D_{n_\theta c}] \end{aligned} \tag{4.12}$$

where D_{ic} is the ith column of D, then

$$\begin{aligned} \Sigma &= \Lambda_2 D \\ &= [(\Lambda_2 D_{1c})\,(\Lambda_2 D_{2c}) \cdots (\Lambda_2 D_{n_\theta c})] \\ &\stackrel{\text{def}}{=} [\Sigma_{1c}\, \Sigma_{2c} \cdots \Sigma_{n_\theta c}] \end{aligned} \tag{4.13}$$

where Σ_{ic} is the ith column of Σ and $\Sigma_{ic} = \Lambda_2 D_{ic}$ which could be further expressed as

$$\Sigma_{ic} = \sum_{k=1}^{n_\beta} D_{ki}(\Lambda_2)_{kc} \tag{4.14}$$

where $(\Lambda_2)_{kc}$ is the kth column of Λ_2, and D_{ki} is the kth row and ith column element of D. According to Equation 4.12, for the jth vibratory mode to be inaccessible, the jth row of D must be a null vector, which makes D_{ji} zero for all i, therefore,

$$\Sigma_{ic} = \sum_{\substack{k=1 \\ k \neq j}}^{n_\beta} D_{ki}(\Lambda_2)_{kc} \tag{4.15}$$

for all $i \in \{1, \cdots, n_\theta\}$. Which means that when all columns of Σ are linearly independent of the jth column of Λ_2, the jth vibratory mode is inaccessible. This could be verified from another approach. By defining

$$\Lambda_2^{-1}\Sigma = \begin{bmatrix} (\Lambda_2^{-1})_{1r}\Sigma \\ \vdots \\ (\Lambda_2^{-1})_{n_\beta r}\Sigma \end{bmatrix} \tag{4.16}$$

where $(\Lambda_2^{-1})_{ir}$ is the ith row of Λ_2^{-1}, so when the jth row of $(\Lambda_2^{-1}\Sigma)$ is a null vector, the jth row of Λ_2^{-1} must be orthogonal to Σ. But Λ_2^{-1} is invertible, its row vectors are not zero, therefore, it will be shown next that the orthogonal relation occurs only when all columns of Σ are linear independent of the jth column of Λ_2. Since Λ_2 is an invertible $n_\beta \times n_\beta$ matrix, its column vectors are linearly independent, and all vectors in $\mathcal{R}^{n_\beta}$ space could be composed by Λ_2 column vectors. Therefore, following the notation used in Equation 4.14, the ith column of Σ could be expressed by a linear combination of Λ_2 column vectors as

$$(\Sigma)_{ic} = \sum_{k=1}^{n_\beta} D_{ki}(\Lambda_2)_{kc} \tag{4.17}$$

where D_{ki} is a scalar coefficient. Also, since $\Lambda_2^{-1}\Lambda_2 = \mathcal{I}$ where $\mathcal{I}$ is an $n_\beta \times n_\beta$ identity matrix, which implies that $(\Lambda_2)_{jr}^{-1}(\Lambda_2)_{kc} = \delta_{jk}$ with $\delta_{jk} = 1$ at $j = k$ and zero if $j \neq k$, then for the jth vibrational mode in Equation 4.16 to be inaccessible, the following relation exits for all i, i.e.,

$$\begin{aligned}(\Lambda_2^{-1})_{jr}(\Sigma)_{ic} &= \sum_{k=1}^{n_\beta} D_{ki}(\Lambda_2^{-1})_{jr}(\Lambda_2)_{kc} \\ &= D_{ji} \\ &= 0 \end{aligned} \tag{4.18}$$

Together with Equation 4.17, the above result shows that when every column of Σ is linearly independent of the jth column of Λ_2, the jth row of C is a null vector and the jth vibratory mode is inaccessible. Which supports the result derived in Equation 4.15. Although the above analysis concentrates on one inaccessible vibratory mode, similar conclusion could be extended directly to the case of multi-inaccessible vibratory modes, which can be stated as: *when all columns of* Σ *are linearly independent of some particular columns of* Λ_2, *the corresponding vibratory mode accelerations are inaccessible to control input* u.

4.3.2 The Physical Interpretation of Vibratory Mode Accessibility

The above analytic results answer accessibility problem from algebraic point of view. Additional physical interpretation can be obtained from the roles of Σ and Λ_2 in the compliant system dynamic equations. According to Equation 4.6, the dynamics associated with vibratory modes are expressed by

$$\Sigma\ddot{\theta} + \Lambda_2\ddot{\beta} + f_2 + K\beta = 0 \tag{4.19}$$

where $\Sigma\ddot{\theta}$ and $\Lambda_2\ddot{\beta}$ are the inertial forces applied on the modeled vibratory coordinates and could be written in terms of column vectors as

$$\begin{aligned} \Sigma\ddot{\theta} &= \sum_{i=1}^{n_\theta}(\Sigma)_{ic}\ddot{\theta}_i \\ \Lambda_2\ddot{\beta} &= \sum_{i=1}^{n_\beta}(\Lambda_2)_{ic}\ddot{\beta}_i \end{aligned} \tag{4.20}$$

where $\ddot{\theta}_i$ and $\ddot{\beta}_i$ are the ith element of the corresponding acceleration vectors. Then the physical meanings of these column vectors could be interpreted as follows. For a unit $\ddot{\theta}_i$, the column vector $(\Sigma)_{ic}$ represents the associated inertial forces on all vibratory modes, and $(\Lambda_2)_{ic}$ is a vector of similar inertial forces contributed by a unit $\ddot{\beta}_i$. According to the inaccessibility analysis and Equation 4.19, it could be concluded that *when all $\ddot{\theta}$ inertial forces on all vibratory modes are linearly independent of a particular vibratory acceleration force on all vibratory modes then that specific vibratory acceleration is inaccessible to u.* Interestingly, premultiplying Equation 4.19 by Λ_2^{-1}, we obtain

$$\ddot{\beta} = -\Lambda_2^{-1}\Sigma\ddot{\theta} - \Lambda_2^{-1}(f_2 + K\beta) \tag{4.21}$$

If $\ddot{\theta}$ is considered as control input to the above equation, then any vibratory acceleration inaccessible to u will also be inaccessible to $\ddot{\theta}$ and vice versa.

In order to have a better understanding on the above interpretation of vibratory mode inaccessibility, a simple lumped parameter example is given for illustration purposes. The example, as shown in Figure 4.1 (a) and (b), is a two-link arm modeled with two lumped vibratory modes. For simplicity both links are assumed massless, only the payload has a mass m and moment of inertia I with $I = \text{diag}[I_x, I_y, I_z]$. In this example, the first link is considered rigid but the second link is compliant and modeled by two vibratory modes, in which the first vibratory mode, denoted by β_1, represents end-point transverse deflection, and the second vibratory mode, denoted by β_2, depicts axial torsional deformation. As shown in Figure 4.1, these vibratory modes are orthogonal to each other, and the translational mode β_1 is parallel to second joint axis, i.e., Z_2, and the twisting mode β_2 is along the X_2 direction. Let θ_1 be the first joint parameter and θ_2 be the second joint displacement measured from horizontal position. Then the generalized inertial matrix for this example is

$$I^* = \begin{bmatrix} (ml_2^2 + I_y - I_x)\cos^2\theta_2 + I_x & 0 & -ml_2\cos\theta_2 & I_x\sin\theta_2 \\ 0 & ml_2^2 + I_x & 0 & 0 \\ -ml_2\cos\theta_2 & 0 & M & 0 \\ I_x\sin\theta_2 & 0 & 0 & I_x \end{bmatrix} \tag{4.22}$$

In Figure 4.1(a), the second link is positioned at $\theta_2 = 0$ where

$$\Sigma = \begin{bmatrix} -ml_2 & 0 \\ 0 & 0 \end{bmatrix} \; ; \; \Lambda_2 = \begin{bmatrix} M & 0 \\ 0 & I_x \end{bmatrix} \tag{4.23}$$

In the above equation, both columns of Σ are linearly independent of the second column of Λ_2, therefore, according to the analytical interpretation, $\ddot{\beta}_2$ (but not $\ddot{\beta}_1$) is inaccessible. It will be shown that the inaccessibility is due to geometric orthogonality. Since $\ddot{\theta}_2$ rotates about Z_2 which is normal to both β_1 and β_2 vibrations, the second joint input can not access both modes in

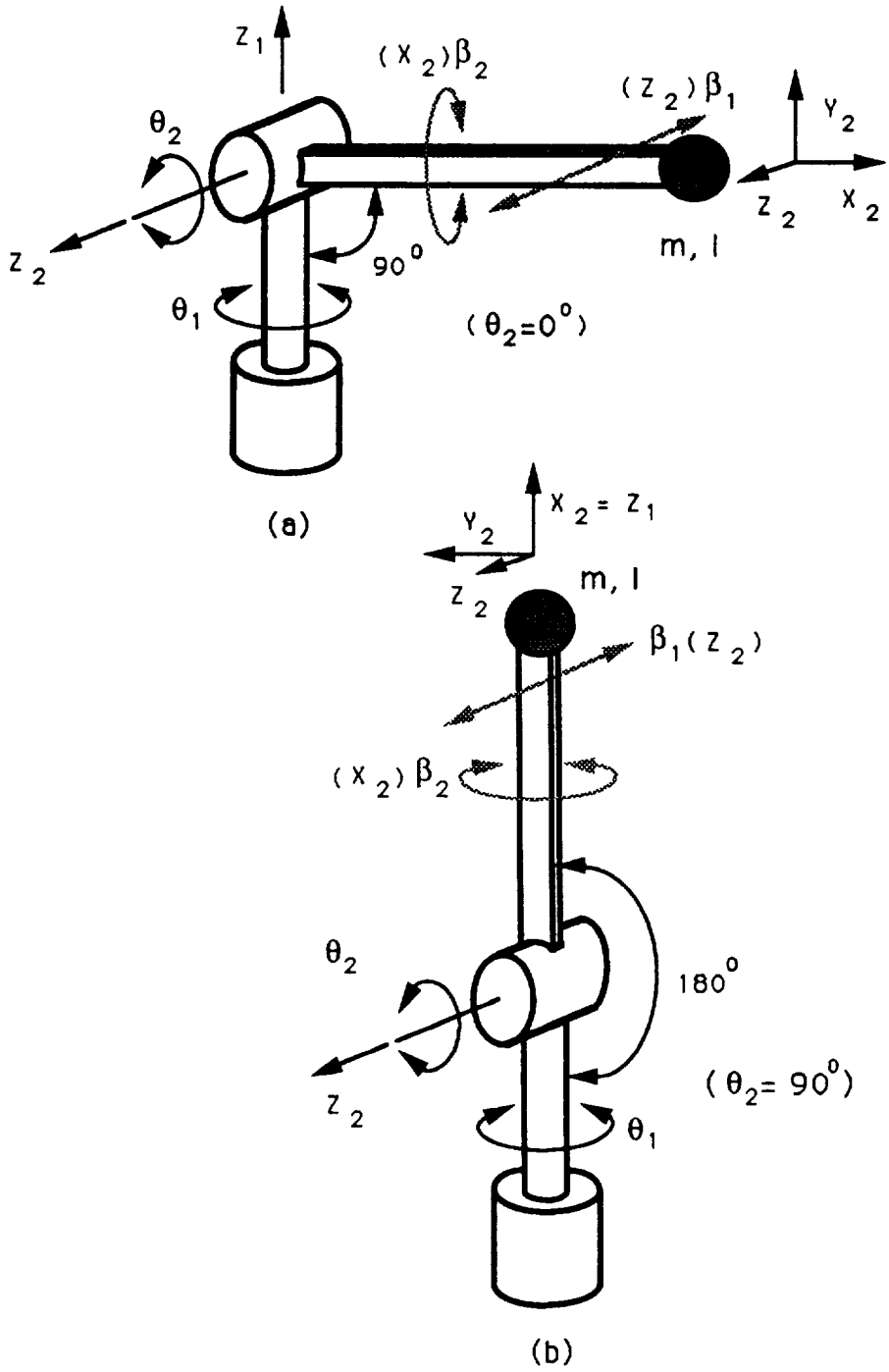


Figure 4.1: Inaccessible Positions of Two-Link Arm

any configuration. This result could be verified from Equation 4.22 directly where the third and fourth row elements of the second column are always zero. Hence, the accessibility analysis concentrates on the first joint effect. When $\theta_2 = 0$, $\ddot{\theta}_1$ creates an inertial force on the payload along Z_2 which is parallel to β_1 direction, then according to Equation 4.21, β_1 will be accessible. However, the torque on payload due to $\ddot{\theta}_1$ is along the Y_2 direction which is normal to the β_2 oscillation, hence β_2 is inaccessible at this position.

In Figure 4.1(b), the second link moves to $\theta_2 = 90$ deg, and

$$\Sigma = \begin{bmatrix} 0 & 0 \\ I_x & 0 \end{bmatrix} \; ; \; \Lambda_2 = \begin{bmatrix} M & 0 \\ 0 & I_x \end{bmatrix} \tag{4.24}$$

which shows that both columns in Σ are independent of the first column of Λ_2, therefore, β_1 (but not β_2) becomes inaccessible in this position. Physically, θ_1 and β_2 spin about the same axis at this position. Hence, the first joint can access the twisting mode. But θ_1 motion is normal to the β_1 lateral deflection, hence both joint inputs can not access β_1. Through the simple two-link example, we introduce the physical meaning of inaccessibility problem and also address its dependency on system kinematics. Notice that in this example, the inertial force and torque on both pseudo joints are contributed only by the payload, so the assumption of massless links does not oversimplify the results.

The next example is to investigate the inaccessible positions of a distributed parameter model. Now, the second link of the two-link arm in Figure 4.2 is flexible and modeled as a continuous beam whose lateral deflection is described by two assumed modes with polynomial mode shapes

$$\phi_1 = (\frac{r}{L_2})^2 \; ; \; \phi_2 = (\frac{r}{L_2})^2 - 1.2(\frac{r}{L_2})^3$$

in which r is a distance variable along link 2 which has a length L_2, and ϕ_1 and ϕ_2 are orthogonal such that $\int_0^{L_2} \phi_1 \phi_2 dr = 0$. In this example, only the lateral deflection along Z_2 is modeled, and it is assumed that rotatory inertial effect is negligible. So, for a uniform beam with mass m, mass moment of inertia $I_2 = \text{diag}[I_{x2}\ I_{y2}\ I_{z2}]$, and a point-mass payload M, the 2×2 Σ and Λ_2 matrices are given by

$$\Sigma = \begin{bmatrix} -\frac{mL_2}{4}\cos\theta_2 - ML_2\cos\theta_2 & 0 \\ -\frac{0.1}{10}mL_2\cos\theta_2 + 0.2ML_2\cos\theta_2 & 0 \end{bmatrix}; \Lambda_2 = \begin{bmatrix} \frac{m}{5} + M & -0.2M \\ -0.2M & \frac{1.2m}{210} + 0.02M \end{bmatrix}$$

Apparently, at $\cos\theta_2 = 0$ both columns in Σ are linearly independent of any column of Λ_2, hence both vibratory modes are inaccessible when $\theta_2 = \frac{2n+1}{2}\pi$ for any integer n. This result is consistent with that of the lumped two-link example in Figure 4.1. Notice that the second column of the above Σ is always zero, which means that the second joint actuator can not access the lateral vibration all the time. Such a result is predicable from the geometric orthogonality between the second joint and the lateral vibration. Geometric orthogonality also causes inaccessibility of both assumed modes to the first joint, that is, when the second link is coaxial with the first joint, the moment arm between the first joint and the lateral deflection vanishes, therefore, the first joint contributes no motion to the lateral deflection and consequently loses access of the lateral deflection.

4.3.3 The Structure of Σ and Λ_2

Due to geometric orthogonality, both inaccessibility analyses in the above two-link examples have interesting kinematic interpretation. However, in a general compliant manipulator where multi-link compliances are encountered, kinematic interpretation of the inaccessibility problem is not as transparent as in the two-link case. The following analysis will reveal the actual

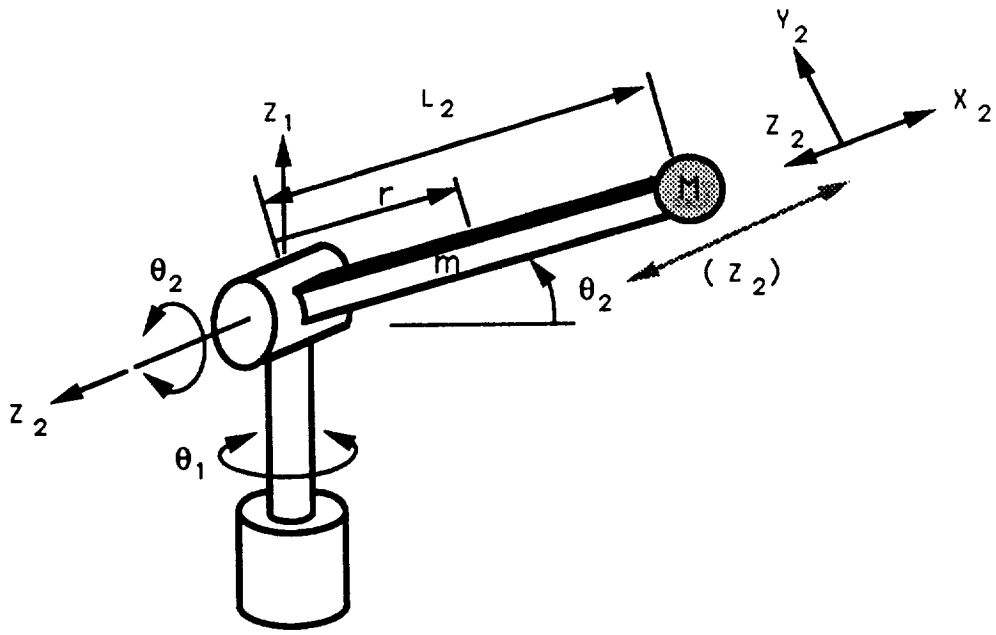


Figure 4.2: A Two-Link Arm with the Second Link Modeled by a Continuous Beam

nature of Σ and Λ_2 from which kinematic complexity of the inaccessibility problem will be evidently clear. For a lumped parameter model, the inertial matrix could be expressed by the first order influence coefficient representations as

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} = \sum_{i=1}^{N} \left\{ m_i \, {}_TG_i^T \, {}_TG_i + \, {}_RG_i^T I_i \, {}_RG_i \right\} \tag{4.25}$$

where ${}_TG_i \in \mathcal{R}^{3\times(n_\theta+n_\beta)}$ is the G function associated with translational motion of link i, and ${}_RG_i \in \mathcal{R}^{3\times(n_\theta+n_\beta)}$ is the G function associated with link i rotational motion. Also, $I_i \in \mathcal{R}^{3\times3}$ is a generalized moment of inertia defined as

$$I_i = T_i I_c^i T_i^T \tag{4.26}$$

in which $T_i \in \mathcal{R}^{3\times3}$ is a transformation matrix converting the local coordinates defined in the ith frame to global coordinates, and $I_c^i \in \mathcal{R}^{3\times3}$ is the moment of inertia of link i defined in the ith frame. The upper limit of summation, N, is the total number of links including payload. By dividing the G functions into two submatrices as

$$\begin{aligned} {}_TG_i &= [({}_TG_i)_\theta \, ({}_TG_i)_\beta] \\ {}_RG_i &= [({}_RG_i)_\theta \, ({}_RG_i)_\beta] \end{aligned} \tag{4.27}$$

where $({}_TG_i)_\theta$, $({}_RG_i)_\theta \in \mathcal{R}^{3\times n_\theta}$ and $({}_TG_i)_\beta$, $({}_RG_i)_\beta \in \mathcal{R}^{3\times n_\beta}$, then Equation 4.25 could be expanded into

$$\begin{aligned} \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} = \sum_{i=1}^{N} \Bigg\{ m_i \begin{bmatrix} ({}_TG_i)_\theta^T({}_TG_i)_\theta & ({}_TG_i)_\theta^T({}_TG_i)_\beta \\ ({}_TG_i)_\beta^T({}_TG_i)_\theta & ({}_TG_i)_\beta^T({}_TG_i)_\beta \end{bmatrix} \\ + \begin{bmatrix} ({}_RG_i)_\theta^T I_i({}_RG_i)_\theta & ({}_RG_i)_\theta^T I_i({}_RG_i)_\beta \\ ({}_RG_i)_\beta^T I_i({}_RG_i)_\theta & ({}_RG_i)_\beta^T I_i({}_RG_i)_\beta \end{bmatrix} \Bigg\} \end{aligned} \tag{4.28}$$

by comparing term by term, the submatrices Σ and Λ_2 have the following equivalent forms

$$\Sigma = \sum_{i=1}^{N} \left\{ m_i ({}_TG_i)_\beta^T ({}_TG_i)_\theta + ({}_RG_i)_\beta^T I_i ({}_RG_i)_\theta \right\} \tag{4.29}$$

$$\Lambda_2 = \sum_{i=1}^{N} \left\{ m_i ({}_T G_i)_\beta^T ({}_T G_i)_\beta + ({}_R G_i)_\beta^T I_i ({}_R G_i)_\beta \right\} \tag{4.30}$$

Similarly, for distributed parameter model,

$$\Sigma = \sum_{i=1}^{N} \left\{ \int ({}_T G_{p'})_\beta^T ({}_T G_{p'})_\theta dm + \int ({}_R G_{p'})_\beta^T (dI) ({}_R G_{p'})_\theta \right\} \tag{4.31}$$

$$\Lambda_2 = \sum_{i=1}^{N} \left\{ \int ({}_T G_{p'})_\beta^T ({}_T G_{p'})_\beta dm + \int ({}_R G_{p'})_\beta^T (dI) ({}_R G_{p'})_\beta \right\} \tag{4.32}$$

Details of both lumped and distributed parameter dynamics are given in Chapter 3. Although the kinematic interpretations of the above G functions are well-defined, they can not be extended to Σ and Λ_2 directly because the kinematic relations are coupled after the matrix multiplication. Besides that the mass and moment of inertia of all links are mixed in the Σ and Λ_2 expressions, that makes it difficult to obtain a simple geometric interpretation of the inaccessibility problem. So, before the kinematic effects of vibratory mode accessibility could be understood thoroughly, inaccessible modes can only be identified from Σ and Λ_2 analytically. Since it is highly demanding to check the dependency of each column of Σ and Λ_2, a practical approach is to compute $\Lambda_2^{-1}\Sigma$ symbolically to examine the occurrence of row or rows of zero vectors.

4.3.4 Case Studies of Inaccessible Vibratory Modes

To further examine robot position effects on vibratory mode accessibility, a three-link manipulator is modeled increasingly with one, two, four, and eight lumped vibratory modes. In each case, the inaccessible nominal position is computed symbolically for each vibratory mode. The purpose of using increasing number of vibratory modes is to check whether consistent inaccessible positions would be obtained as the modeled vibrations on a given

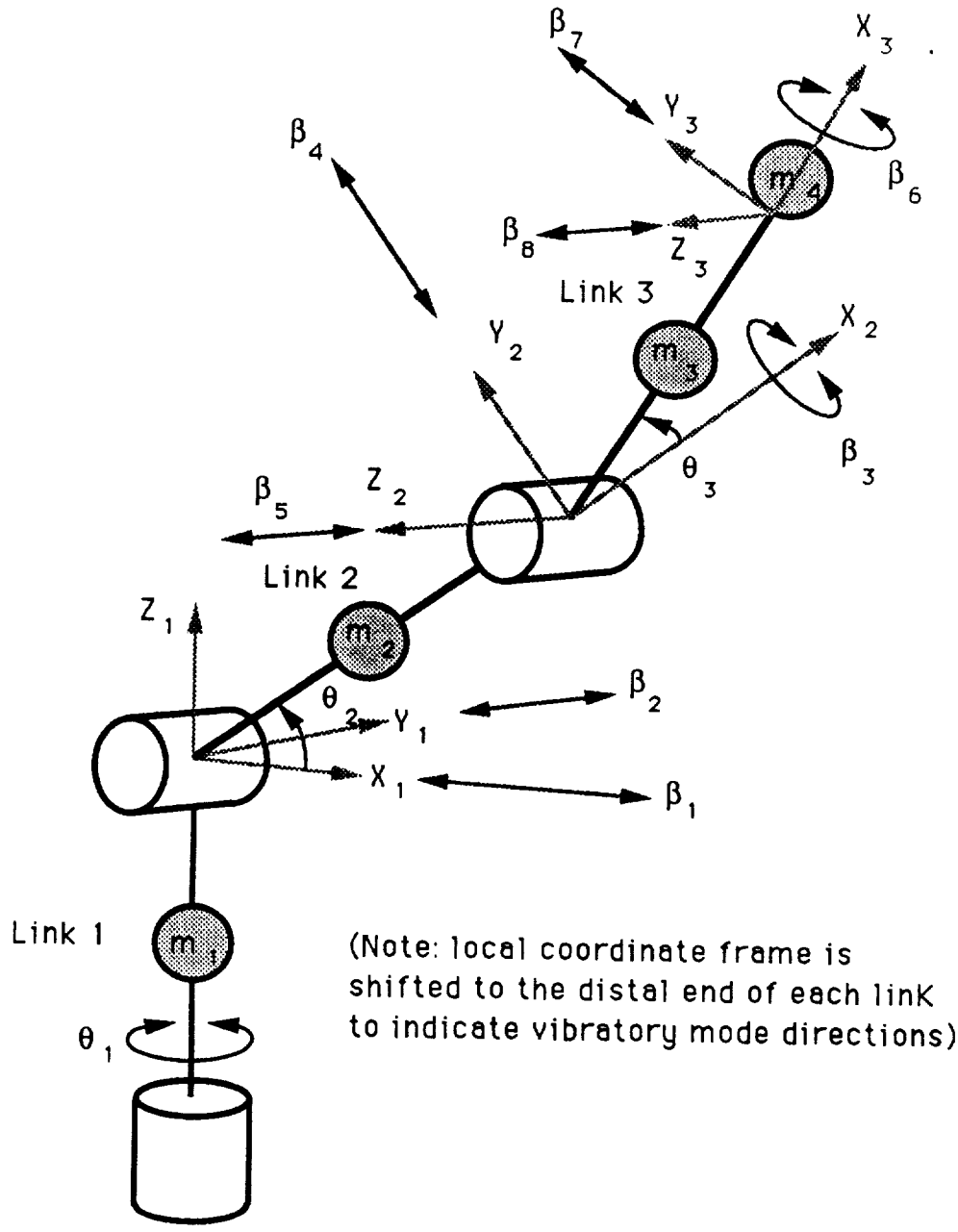


Figure 4.3: A 3-Link Arm Modeled with 8 Lumped Compliances

system are varied. Figure 4.3 is the three-link manipulator with all eight lumped vibratory modes. Each vibratory mode depicts a link deflection in a particular direction. Notice that each local frame is shifted to the distal end of the link to assist description of vibratory mode directions. For example, β_1 is the first link lateral deflection along the x_1 direction, β_4 is the second link lateral deformation along the y_2 direction, and β_6 is the third link twisting along the x_3 direction, and so forth. These local coordinates will be helpful in presenting the investigation results. For example, when four modes, β_1, β_2, β_3, and β_5 are modeled, they will be denoted as $x_1y_1x_2z_2$ for compact indication of which link and direction of vibrations are involved in the study. In these eight vibratory modes, β_4 (y_2) and β_7 (y_3) are always accessible because kinematically $\ddot{\theta}_2$ can affect $\ddot{\beta}_4$ directly, so does $\ddot{\theta}_3$ to $\ddot{\beta}_7$. Therefore, they are excluded from the following report. The other six inaccessible mode results are tabulated in Tables 4.1 (a) to (f), and in each table, the inaccessible mode, the modeled vibratory modes, and the inaccessible position are listed.

Table 4.1 (a) β_1 (x_1) mode

modeled vibrational modes	inaccessible position (deg)
x_1	$\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180
x_1y_1	$\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180
$x_1y_1x_2z_2$	$\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180
$x_1y_1x_2z_2x_3z_3$	$\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180
$x_1y_1x_2y_2z_2x_3y_3z_3$	$\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180

Table 4.1 (b) β_2 (y_1) mode

modeled vibrational modes	inaccessible position (deg)
y_1	$\theta_2 = 90$ or 270 and $\theta_3 = 0$ or 180
x_1y_1	$\theta_2 = 90$ or 270 and $\theta_3 = 0$ or 180
$x_1y_1x_2z_2$	$\theta_2 = 90$ or 270
$x_1y_1x_2z_2x_3z_3$	$\theta_2 = 90$ or 270
$x_1y_1x_2y_2z_2x_3y_3z_3$	$\theta_2 = 90$ or 270

Table 4.1 (c) β_3 (x_2) mode	
modeled vibrational modes	inaccessible position (deg)
x_2	$\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180
$x_2 z_2$	$\theta_2 = 0$ or 180 and ($\theta_3 = 0$ or 180 or $\theta_3 = 90$ or 270)
$x_1 y_1 x_2 z_2$	$\theta_2 = 0$ or 180 and ($\theta_3 = 0$ or 180 or $\theta_3 = 90$ or 270)
$x_1 y_1 x_2 z_2 x_3 z_3$	$\theta_2 = 0$ or 180 and ($\theta_3 = 0$ or 180 or $\theta_3 = 90$ or 270)
$x_1 y_1 x_2 y_2 z_2 x_3 y_3 z_3$	$\theta_2 = 0$ or 180 and ($\theta_3 = 0$ or 180 or $\theta_3 = 90$ or 270)

Table 4.1 (d) β_5 (z_2) mode	
modeled vibrational modes	inaccessible position (deg)
z_2	$\theta_2 = 90$ or 270 and $\theta_3 = 0$ or 180
$x_2 z_2$	$\theta_2 = 90$ or 270 and $\theta_3 = 0$ or 180
$x_1 y_1 x_2 z_2$	$\theta_2 = 90$ or 270
$x_1 y_1 x_2 z_2 x_3 z_3$	$\theta_2 = 90$ or 270
$x_1 y_1 x_2 y_2 z_2 x_3 y_3 z_3$	$\theta_2 = 90$ or 270

Table 4.1 (e) β_6 (x_3) mode	
modeled vibrational modes	inaccessible position (deg)
x_3	($\theta_2 = 90$ or 270 and $\theta_3 = 90$ or 270) or ($\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180)
$x_3 z_3$	($\theta_2 = 90$ or 270 and $\theta_3 = 90$ or 270) or ($\theta_2 = 0$ or 180 and $\theta_3 = 0$ or 180)
$x_1 y_1 x_2 z_2 x_3 z_3$	$\theta_2 = 90$ or 270 or $\theta_3 = 0$ or 180
$x_1 y_1 x_2 y_2 z_2 x_3 y_3 z_3$	$\theta_2 = 90$ or 270 or $\theta_3 = 0$ or 180

Table 4.1 (f) β_8 (z_3) mode	
modeled vibrational modes	inaccessible position (deg)
z_3	$\theta_2 = 90$ or 270 and $\theta_3 = 0$ or 180
$x_3 z_3$	$\theta_2 = 90$ or 270 and $\theta_3 = 0$ or 180
$x_1 y_1 x_2 z_2 x_3 z_3$	$\theta_2 = 90$ or 270 or $\theta_3 = 90$ or 270
$x_1 y_1 x_2 y_2 z_2 x_3 y_3 z_3$	$\theta_2 = 90$ or 270 or $\theta_3 = 90$ or 270

For example, in Table 4.1 (b), when $x_1 y_1$ vibratory modes are modeled, $\beta_2(y_1)$ is inaccessible at $\theta_2 = 90$ or 270 deg and $\theta_3 = 0$ or 180 deg, and in the $x_1 y_1 x_2 z_2 x_3 z_3$ case, the inaccessible position of $\beta_2(y_1)$ mode is at $\theta_2 = 90$ or 270 deg regardless θ_3 value. Notice that both results are consistent,

because $\theta_2 = 90$ or 270 deg and $\theta_2 = 0$ and 180 deg is a special case covered by the general outcome that $\beta_2(y_1)$ is inaccessible when $\theta_2 = 90$ or 270 deg. Similar tendency could also be observed from the above tables that when larger number of vibrational modes are modeled the results are much complete and general. Details of $\Lambda_2^{-1}\Sigma$ symbolic results are listed in Table 4.2. Since Λ_2 is an invertible matrix whose determinant is not zero and has no effect on the final results, Table 4.2 is generated by the product of the adjoint of Λ_2 and Σ. Since the example is a three-link manipulator, each row vector of $\Lambda_2^{-1}\Sigma$ contains three elements. In Table 4.2, the three row-vector elements associated with each vibratory mode are denoted sequentially by [1], [2], and [3] for the first, second, and third elements. The symbols used in Table 4.2 are described below. m_i is the mass of link i with $i = 1, 2, 3$, and payload with $i = 4$, also l_i is the link length, r_i is the center of mass location along link i, and $I_i = \text{diag}[I_{xi}, I_{yi}, I_{zi}]$ is the moment of inertial. Interestingly, in the results some particular row elements are always zero despite the manipulator configuration, which means that the corresponding actuator inputs can not access that vibratory mode in any robot position. In the symbolic results, the common sinusoidal terms are highlighted, and once they take on a value of zero, that vibratory mode has a null row vector and becomes inaccessible. In the case studies, we examine the inaccessible position of a lumped parameter model. Now, one interesting question is that could lumped parameter results be extended directly to distributed parameter model? This means that if the first mode is inaccessible then does that imply inaccessibility of higher order modes? Although such implications are observed from the two-link examples in Figures 4.1 and 4.2, yet, due to complexity of Σ and Λ_2, we could not answer the question analytically. However, since the first mode dominates structural deflection, its inaccessibility should be identified and avoided. Therefore, we

emphasize the first mode instead of other higher order modes inaccessibility in the three-link arm case studies. But it should be noticed that the algebraic and physical interpretations of vibratory mode inaccessibility are built on a symbolic form general for both lumped and distributed parameter models.

Notice that no joint compliance is modeled in the case studies because in modeling a joint compliance the pseudo joint is added collinearly with the physical joint, hence both nominal and vibratory inertial torques projected on all vibratory coordinates are linearly dependent, therefore, joint vibration is always accessible and hence excluded from the study. The investigation of inaccessible vibratory modes is important to off-line decision making on manipulator architecture and working position. It constitutes a criterion to help the user to choose a suitable manipulator for a given task. And for existent manipulators, finding out the inaccessible position will avoid manipulators from working in undesirable positions where structural vibrations can not be dampened actively. As mentioned before, inaccessibility problem is distinct from the controllability problem. Despite the fact that it is difficult to control an inaccessible vibratory mode, the controllability of an inaccessible mode will be discussed in the next section.

Table 4.2

x_1 of x_1

[1]

0

[2]

$(-l_3m_4 - r_3m_3)\cos(\theta_2)\mathbf{sin}(\theta_3) + (-l_3m_4 - r_3m_3)\mathbf{sin}(\theta_2)\cos(\theta_3) +$
$(-l_2m_4 - l_2m_3 - r_2m_2)\mathbf{sin}(\theta_2)$

[3]

$(-l_3m_4 - r_3m_3)\cos(\theta_2)\mathbf{sin}(\theta_3) + (-l_3m_4 - r_3m_3)\mathbf{sin}(\theta_2)\cos(\theta_3)$

x_1 of x_1y_1

[1]

0

[2]

$(-l_3{m_4}^2 + ((-l_3 - r_3)m_3 - l_3m_2)m_4 - r_3{m_3}^2 - r_3m_2m_3)\cos(\theta_2)\mathbf{sin}(\theta_3) +$
$(-l_3{m_4}^2 + ((-l_3 - r_3)m_3 - l_3m_2)m_4 - r_3{m_3}^2 - r_3m_2m_3)\mathbf{sin}(\theta_2)\cos(\theta_3)$
$+(-l_2{m_4}^2 + ((-l_2 - r_2)m_2 - 2l_2m_3)m_4 - l_2{m_3}^2 +$
$(-l_2 - r_2)m_2m_3 - r_2{m_2}^2)\mathbf{sin}(\theta_2)$

[3]

$(-l_3{m_4}^2 + ((-l_3 - r_3)m_3 - l_3m_2)m_4 - r_3{m_3}^2 - r_3m_2m_3)\cos(\theta_2)\mathbf{sin}(\theta_3) +$
$(-l_3{m_4}^2 + ((-l_3 - r_3)m_3 - l_3m_2)m_4 - r_3{m_3}^2 - r_3m_2m_3)\mathbf{sin}(\theta_2)\cos(\theta_3)$

Table 4.2 (continued)

x_1 of $x_1y_1x_2z_2$

[1]

0

[2]

$$(((-{l_3}^2+2r_3{l_3}^2-{r_3}^2l_3)m_2m_3+(-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3m_2)m_4{}^2+$$
$$((-r_3{l_3}^2+2{r_3}^2l_3-{r_3}^3)m_2{m_3}^2+((-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3-$$
$$r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+r_3I_{x3})m_2m_3)m_4+$$
$$(-r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+r_3I_{x3})m_2{m_3}^2)\cos(\theta_2)\mathbf{sin}^3(\theta_3)+$$
$$((((-{l_3}^3+2r_3{l_3}^2-{r_3}^2l_3)m_2m_3+(-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3m_2)m_4{}^2+$$
$$((-r_3{l_3}^2+2{r_3}^2l_3-{r_3}^3)m_2{m_3}^2+((-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3-$$
$$r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+r_3I_{x3})m_2m_3)m_4+(-r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+$$
$$r_3I_{x3})m_2{m_3}^2)\mathbf{sin}(\theta_2)\cos(\theta_3)+(((-l_2{l_3}^2+2r_3l_2l_3-{r_3}^2l_2)m_2m_3+$$
$$(-I_{y4}l_2-I_{y3}l_2+I_{x4}l_2+I_{x3}l_2)m_2)m_4{}^2+$$
$$((-l_2{l_3}^2+2r_3l_2l_3-{r_3}^2l_2)m_2{m_3}^2+$$
$$((-r_2{l_3}^2+2r_2r_3l_3-r_2{r_3}^2)m_2{}^2+(-2I_{y4}l_2-2I_{y3}l_2+2I_{x4}l_2+2I_{x3}l_2)m_2)m_3+$$
$$(-r_2I_{y4}-r_2I_{y3}+r_2I_{x4}+r_2I_{x3})m_2{}^2)m_4+$$
$$(-I_{y4}l_2-I_{y3}l_2+I_{x4}l_2+I_{x3}l_2)m_2{m_3}^2+$$
$$(-r_2I_{y4}-r_2I_{y3}+r_2I_{x4}+r_2I_{x3})m_2{}^2m_3)\mathbf{sin}(\theta_2)\mathbf{sin}^2(\theta_3)+$$
$$((-I_{x4}-I_{x3})l_3m_2m_4{}^2+((-I_{x4}-I_{x3})l_3-r_3I_{x4}-r_3I_{x3})m_2m_3m_4+$$
$$(-r_3I_{x4}-r_3I_{x3})m_2{m_3}^2)\cos(\theta_2)\mathbf{sin}(\theta_3)+$$
$$((-I_{x4}-I_{x3})l_3m_2m_4{}^2+((-I_{x4}-I_{x3})l_3-r_3I_{x4}-r_3I_{x3})m_2m_3m_4+$$
$$(-r_3I_{x4}-r_3I_{x3})m_2{m_3}^2)\mathbf{sin}(\theta_2)\cos(\theta_3)+((-I_{x4}l_2-I_{x3}l_2)m_2m_4{}^2$$
$$+((-2I_{x4}l_2-2I_{x3}l_2)m_2m_3+(-r_2I_{x4}-r_2I_{x3})m_2{}^2)m_4+$$
$$(-I_{x4}l_2-I_{x3}l_2)m_2{m_3}^2+(-r_2I_{x4}-r_2I_{x3})m_2{}^2m_3)\mathbf{sin}(\theta_2)$$

[3]

$$(((-{l_3}^3+2r_3{l_3}^2-{r_3}^2l_3)m_2m_3+(-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3m_2)m_4{}^2+$$
$$((-r_3{l_3}^2+2{r_3}^2l_3-{r_3}^3)m_2{m_3}^2+$$
$$((-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3-r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+$$
$$r_3I_{x3})m_2m_3)m_4+(-r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+r_3I_{x3})m_2{m_3}^2)\cos(\theta_2)\mathbf{sin}^3(\theta_3)$$
$$+(((-{l_3}^3+2r_3{l_3}^2-{r_3}^2l_3)m_2m_3+(-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3m_2)m_4{}^2+$$
$$((-r_3{l_3}^2+2{r_3}^2l_3-{r_3}^3)m_2{m_3}^2+((-I_{y4}-I_{y3}+I_{x4}+I_{x3})l_3-$$
$$r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+r_3I_{x3})m_2m_3)m_4+$$
$$(-r_3I_{y4}-r_3I_{y3}+r_3I_{x4}+r_3I_{x3})m_2{m_3}^2)\mathbf{sin}(\theta_2)\cos(\theta_3)\mathbf{sin}^2(\theta_3)+$$
$$((-I_{x4}-I_{x3})l_3m_2m_4{}^2+((-I_{x4}-I_{x3})l_3-r_3I_{x4}-r_3I_{x3})m_2m_3m_4+$$
$$(-r_3I_{x4}-r_3I_{x3})m_2{m_3}^2)\cos(\theta_2)\mathbf{sin}(\theta_3)+((-I_{x4}-I_{x3})l_3m_2m_4{}^2+$$
$$((-I_{x4}-I_{x3})l_3-r_3I_{x4}-r_3I_{x3})m_2m_3m_4+$$
$$(-r_3I_{x4}-r_3I_{x3})m_2{m_3}^2)\mathbf{sin}(\theta_2)\cos(\theta_3)$$

Table 4.2 (continued)

x_1 of $x_1y_1x_2z_2x_3z_3$

[1]

0

[2]

$((-I_{x4}I_{y4}l_3 - I_{x4}I_{y3}l_3 + I_{x3}I_{x4}l_3)m_2m_3m_4{}^2+$
$(-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^2m_4)\cos(\theta_2)\sin^3(\theta_3)+$
$(((-I_{x4}I_{y4}l_3 - I_{x4}I_{y3}l_3 + I_{x3}I_{x4}l_3)m_2m_3m_4{}^2+$
$(-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^2m_4)\sin(\theta_2)\cos(\theta_3)+$
$((-I_{x4}I_{y4}l_2 - I_{x4}I_{y3}l_2 + I_{x3}I_{x4}l_2)m_2m_3m_4{}^2+$
$((-I_{x4}I_{y4}l_2 - I_{x4}I_{y3}l_2 + I_{x3}I_{x4}l_2)m_2m_3{}^2+$
$(-r_2I_{x4}I_{y4} - r_2I_{x4}I_{y3} + r_2I_{x3}I_{x4})m_2{}^2m_3)m_4)\sin(\theta_2))\sin^2(\theta_3)+$
$(-I_{x3}I_{x4}l_3m_2m_3m_4{}^2 - r_3I_{x3}I_{x4}m_2m_3{}^2m_4)\cos(\theta_2)\sin(\theta_3)+$
$(-I_{x3}I_{x4}l_3m_2m_3m_4{}^2 - r_3I_{x3}I_{x4}m_2m_3{}^2m_4)\sin(\theta_2)\cos(\theta_3)+$
$((-I_{x3}I_{x4}l_2m_2m_3{}^2 - r_2I_{x3}I_{x4}m_2{}^2m_3)m_4 - I_{x3}I_{x4}l_2m_2m_3m_4{}^2)\sin(\theta_2)$

[3]

$((-I_{x4}I_{y4}l_3 - I_{x4}I_{y3}l_3 + I_{x3}I_{x4}l_3)m_2m_3m_4{}^2+$
$(-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^2m_4)\cos(\theta_2)\sin^3(\theta_3)+$
$((-I_{x4}I_{y4}l_3 - I_{x4}I_{y3}l_3 + I_{x3}I_{x4}l_3)m_2m_3m_4{}^2+$
$(-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^2m_4)\sin(\theta_2)\cos(\theta_3)\sin^2(\theta_3)+$
$(-I_{x3}I_{x4}l_3m_2m_3m_4{}^2 - r_3I_{x3}I_{x4}m_2m_3{}^2m_4)\cos(\theta_2)\sin(\theta_3)+$
$(-I_{x3}I_{x4}l_3m_2m_3m_4{}^2 - r_3I_{x3}I_{x4}m_2m_3{}^2m_4)\sin(\theta_2)\cos(\theta_3)$

x_1 of $x_1y_1x_2y_2z_2x_3y_3z_3$

[1]

0

[2]

$(-r_2I_{x4}I_{y4} - r_2I_{x4}I_{y3} + r_2I_{x3}I_{x4})m_2{}^2m_3m_4{}^3\sin(\theta_2)\sin^4(\theta_3)+$
$((-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^2m_4{}^3+$
$(-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^3m_4{}^2)\cos(\theta_2)\sin^3(\theta_3)+$
$((-r_2I_{x4}I_{y4} - r_2I_{x4}I_{y3} + r_2I_{x3}I_{x4})m_2{}^2m_3{}^3m_4{}^2-$
$r_2I_{x3}I_{x4}m_2{}^2m_3m_4{}^3)\sin(\theta_2)\sin^2(\theta_3)+$
$(-r_3I_{x3}I_{x4}m_2m_3{}^2m_4{}^3 - r_3I_{x3}I_{x4}m_2m_3{}^3m_4{}^2)\cos(\theta_2)\sin(\theta_3)$
$-r_2I_{x3}I_{x4}m_2{}^2m_3{}^2m_4{}^2\sin(\theta_2)$

[3]

$((-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^2m_4{}^3+$
$(-r_3I_{x4}I_{y4} - r_3I_{x4}I_{y3} + r_3I_{x3}I_{x4})m_2m_3{}^3m_4{}^2)\cos(\theta_2)\sin^3(\theta_3)$
$+(-r_3I_{x3}I_{x4}m_2m_3{}^2m_4{}^3 - r_3I_{x3}I_{x4}m_2m_3{}^3m_4{}^2)\cos(\theta_2)\sin(\theta_3)$

Table 4.2 (continued)

y_1 of y_1

[1]

$(-l_3m_4 - r_3m_3)\sin(\theta_2)\mathbf{sin}(\boldsymbol{\theta}_3) + (l_3m_4 + r_3m_3)\mathbf{cos}(\boldsymbol{\theta}_2)\cos(\theta_3)$
$+(l_2m_4 + l_2m_3 + r_2m_2)\mathbf{cos}(\boldsymbol{\theta}_2)$

[2]

0

[3]

0

y_1 of x_1y_1

[1]

$(-l_3{m_4}^2 + ((-l_3 - r_3)m_3 - l_3m_2)m_4 - r_3{m_3}^2 - r_3m_2m_3)\sin(\theta_2)\mathbf{sin}(\boldsymbol{\theta}_3)+$
$(l_3{m_4}^2 + ((l_3 + r_3)m_3 + l_3m_2)m_4 + r_3{m_3}^2 + r_3m_2m_3)\mathbf{cos}(\boldsymbol{\theta}_2)\cos(\theta_3)+$
$(l_2{m_4}^2 + (2l_2m_3 + (l_2 + r_2)m_2)m_4 + l_2{m_3}^2 + (l_2 + r_2)m_2m_3 + r_2{m_2}^2)\mathbf{cos}(\boldsymbol{\theta}_2)$

[2]

0

[3]

0

y_1 of $x_1y_1x_2z_2$

[1]

$(((r_2{l_3}^2 - 2r_2r_3l_3 + r_2{r_3}^2)m_2m_3 + (r_2I_{y4} + r_2I_{y3} - r_2I_{x4} - r_2I_{x3})m_2){m_4}^2+$
$((r_2{l_3}^2 - 2r_2r_3l_3 + r_2{r_3}^2)m_2{m_3}^2 + ((r_2{l_3}^2 - 2r_2r_3l_3 + r_2{r_3}^2){m_2}^2+$
$(2r_2I_{y4} + 2r_2I_{y3} - 2r_2I_{x4} - 2r_2I_{x3})m_2)m_3+$
$(r_2I_{y4} + r_2I_{y3} - r_2I_{x4} - r_2I_{x3}){m_2}^2)m_4+$
$(r_2I_{y4} + r_2I_{y3} - r_2I_{x4} - r_2I_{x3})m_2{m_3}^2 + (r_2I_{y4} + r_2I_{y3} - r_2I_{x4} - r_2I_{x3}){m_2}^2m_3)$
$\mathbf{cos}(\boldsymbol{\theta}_2)\sin^2(\theta_3) + ((r_2I_{x4} + r_2I_{x3})m_2{m_4}^2 + ((2r_2I_{x4} + 2r_2I_{x3})m_2m_3+$
$(r_2I_{x4} + r_2I_{x3}){m_2}^2)m_4 + (r_2I_{x4} + r_2I_{x3})m_2{m_3}^3+$
$(r_2I_{x4} + r_2I_{x3}){m_2}^2m_3)\mathbf{cos}(\boldsymbol{\theta}_2)$

[2]

0

[3]

0

Table 4.2 (continued)

y_1 of $x_1y_1x_2z_2x_3z_3$

[1]

$((r_2I_{x4}I_{y4} + r_2I_{x4}I_{y3} - r_2I_{x3}I_{x4})m_2m_3{m_4}^2+$
$((r_2I_{x4}I_{y4} + r_2I_{x4}I_{y3} - r_2I_{x3}I_{x4})m_2{m_3}^2+$
$(r_2I_{x4}I_{y4} + r_2I_{x4}I_{y3} - r_2I_{x3}I_{x4}){m_2}^2m_3)m_4)\mathbf{cos}(\theta_2)\sin^2(\theta_3)+$
$(r_2I_{x3}I_{x4}m_2m_3{m_4}^2 + (r_2I_{x3}I_{x4}m_2{m_3}^2 + r_2I_{x3}I_{x4}{m_2}^2m_3)m_4)\mathbf{cos}(\theta_2)$

[2]

0

[3]

0

y_1 of $x_1y_1x_2y_2z_2x_3y_3z_3$

[1]

$(r_2I_{x4}I_{y4} + r_2I_{x4}I_{y3} - r_2I_{x3}I_{x4}){m_2}^2m_3{m_4}^3\mathbf{cos}(\theta_2)\sin^4(\theta_3)+$
$(((r_2I_{x4}I_{y4} + r_2I_{x4}I_{y3} - r_2I_{x3}I_{x4})m_2{m_3}^2{m_4}^3+$
$(r_2I_{x4}I_{y4} + r_2I_{x4}I_{y3} - r_2I_{x3}I_{x4})m_2{m_3}^3{m_4}^2)\mathbf{cos}^3(\theta_2)+$
$(r_2I_{x3}I_{x4}{m_2}^2m_3{m_4}^3 + (r_2I_{x4}I_{y4} + r_2I_{x4}I_{y3} - r_2I_{x3}I_{x4})$
${m_2}^2{m_3}^2{m_4}^2)\mathbf{cos}(\theta_2)\sin^2(\theta_3)+$
$(r_2I_{x3}I_{x4}m_2{m_3}^2{m_4}^3 + r_2I_{x3}I_{x4}m_2{m_3}^3{m_4}^2)\mathbf{cos}^3(\theta_2)+$
$r_2I_{x3}I_{x4}{m_2}^2{m_3}^2{m_4}^2\mathbf{cos}(\theta_2)$

[2]

0

[3]

0

Table 4.2 (continued)

x_2 of x_2

[1]

$(l_3{}^2 m_4 + r_3{}^2 m_3 + I_{y4} + I_{y3})\mathbf{sin}(\theta_2)\mathbf{sin}^2(\theta_3)+$
$((-l_3{}^2 m_4 - r_3{}^2 m_3 - I_{y4} - I_{y3} + I_{x4} + I_{x3})\cos(\theta_2)\cos(\theta_3)+$
$(-l_2 l_3 m_4 - r_3 l_2 m_3)\cos(\theta_2))\mathbf{sin}(\theta_3)+$
$(I_{x4} + I_{x3})\mathbf{sin}(\theta_2)\cos^2(\theta_3)$

[2]

0

[3]

0

x_2 of $x_2 z_2$

[1]

$(((l_3{}^2 - r_3 l_3)m_3 + I_{y4} + I_{y3} - I_{x4} - I_{x3})m_4+$
$(I_{y4} + I_{y3} - I_{x4} - I_{x3})m_3)\mathbf{sin}(\theta_2)\mathbf{sin}^2(\theta_3)+$
$(((r_3 l_3 - l_3{}^2)m_3 - I_{y4} - I_{y3} + I_{x4} + I_{x3})m_4+$
$(-I_{y4} - I_{y3} + I_{x4} + I_{x3})m_3)\cos(\theta_2)\mathbf{cos}(\theta_3)\mathbf{sin}(\theta_3)+$
$((I_{x4} + I_{x3})m_4 + (I_{x4} + I_{x3})m_3)\mathbf{sin}(\theta_2)$

[2]

0

[3]

0

x_2 of $x_1 y_1 x_2 z_2$

[1]

$(((l_3{}^2 - 2r_3 l_3 + r_3{}^2)m_2 m_3 + (I_{y4} + I_{y3} - I_{x4} - I_{x3})m_2)m_4{}^2+$
$((l_3{}^2 - 2r_3 l_3 + r_3{}^2)m_2 m_3{}^2 + ((l_3{}^2 - 2r_3 l_3 + r_3{}^2)m_2{}^2+$
$(2I_{y4} + 2I_{y3} - 2I_{x4} - 2I_{x3})m_2)m_3 + (I_{y4} + I_{y3} - I_{x4} - I_{x3})m_2{}^2)m_4+$
$(I_{y4} + I_{y3} - I_{x4} - I_{x3})m_2 m_3{}^2 + (I_{y4} + I_{y3} - I_{x4} - I_{x3})m_2{}^2 m_3)\mathbf{sin}(\theta_2)\mathbf{sin}^2(\theta_3)+$
$(((-l_3{}^2 + 2r_3 l_3 - r_3{}^2)m_2 m_3 + (-I_{y4} - I_{y3} + I_{x4} + I_{x3})m_2)m_4{}^2+$
$((-l_3{}^2 + 2r_3 l_3 - r_3{}^2)m_2 m_3{}^2 + ((-l_3{}^2 + 2r_3 l_3 - r_3{}^2)m_2{}^2+$
$(-2I_{y4} - 2I_{y3} + 2I_{x4} + 2I_{x3})m_2)m_3 + (-I_{y4} - I_{y3} + I_{x4} + I_{x3})m_2{}^2)m_4+$
$(-I_{y4} - I_{y3} + I_{x4} + I_{x3})m_2 m_3{}^2+$
$(-I_{y4} - I_{y3} + I_{x4} + I_{x3})m_2{}^2 m_3)\cos(\theta_2)\mathbf{cos}(\theta_3)\mathbf{sin}(\theta_3)+$
$((I_{x4} + I_{x3})m_2 m_4{}^2 + ((2I_{x4} + 2I_{x3})m_2 m_3 + (I_{x4} + I_{x3})m_2{}^2)m_4+$
$(I_{x4} + I_{x3})m_2 m_3{}^2 + (I_{x4} + I_{x3})m_2{}^2 m_3)\mathbf{sin}(\theta_2)$

[2]

0

[3]

0

Table 4.2 (continued)

x_2 of $x_1y_1x_2z_2x_3z_3$

[1]

$((I_{x4}I_{y4} + I_{x4}I_{y3} - I_{x3}I_{x4})m_2m_3m_4{}^2 + ((I_{x4}I_{y4} + I_{x4}I_{y3} - I_{x3}I_{x4})m_2m_3{}^2 +$
$(I_{x4}I_{y4} + I_{x4}I_{y3} - I_{x3}I_{x4})m_2{}^2m_3)m_4)\mathbf{sin}(\theta_2)\mathbf{sin}^2(\theta_3) +$
$((-I_{x4}I_{y4} - I_{x4}I_{y3} + I_{x3}I_{x4})m_2m_3m_4{}^2 +$
$((-I_{x4}I_{y4} - I_{x4}I_{y3} + I_{x3}I_{x4})m_2m_3{}^2 +$
$(-I_{x4}I_{y4} - I_{x4}I_{y3} + I_{x3}I_{x4})m_2{}^2m_3)m_4)\cos(\theta_2)\mathbf{cos}(\theta_3)\mathbf{sin}(\theta_3) +$
$(I_{x3}I_{x4}m_2m_3m_4{}^2 + (I_{x3}I_{x4}m_2m_3{}^2 + I_{x3}I_{x4}m_2{}^2m_3)m_4)\mathbf{sin}(\theta_2)$

[2]

0

[3]

0

x_2 of $x_1y_1x_2y_2z_2x_3y_3z_3$

[1]

$(I_{x4}I_{y4} + I_{x4}I_{y3} - I_{x3}I_{x4})m_2{}^2m_3m_4{}^3\mathbf{sin}(\theta_2)\mathbf{sin}^4(\theta_3) +$
$(-I_{x4}I_{y4} - I_{x4}I_{y3} + I_{x3}I_{x4})m_2{}^2m_3m_4{}^3\cos(\theta_2)\mathbf{cos}(\theta_3)\mathbf{sin}^3(\theta_3) +$
$(((I_{x4}I_{y4} + I_{x4}I_{y3} - I_{x3}I_{x4})m_2m_3{}^2m_4{}^3 +$
$(I_{x4}I_{y4} + I_{x4}I_{y3} - I_{x3}I_{x4})m_2m_3{}^3m_4{}^2)\cos^2(\theta_2) + I_{x3}I_{x4}m_2{}^2m_3m_4{}^3 +$
$(I_{x4}I_{y4} + I_{x4}I_{y3} - I_{x3}I_{x4})m_2{}^2m_3{}^2m_4{}^2)\mathbf{sin}(\theta_2)\mathbf{sin}^2(\theta_3) +$
$(((-I_{x4}I_{y4} - I_{x4}I_{y3} + I_{x3}I_{x4})m_2m_3{}^2m_4{}^3 +$
$(-I_{x4}I_{y4} - I_{x4}I_{y3} + I_{x3}I_{x4})m_2m_3{}^3m_4{}^2)\cos^3(\theta_2) +$
$(-I_{x4}I_{y4} - I_{x4}I_{y3} + I_{x3}I_{x4})m_2{}^2m_3{}^2m_4{}^2\cos(\theta_2))\mathbf{cos}(\theta_3)\mathbf{sin}(\theta_3) +$
$((I_{x3}I_{x4}m_2m_3{}^2m_4{}^3 + I_{x3}I_{x4}m_2m_3{}^3m_4{}^2)\cos^2(\theta_2) +$
$I_{x3}I_{x4}m_2{}^2m_3{}^2m_4{}^2)\mathbf{sin}(\theta_2)$

[2]

0

[3]

0

Table 4.2 (continued)

z_2 of z_2

[1]

$(l_3m_4 + r_3m_3)\sin(\theta_2)\mathbf{sin}(\boldsymbol{\theta}_3) + (-l_3m_4 - r_3m_3)\mathbf{cos}(\boldsymbol{\theta}_3)\cos(\theta_3) +$
$(-l_2m_4 - l_2m_3)\mathbf{cos}(\boldsymbol{\theta}_3)$

[2]

0

[3]

0

z_2 of x_2z_2

[1]

$(r_3{l_3}^2m_3m_4 + (r_3I_{y4} + r_3I_{y3} - r_3I_{x4} - r_3I_{x3})m_3)\sin(\theta_2)\mathbf{sin}^3(\boldsymbol{\theta}_3) +$
$(((-r_3I_{y4} - r_3I_{y3} + r_3I_{x4} + r_3I_{x3})m_3 - r_3{l_3}^2m_3m_4)\mathbf{cos}(\boldsymbol{\theta}_3)\cos(\theta_3) +$
$((-l_2{l_3}^2m_3 - I_{y4}l_2 - I_{y3}l_2 + I_{x4}l_2 + I_{x3}l_2)m_4 +$
$(-I_{y4}l_2 - I_{y3}l_2 + I_{x4}l_2 + I_{x3}l_2)m_3)\mathbf{cos}(\boldsymbol{\theta}_3))\mathbf{sin}^2(\boldsymbol{\theta}_3) +$
$(r_3I_{x4} + r_3I_{x3})m_3\sin(\theta_2)\mathbf{sin}(\boldsymbol{\theta}_3) +$
$((-I_{x4} - I_{x3})l_3m_4 + (-r_3I_{x4} - r_3I_{x3})m_3)\mathbf{cos}(\boldsymbol{\theta}_3)\cos(\theta_3) +$
$((-I_{x4}l_2 - I_{x3}l_2)m_4 + (-I_{x4}l_2 - I_{x3}l_2)m_3)\mathbf{cos}(\boldsymbol{\theta}_3)$

[2]

0

0

z_2 of $x_1y_1x_2z_2$

[1]

$((((r_2 - l_2){l_3}^2 + r_3(2l_2 - 2r_2)l_3 + {r_3}^2(r_2 - l_2))m_2m_3 +$
$(I_{x4}(l_2 - r_2) + I_{x3}(l_2 - r_2) + I_{y4}(r_2 - l_2) + I_{y3}(r_2 - l_2))m_2){m_4}^2 +$
$(((r_2 - l_2){l_3}^2 + r_3(2l_2 - 2r_2)l_3 + {r_3}^2(r_2 - l_2))m_2{m_3}^2 +$
$(((r_2 - l_2){l_3}^2 + r_3(2l_2 - 2r_2)l_3 + {r_3}^2(r_2 - l_2))m_2^2 +$
$(I_{x4}(2l_2 - 2r_2) + I_{x3}(2l_2 - 2r_2) + I_{y4}(2r_2 - 2l_2) +$
$I_{y3}(2r_2 - 2l_2))m_2)m_3 + (I_{x4}(l_2 - r_2) + I_{x3}(l_2 - r_2) +$
$I_{y4}(r_2 - l_2) + I_{y3}(r_2 - l_2))m_2^2)m_4 +$
$(I_{x4}(l_2 - r_2) + I_{x3}(l_2 - r_2) + I_{y4}(r_2 - l_2) + I_{y3}(r_2 - l_2))m_2{m_3}^2 +$
$(I_{x4}(l_2 - r_2) + I_{x3}(l_2 - r_2) + I_{y4}(r_2 - l_2) + I_{y3}(r_2 - l_2))m_2^2m_3)\mathbf{cos}(\boldsymbol{\theta}_3)\sin^2(\theta_3) +$
$((-I_{x4} - I_{x3})l_3m_2{m_4}^2 + (((-I_{x4} - I_{x3})l_3 - r_3I_{x4} - r_3I_{x3})m_2m_3 +$
$(-I_{x4} - I_{x3})l_3m_2^2)m_4 + (-r_3I_{x4} - r_3I_{x3})m_2{m_3}^2 +$
$(-r_3I_{x4} - r_3I_{x3})m_2^2m_3)\mathbf{cos}(\boldsymbol{\theta}_3)\cos(\theta_3) +$
$((I_{x4}(r_2 - l_2) + I_{x3}(r_2 - l_2))m_2{m_4}^2 + ((I_{x4}(2r_2 - 2l_2) + I_{x3}(2r_2 - 2l_2))m_2m_3 +$
$(I_{x4}(r_2 - l_2) + I_{x3}(r_2 - l_2))m_2^2)m_4 + (I_{x4}(r_2 - l_2) + I_{x3}(r_2 - l_2))m_2{m_3}^2 +$
$(I_{x4}(r_2 - l_2) + I_{x3}(r_2 - l_2))m_2^2m_3)\mathbf{cos}(\boldsymbol{\theta}_3)$

[2]

0

[3]

0

Table 4.2 (continued)

z_2 of $x_1y_1x_2z_2x_3z_3$

[1]

$((I_{x3}I_{x4}(l_2-r_2)+I_{x4}I_{y4}(r_2-l_2)+I_{x4}I_{y3}(r_2-l_2))m_2m_3m_4{}^2+$
$((I_{x3}I_{x4}(l_2-r_2)+I_{x4}I_{y4}(r_2-l_2)+I_{x4}I_{y3}(r_2-l_2))m_2m_3{}^2+$
$(I_{x3}I_{x4}(l_2-r_2)+I_{x4}I_{y4}(r_2-l_2)+I_{x4}I_{y3}(r_2-l_2))m_2{}^2m_3)m_4)\mathbf{cos}(\boldsymbol{\theta_2})\sin^2(\theta_3)+$
$((-r_3I_{x3}I_{x4}m_2m_3{}^2-r_3I_{x3}I_{x4}m_2{}^2m_3)m_4-r_3I_{x3}I_{x4}m_2m_3m_4{}^2)\mathbf{cos}(\boldsymbol{\theta_2})\cos(\theta_3)+$
$(I_{x3}I_{x4}(r_2-l_2)m_2m_3m_4{}^2+(I_{x3}I_{x4}(r_2-l_2)m_2m_3{}^2+$
$I_{x3}I_{x4}(r_2-l_2)m_2{}^2m_3)m_4)\mathbf{cos}(\boldsymbol{\theta_2})$

[2]

0

[3]

0

z_2 of $x_1y_1x_2y_2z_2x_3y_3z_3$

[1]

$(I_{x3}I_{x4}(l_2-r_2)+I_{x4}I_{y4}(r_2-l_2)+$
$I_{x4}I_{y3}(r_2-l_2))m_2{}^2m_3m_4{}^3\mathbf{cos}(\boldsymbol{\theta_2})\sin^4(\theta_3)+$
$(-r_3I_{x3}I_{x4}m_2{}^2m_3m_4{}^3\mathbf{cos}(\boldsymbol{\theta_2})\cos(\theta_3)+((I_{x3}I_{x4}(l_2-r_2)+I_{x4}I_{y4}(r_2-l_2)+$
$I_{x4}I_{y3}(r_2-l_2))m_2m_3{}^2m_4{}^3+(I_{x3}I_{x4}(l_2-r_2)+I_{x4}I_{y4}(r_2-l_2)+$
$I_{x4}I_{y3}(r_2-l_2))m_2m_3{}^3m_4{}^2)\mathbf{cos}^3(\boldsymbol{\theta_2})+$
$(I_{x3}I_{x4}(r_2-l_2)m_2{}^2m_3m_4{}^3+(I_{x3}I_{x4}(l_2-r_2)+I_{x4}I_{y4}(r_2-l_2)+$
$I_{x4}I_{y3}(r_2-l_2))m_2{}^2m_3{}^2m_4{}^2)\mathbf{cos}(\boldsymbol{\theta_2}))\sin^2(\theta_3)+$
$((-r_3I_{x3}I_{x4}m_2m_3{}^2m_4{}^3-r_3I_{x3}I_{x4}m_2m_3{}^3m_4{}^2)\mathbf{cos}^3(\boldsymbol{\theta_2})-$
$r_3I_{x3}I_{x4}m_2{}^2m_3{}^2m_4{}^2\mathbf{cos}(\boldsymbol{\theta_2}))\cos(\theta_3)+(I_{x3}I_{x4}(r_2-l_2)m_2m_3{}^2m_4{}^3+$
$I_{x3}I_{x4}(r_2-l_2)m_2m_3{}^3m_4{}^2)\mathbf{cos}^3(\boldsymbol{\theta_2})+I_{x3}I_{x4}(r_2-l_2)m_2{}^2m_3{}^2m_4{}^2\mathbf{cos}(\boldsymbol{\theta_2})$

[2]

0

[3]

0

Table 4.2 (continued)

x_3 of x_3

[1]

$I_{x4}\mathbf{cos}(\theta_2)\mathbf{sin}(\theta_3) + I_{x4}\mathbf{sin}(\theta_2)\mathbf{cos}(\theta_3)$

[2]

0

[3]

0

x_3 of x_3z_3

[1]

$I_{x4}m_4\mathbf{cos}(\theta_2)\mathbf{sin}(\theta_3) + I_{x4}m_4\mathbf{sin}(\theta_2)\mathbf{cos}(\theta_3)$

[2]

0

[3]

0

x_3 of $x_1y_1x_2z_2x_3z_3$

[1]

$((I_{x4}I_{y4} + I_{x4}I_{y3})m_2m_3{m_4}^2 + ((I_{x4}I_{y4} + I_{x4}I_{y3})m_2{m_3}^2 +$
$(I_{x4}I_{y4} + I_{x4}I_{y3}){m_2}^2m_3)m_4)\mathbf{cos}(\theta_2)\mathbf{sin}(\theta_3)$

[2]

0

[3]

0

x_3 of $x_1y_1x_2y_2z_2x_3y_3z_3$

[1]

$(I_{x4}I_{y4} + I_{x4}I_{y3}){m_2}^2m_3{m_4}^3\mathbf{cos}(\theta_2)\mathbf{sin}^3(\theta_3) +$
$(((I_{x4}I_{y4} + I_{x4}I_{y3})m_2{m_3}^2{m_4}^3 + (I_{x4}I_{y4} + I_{x4}I_{y3})m_2{m_3}^3{m_4}^2)\mathbf{cos}^3(\theta_2) +$
$(I_{x4}I_{y4} + I_{x4}I_{y3}){m_2}^2{m_3}^2{m_4}^2\mathbf{cos}(\theta_2))\mathbf{sin}(\theta_3)$

[2]

0

[3]

Table 4.2 (continued)

z_3 of z_3

[1]

$l_3 m_4 \sin(\theta_2)\sin(\theta_3) - l_3 m_4 \cos(\theta_2)\cos(\theta_3) - l_2 m_4 \cos(\theta_2)$

[2]

0

[3]

0

z_3 of $x_3 z_3$

[1]

$I_{x4} l_3 m_4 \sin(\theta_2)\sin(\theta_3) - I_{x4} l_3 m_4 \cos(\theta_2)\cos(\theta_3) - I_{x4} l_2 m_4 \cos(\theta_2)$

[2]

0

[3]

0

z_3 of $x_1 y_1 x_2 z_2 x_3 z_3$

[1]

$(I_{x3} I_{x4}(r_3 - l_3) m_2 m_3 {m_4}^2 + (I_{x3} I_{x4}(r_3 - l_3) m_2 {m_3}^2 +$
$I_{x3} I_{x4}(r_3 - l_3) {m_2}^2 m_3) m_4) \cos(\theta_2)\cos(\theta_3)$

[2]

0

[3]

0

z_3 of $x_1 y_1 x_2 y_2 z_2 x_3 y_3 z_3$

[1]

$I_{x3} I_{x4}(r_3 - l_3) {m_2}^2 m_3 {m_4}^3 \cos(\theta_2)\cos(\theta_3)\sin^2(\theta_3) +$
$((I_{x3} I_{x4}(r_3 - l_3) m_2 {m_3}^2 {m_4}^3 + I_{x3} I_{x4}(r_3 - l_3) m_2 {m_3}^3 {m_4}^2)\cos^3(\theta_2) +$
$I_{x3} I_{x4}(r_3 - l_3) {m_2}^2 {m_3}^2 {m_4}^2 \cos(\theta_2))\cos(\theta_3)$

[2]

0

[3]

0

4.4 Controllability of Inaccessible Vibratory Modes

It is well known that for a linear, time invariant system

$$\dot{x} = Ax + Bu \in \mathcal{R}^n \tag{4.33}$$

the global controllability can be examined by the rank of the controllability matrix

$$\Psi = \left[B\ AB\ A^2B\ \cdots\ A^{n-1}B\right] \tag{4.34}$$

If Ψ has a rank n then the control space of u covers $\mathcal{R}^n$ and every desired state could be reached in finite time. However, nonlinear system controllability can only be examined point by point (local controllability). Local controllability indicates the reachability of a local point by any nearby point inside a small region surrounding that local point. If every local point is controllable and the union of all small reachable regions covers the whole state space, then the global controllability of nonlinear system is ensured. According to Kalman's discussion in [Markus and Lee, 1962], for a nonlinear system

$$\dot{x} = f(t, x, u) \tag{4.35}$$

the local controllability at (x_o, u_o) can be detected by linearizing the nonlinear equation around the given point and then checking the controllability of the linear equation. This method will be used here to investigate the controllability of inaccessible modes. The results will show the distinction between accessibility and controllability of vibratory modes. Recalling that compliant manipulator system control equation is defined by

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} A_1 & C^T \\ C & A_2 \end{bmatrix} \begin{bmatrix} -f_1 \\ -f_2 - K\beta \end{bmatrix} + \begin{bmatrix} A_1 \\ C \end{bmatrix} u \tag{4.36}$$

Let

$$x_1 = \begin{bmatrix} \theta \\ \beta \end{bmatrix} \in \mathcal{R}^{n_\theta + n_\beta};\ x_2 = \begin{bmatrix} \dot{\theta} \\ \dot{\beta} \end{bmatrix} \in \mathcal{R}^{n_\theta + n_\beta};\ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathcal{R}^{2(n_\theta + n_\beta)} \tag{4.37}$$

and also set

$$h(x_1, x_2) = \begin{bmatrix} A_1 & C^T \\ C & A_2 \end{bmatrix} \begin{bmatrix} -f_1 \\ -f_2 - K\beta \end{bmatrix} \in \mathcal{R}^{n_\theta + n_\beta} \tag{4.38}$$

and

$$W(x_1) = \begin{bmatrix} A_1 \\ C \end{bmatrix} \tag{4.39}$$

then Equation 4.36 could be written in a state space form as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= h(x_1, x_2) + W(x_1)u \end{aligned} \tag{4.40}$$

whose variational equation around an operational point (x_1, x_2, u) is

$$\begin{aligned} \delta\dot{x}_1 &= \delta x_2 \\ \delta\dot{x}_2 &= \delta h(x_1, x_2) + \delta(W(x_1)u) \end{aligned} \tag{4.41}$$

By taking the first order approximation, the variational equation has a linear form

$$\begin{aligned} \delta\dot{x}_1 &= \delta x_2 \\ \delta\dot{x}_2 &= \Phi_1\delta x_1 + \Phi_2\delta x_2 + W(x_1)\delta u \end{aligned} \tag{4.42}$$

where

$$\begin{aligned} \Phi_1 &= \left[\frac{\partial h(x_1, x_2)}{\partial x_1}\right] + \left[\frac{\partial W(x_1)}{\partial x_1}u\right] \in \mathcal{R}^{(n_\theta+n_\beta)\times(n_\theta+n_\beta)} \\ \Phi_2 &= \left[\frac{\partial h(x_1, x_2)}{\partial x_2}\right] \in \mathcal{R}^{(n_\theta+n_\beta)\times(n_\theta+n_\beta)} \end{aligned} \tag{4.43}$$

whose elements could be expressed in a more detailed form as

$$\begin{aligned} \left[\frac{\partial h}{\partial x_1}\right] &= \begin{bmatrix} \frac{\partial h_1}{\partial\theta_1} & \frac{\partial h_1}{\partial\theta_2} & \cdots & \frac{\partial h_1}{\partial\theta_{n_\theta}} & \frac{\partial h_1}{\partial\beta_1} & \frac{\partial h_1}{\partial\beta_2} & \cdots & \frac{\partial h_1}{\partial\beta_{n_\beta}} \\ \frac{\partial h_2}{\partial\theta_1} & \frac{\partial h_2}{\partial\theta_2} & \cdots & \frac{\partial h_2}{\partial\theta_{n_\theta}} & \frac{\partial h_2}{\partial\beta_1} & \frac{\partial h_2}{\partial\beta_2} & \cdots & \frac{\partial h_2}{\partial\beta_{n_\beta}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ \frac{\partial h_{n_\theta+n_\beta}}{\partial\theta_1} & \frac{\partial h_{n_\theta+n_\beta}}{\partial\theta_2} & \cdots & \frac{\partial h_{n_\theta+n_\beta}}{\partial\theta_{n_\theta}} & \frac{\partial h_{n_\theta+n_\beta}}{\partial\beta_1} & \frac{\partial h_{n_\theta+n_\beta}}{\partial\beta_2} & \cdots & \frac{\partial h_{n_\theta+n_\beta}}{\partial\beta_{n_\beta}} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\partial h}{\partial\theta_1} & \frac{\partial h}{\partial\theta_2} & \cdots & \frac{\partial h}{\partial\theta_{n_\theta}} & \frac{\partial h}{\partial\beta_1} & \frac{\partial h}{\partial\beta_2} & \cdots & \frac{\partial h}{\partial\beta_{n_\beta}} \end{bmatrix} \end{aligned} \tag{4.44}$$

where h_i is the ith element of vector h, similarly

$$\left[\frac{\partial W}{\partial x_1}u\right] = \left[\begin{array}{cccccccc} \frac{\partial W}{\partial \theta_1}u & \frac{\partial W}{\partial \theta_2}u & \cdots & \frac{\partial W}{\partial \theta_{n_\theta}}u & \frac{\partial W}{\partial \beta_1}u & \frac{\partial W}{\partial \beta_2}u & \cdots & \frac{\partial W}{\partial \beta_{n_\beta}}u \end{array}\right] \tag{4.45}$$

and

$$\left[\frac{\partial h}{\partial x_2}\right] = \left[\begin{array}{cccccccc} \frac{\partial h}{\partial \theta_1} & \frac{\partial h}{\partial \theta_2} & \cdots & \frac{\partial h}{\partial \theta_{n_\theta}} & \frac{\partial h}{\partial \beta_1} & \frac{\partial h}{\partial \beta_2} & \cdots & \frac{\partial h}{\partial \beta_{n_\beta}} \end{array}\right] \tag{4.46}$$

With these defined notations, Equation 4.42 could be expressed as a linear state equation

$$\begin{aligned} \delta\dot{x} &= \left[\begin{array}{cc} 0 & \mathcal{I} \\ \Phi_1 & \Phi_2 \end{array}\right]\delta x + \left[\begin{array}{c} 0 \\ W \end{array}\right]\delta u \\ &= A\delta x + B\delta u \end{aligned} \tag{4.47}$$

where $\mathcal{I}$ is an $(n_\theta + n_\beta) \times (n_\theta + n_\beta)$ identity matrix, $A \in \mathcal{R}^{2(n_\theta+n_\beta)\times 2(n_\theta+n_\beta)}$, and $B \in \mathcal{R}^{2(n_\theta+n_\beta)\times n_\theta}$.

From Equations 4.34 and 4.47, the controllability matrix of the linearized system is given by

$$\begin{aligned} \Psi &= \left[B\ AB\ A^2B\ \cdots\ A^{2(n_\theta+n_\beta)-1}B\right] \\ &= \left[\begin{array}{cccc} 0 & W & \Phi_2 W & (\Phi_1+\Phi_2^2)W & \cdots \\ W & \Phi_2 W & (\Phi_1+\Phi_2^2)W & (\Phi_1\Phi_2+\Phi_2\Phi_1+\Phi_2^3)W & \cdots \end{array}\right] \end{aligned} \tag{4.48}$$

whose rank serves as an indicator of the local controllability of the original nonlinear system in Equation 4.36. Since Φ_1 and Φ_2 are functions of θ, β, $\dot{\theta}$, $\dot{\beta}$, and u, and W is a function of θ and β, the rank of Ψ relies on the specific operational point (x_o, u_o) around which the system is linearized. In the following sections, two examples will be presented. The first example is a one-link arm modeled with one lumped lateral deflection, where due to the specific kinematic structure the vibratory mode is always inaccessible and uncontrollable. The second example is a two-link arm modeled with one lumped lateral vibratory mode which turns out to be controllable even in an

inaccessible position. Both examples are studied by the symbolic program MACSYMA, and due to computer memory limitations, some simplifications are applied in order to obtain the final results.

4.4.1 Example 1

The first example model is shown in Figure 4.4 (a), in which (X, Y, Z) is the global coordinate frame, and (X_1, Y_1, Z_1) is the local frame of the arm. The link has mass m_1 and moment of inertia I_1. The mass is located at a distance r from the joint axis, and the link length is l. A payload is added to the end of the link, which has mass m_2 and moment of inertia I_2. The vibratory mode is modeled at the end of the arm and parallel to Z_1 direction. Then the dynamic equations are described by

$$\begin{aligned} (I_1 + I_2 + m_1 r^2 + m_2 l^2)\ddot{\theta} &= u \\ m_2\ddot{\beta} + K\beta &= 0 \end{aligned} \tag{4.49}$$

where K is the modeled spring stiffness. Letting $a = (I_1 + I_2 + m_1 r^2 + m_2 l^2)$, the inverse of the inertial matrix is expressed as

$$\begin{bmatrix} A_1 & C^T \\ C & A_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{a} & 0 \\ 0 & \frac{1}{m_2} \end{bmatrix} \tag{4.50}$$

which indicates that the 1×1 dimensional C matrix is given by $C = 0$. This suggests that oscillation β always remains inaccessible. Since the derived dynamic equations are linear and time invariant, they could be expressed in a state space form as

$$\begin{bmatrix} \dot{\theta} \\ \dot{\beta} \\ \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & -\frac{K}{m_2} & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \beta \\ \dot{\theta} \\ \dot{\beta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{a} \\ 0 \end{bmatrix} u \tag{4.51}$$

and the controllability matrix is

$$\Psi = \begin{bmatrix} 0 & \frac{1}{a} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{a} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.52}$$

which possesses only two independent columns so it has a rank of two. Since Ψ does not have a full rank, the system is uncontrollable. Hence in this example, the controller loses both controllability and accessibility of the vibratory mode.

4.4.2 Example 2

The second example is depicted in Figure 4.4 (b), which is a two-link arm modeled with one vibratory mode. The first link has a local frame (X_1, Y_1, Z_1), and that of the second link is (X_2, Y_2, Z_2). The second link is oscillating laterally along Z_2 direction. Each link length is indicated in the figure. For simplicity in analysis, the links are assumed massless, only the payload is considered to have a mass m_3 and a moment of inertia I_3 around the X_2 axis. Then the system dynamics is given by

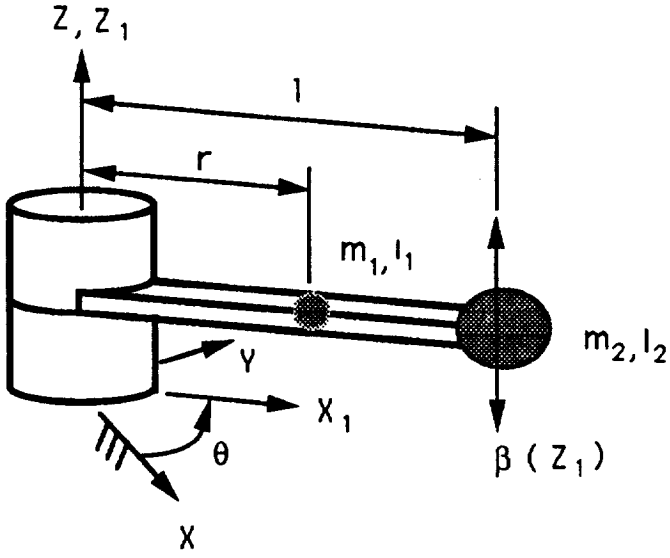
$$I^* \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\beta} \end{bmatrix} + f = \begin{bmatrix} u_1 \\ u_2 \\ 0 \end{bmatrix} \tag{4.53}$$

or

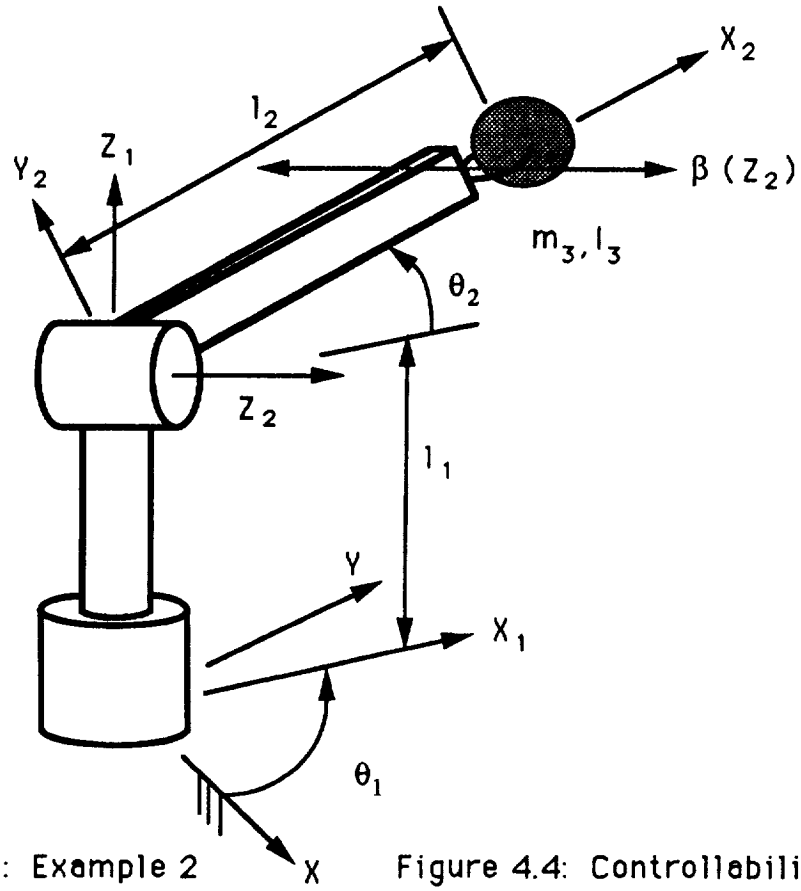
$$\begin{aligned} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\beta} \end{bmatrix} &= (I^*)^{-1}(-f) + (I^*)^{-1}u \\ &= h + Wu \end{aligned} \tag{4.54}$$

in which I^* is the generalized inertia matrix which is symbolically derived as

$$I^* = \begin{bmatrix} I_3 + m_3\beta^2 + (m_3 l_2^3 - I_3)\cos^2(\theta_2) & -m_3 l_2 \sin(\theta_2)\beta & -m_3 l_2 \cos(\theta_2) \\ -m_3 l_2 \sin(\theta_2)\beta & m_3 l_2^2 & 0 \\ -m_3 l_2 \cos(\theta_2) & 0 & m_3 \end{bmatrix}$$



(a) : Example 1



(b) : Example 2

Figure 4.4: Controllability Models

and f contains the Coriolis, centrifugal, gravitational, and spring force effects, and is described by

$$f = \begin{bmatrix} -2(m_3 l_2^2 - I_3)\cos(\theta_2)\sin(\theta_2)\dot{\theta}_1\dot{\theta}_2 - m_3 l_2 \cos(\theta_2)\beta\dot{\theta}_2^2 + 2m_3\beta\dot{\theta}_1\dot{\beta} \\ (m_3 l_2^2 - I_3)\cos(\theta_2)\sin(\theta_2)\dot{\theta}_1^2 - 2m_3 l_2 \sin(\theta_2)\dot{\theta}_1\beta + m_3 g l_2 \cos(\theta_2) \\ -m_3\beta\dot{\theta}_1^2 + 2m_3 l_2 \sin(\theta_2)\dot{\theta}_1\dot{\theta}_2 + K\beta \end{bmatrix}$$

where g is the gravitational acceleration, and K is the modeled spring stiffness. The inaccessible position of the modeled vibratory mode is determined from the submatrix C of $(I^*)^{-1}$, which has a symbolic form

$$\begin{aligned} C &= \frac{1}{\Delta}\left[m_3^2 l_2^3 \cos(\theta_2) \quad m_3^2 l_2^2 \cos(\theta_2)\sin(\theta_2)\beta\right] \\ \Delta &= (m_3^2 I_3 l_2^2)\sin^2(\theta_2) + (m_3^3 l_2^2 \beta^2)\cos^2(\theta_2) \end{aligned} \tag{4.55}$$

Apparently, $\cos(\theta_2) = 0$ gives the inaccessible position, i.e., when the second link is vertical and collinear with the first joint, or $\theta_2 = 90$ degrees, both actuator inputs can not access $\ddot{\beta}$. To check the local controllability in this inaccessible position, the nonlinear system defined in Equation 4.54 will be linearized around an operational state described by

$$\{\theta_1, \theta_2, \beta, \dot{\theta}_1, \dot{\theta}_2, \dot{\beta}, u_1, u_2\} = \{\theta_{1o}, 90^o, 0, \dot{\theta}_{1o}, \dot{\theta}_{2o}, \dot{\beta}_o, u_{1o}, u_{2o}\}$$

where subscript o denotes the constant values of a selected operational state, and θ_2 is chosen to be 90 degrees to include the inaccessibility condition. The linear variational equation $\delta\dot{x} = A\delta x + B\delta u$ of the nonlinear system is derived symbolically around the operational point and the matrices A and B take the following form

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 2\dot{\theta}_{1o}\dot{\theta}_{2o} & \frac{u_{2o}}{I_3 l_2} & 0 & 0 & 0 \\ 0 & \frac{(m_3 l_2^2 - I_3)\dot{\theta}_{1o}^2 + m_3 g l_2}{m_3 l_2^2} & \frac{u_{1o}}{I_3 l_2} & \frac{2\dot{\beta}_o}{l_2} & 0 & \frac{2\dot{\theta}_{1o}}{l_2} \\ 0 & -\frac{u_{1o} l_2}{I_3} & \frac{m_3\dot{\theta}_{1o}^2 - K}{m_3} & -2l_2\dot{\theta}_{2o} & -2l_2\dot{\theta}_{1o} & 0 \end{bmatrix} \tag{4.56}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{I_3} & 0 \\ 0 & \frac{1}{m_3 l_2^2} \\ 0 & 0 \end{bmatrix} \tag{4.57}$$

A controllability matrix Ψ is formed by using the above A and B. In this case, $n_\theta = 2$ and $n_\beta = 1$, so Ψ is a 6×12 matrix. To check the full rank of Ψ, the determinants of two submatrices of Ψ are derived, which are

$$\det[B\ AB\ A^2B] = \frac{2\dot{\theta}_{2o}u_{1o}}{m_3^3 I_3^4 l_2^4} \tag{4.58}$$

and

$$\det[B\ A^2B\ A^3B] = \frac{16\dot{\theta}_{1o}^2\dot{\theta}_{2o}^2}{m_3^3 I_3^3 l_2^4}\left[\frac{\dot{\theta}_{2o}u_{1o}}{I_3} - \frac{(m_3\dot{\theta}_{1o}^2 - K)\dot{\beta}_o}{m_3 l_2}\right] \tag{4.59}$$

since these determinants can have nonzero values, Ψ can have a full rank. Therefore, the nonlinear system is locally controllable around an inaccessible position. This result indicates that inaccessibility does not necessarily imply uncontrollability. But as pointed out before, in an inaccessible position the vibratory mode is governed by nonlinear terms composed of centrifugal, Coriolis, gravitational, and spring forces. Without the direct access of the control input, it is very difficult to dampen the vibrations through the nonlinear term f_2 and spring force $K\beta$.

Chapter 5

Controller Design for Compliant Manipulators with Well-Known System Parameters

In a linear system of equations $Ax = y$, where $A \in \mathcal{R}^{n_x \times n_y}$, $x \in \mathcal{R}^{n_x}$, and $y \in \mathcal{R}^{n_y}$, the matrix A assigns a y for a given x. Let $\mathcal{X} \subset \mathcal{R}^{n_x}$ be the set of all such x and $\mathcal{Y} \subset \mathcal{R}^{n_y}$ be the set of corresponding y, then $\mathcal{X}$ is defined as the domain of A and $\mathcal{Y}$ is the range of A. If the mapping from $\mathcal{X}$ to $\mathcal{Y}$ is bijective, i.e., one-to-one and $\mathcal{Y} = \mathcal{R}^{n_y}$, then an inverse map exists such that for every element $y \in \mathcal{R}^{n_y}$ there is a unique solution $x \in \mathcal{X}$ satisfying the equation. Since A is generally not a full-rank square matrix, existence and uniqueness of solution x is not guaranteed for any y. For example, if $n_x > n_y$, let $\mathcal{X}^\perp \subset \mathcal{X}$ be a subset defined as $\mathcal{X}^\perp = \{x^\perp : Ax^\perp = 0;\ x^\perp \in \mathcal{X}\}$, i.e., $\mathcal{X}^\perp$ is the nullspace of A, then for every element $x^* = x + x^\perp$, $x \in \mathcal{X}$ and $x^\perp \in \mathcal{X}^\perp$, $Ax^* = y$, which means that for a given y the solution is indecisive. Another example may be given as follows: if $n_x < n_y$ and if we let $\mathcal{Y}^\perp \subset \mathcal{R}^{n_y}$ denote the complement of $\mathcal{Y}$, then for a given $y^* = y + y^\perp$, $y \in \mathcal{Y}$ and $y^\perp \in \mathcal{Y}^\perp$, there is no solution for the equation $Ax = y^*$, which means that since A has a rank smaller than n_y its column space can not span $\mathcal{R}^{n_y}$, hence solution does not exit unless y^* is in the column space of A. The first example is sometimes referred as redundant problem, and the second example is overdetermined problem. It will be shown in this chapter that the controller design of compliant manipulators has the nature of solving

an overdetermined problem, but after choosing proper control criteria the controller design becomes solving a redundant problem.

5.1 The Difficulty of Ideal Acceleration Assignment

According to the third chapter results, compliant manipulator system dynamics have a common symbolic form

$$\begin{bmatrix} \Lambda_1 & \Sigma \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 + K\beta \end{bmatrix} = \begin{bmatrix} u \\ 0 \end{bmatrix} \tag{5.1}$$

for both distributed and lumped parameter models, in which the first matrix represents the generalized inertia associated with nominal and vibratory modes, and its inverse is defined in Equation 4.7 as

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}^{-1} = \begin{bmatrix} A_1 & C^T \\ C & A_2 \end{bmatrix} \tag{5.2}$$

Premultiplying Equation 5.1 by the inverse above, the dynamic equations become

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \left(\begin{bmatrix} A_1 \\ C \end{bmatrix} f_1 + \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \right) = \begin{bmatrix} A_1 \\ C \end{bmatrix} u \tag{5.3}$$

Ideally, it is desirable to select a proper u so that the accelerations obtain the PID state feedback and feedforward values defined by the following relations.

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{5.4}$$

where

$$\begin{aligned} u_1 &= \ddot{\theta}_r + K_{v\theta}(\dot{\theta} - \dot{\theta}_r) + K_{p\theta}(\theta - \theta_r) + K_{I\theta}\left\{ - \int (\theta - \theta_r) dt \right\} \\ u_2 &= \ddot{\beta}_r + K_{v\beta}(\dot{\beta} - \dot{\beta}_r) + K_{p\beta}(\beta - \beta_r) + K_{I\beta}\left\{ - \int (\beta - \beta_r) dt \right\} \end{aligned} \tag{5.5}$$

in which θ_r and β_r represent the desired states, and K_{vi}, K_{pi}, and K_{Ii}, $i = \theta$, β, are separately the stable velocity, position, and integral gains with

appropriate dimensions. Generally, θ_r is predefined by given task, and β_r and its time derivatives are chosen to be zero in order to eliminate vibrations. Additionally, the feedback gain matrices K_{vi}, K_{pi}, and K_{Ii} are diagonal which decouple Equation 5.5. If the acceleration assignment is achieved then the combination of Equations 5.4 and 5.5 produces

$$\begin{aligned}(\ddot{\theta}-\ddot{\theta}_r)-K_{v\theta}(\dot{\theta}-\dot{\theta}_r)-K_{p\theta}(\theta-\theta_r)+K_{I\theta}\left\{\int(\theta-\theta_r)dt\right\}=0\\(\ddot{\beta}-\ddot{\beta}_r)-K_{v\beta}(\dot{\beta}-\dot{\beta}_r)-K_{p\beta}(\beta-\beta_r)+K_{I\beta}\left\{\int(\beta-\beta_r)dt\right\}=0\end{aligned}\qquad(5.6)$$

which indicates that by choosing stable feedback gains nominal joints will track the given task trace, i.e., $\theta \rightarrow \theta_r$, and structural oscillation will be removed at the same time, i.e., $\beta \rightarrow \beta_r = 0$. However in Equation 5.3, both acceleration and nonlinear terms in the left-hand side of the equation are in $\mathcal{R}^{n_\theta+n_\beta}$ space, but the control space of u is composed by the n_θ column vectors of $[A_1^T\ C^T]^T$ which covers only a portion of $\mathcal{R}^{n_\theta+n_\beta}$. So for a given desired acceleration $[u_1^T\ u_2^T]^T$, its sum with the second nonlinear term in Equation 5.3 might not reside in the control space of u, which means that by defining

$$y=\begin{bmatrix}u_1\\u_2\end{bmatrix}+\left(\begin{bmatrix}A_1\\C\end{bmatrix}f_1+\begin{bmatrix}C^T\\A_2\end{bmatrix}(f_2+K\beta)\right)\qquad(5.7)$$

$$A=\begin{bmatrix}A_1\\C\end{bmatrix}\qquad(5.8)$$

and setting $x = u$, finding an ideal acceleration assignment u is equivalent to solving an overdetermined equation $Ax = y$. Obviously, a solution x exists only when y is in the range of A, which is generally difficult to verify for a moving robot. Therefore, direct acceleration assignment is not practical for the control of compliant manipulators due to the dimensional mismatch between the number of modeled degrees of freedom and the available actuators. By contrast, in the control of rigid manipulators where structural compliance

is neglected, every joint, or degree of freedom, is accompanied by an actuator, so direct acceleration assignment is possible after nonlinear compensation of Coriolis, centrifugal, and gravitational forces. Unfortunately, such nonlinear compensation techniques can not be extended directly to compliant manipulators where additional vibratory modes are added to system motion description. Hence, dimensional mismatch makes the control of compliant manipulators a difficult and therefore challenging task.

5.2 The Theorems of the Lyapunov's Second Method

Since direct acceleration assignment is not applicable in the control of compliant manipulators, certain stability criteria must be adopted to facilitate the controller design. The stability criteria chosen in the following controller designs are derived by Lyapunov, which are stated in the following theorems [Landau, 1979].

Theorem 5.1 (Lyapunov) *Consider the free dynamic system*

$$\dot{\underline{x}} = \underline{f}(\underline{x}, t) \tag{5.9}$$

where $\underline{f}(\underline{0}, t) = \underline{0}$ *for all* t. *If there exits a real scalar function* $V(\underline{x}, t)$ *with continuous first partial derivatives with respect to* $\underline{x}$ *and* t *such that*

1. $V(\underline{0}, t) = 0$ *for all* t
2. $V(\underline{x}, t) \geq \alpha(\|\underline{x}\|) > 0$ *for all* $\underline{x} \neq \underline{0}$, $\underline{x} \in \mathcal{R}^n$, *and for all* t, *where* $\alpha(\cdot)$ *is a real, continuous, nondecreasing scalar function such that* $\alpha(0) = 0$
3. $V(\underline{x}, t) \to \infty$ *as* $\|\underline{x}\| \to \infty$ *for all* t
4. $\dot{V} = \frac{d}{dt}V(\underline{x}, t) = \frac{\partial}{\partial t}V + (\mathrm{grad}V)^T \underline{f}(\underline{x}, t) \leq -\gamma(\|\underline{x}\|) < 0$ *where* $\gamma(\cdot)$ *is a real, continuous, scalar function such that* $\gamma(0) = 0$

then the equilibrium state $\underline{x}_e = \underline{0}$ *is globally asymptotically stable, and* $V(\underline{x}, t)$ *is a Lyapunov function for this system.*

Corollary 5.1 *The equilibrium state* $\underline{x}_e = \underline{0}$ *of the autonomous dynamic system*

$$\dot{\underline{x}} = \underline{f}(\underline{x})$$

is globally asymptotically stable if there exists a real scalar function $V(\underline{x})$ *with continuous first partial derivatives with respect to* $\underline{x}$ *such that*

1. $V(\underline{0}) = 0$
2. $V(\underline{x}) > 0$ *for all* $\underline{x} \neq 0$, $\underline{x} \in \mathcal{R}^n$
3. $V(\underline{x}) \to \infty$ *as* $\|\underline{x}\| \to \infty$
4. $\dot{V} = \frac{d}{dt}V(\underline{x}) < 0$ *for all* $\underline{x} \neq 0$, $\underline{x} \in \mathcal{R}^n$

Corollary 5.2 *In the above Corollary, condition 4 may be replaced by*

4.1 $\dot{V}(\underline{x}) \leq 0$ *for all* $\underline{x} \neq 0$, $\underline{x} \in \mathcal{R}^n$

4.2 $\dot{V}(\underline{\phi}(t; \underline{x}_0, t_0))$ *does not vanish identically in* $t \geq t_0$ *for any* t_0 *and* $\underline{x}_0 \neq 0$, *where* $\underline{\phi}(t; \underline{x}_0, t_0)$ *is a solution of Equation 5.9 and* $\underline{\phi}(t_0; \underline{x}_0, t_0) = \underline{x}_0$

Finally, for linear time-invariant free dynamic system, Lyapunov provides the following theorem giving the necessary and sufficient conditions for globally asymptotically stability.

Theorem 5.2 *The equilibrium state* $\underline{x}_e = \underline{0}$ *of a linear time-invariant free dynamic system*

$$\dot{\underline{x}} = A\underline{x} \tag{5.10}$$

is (globally) asymptotically stable if and only if given any positive definite matrix Q there exists a symmetric positive definite matrix P which is the unique solution of the matrix equation

$$A^T P + PA = -Q$$

and $V = \underline{x}^T P \underline{x}$ is a Lyapunov function for the system in Equation 5.10.

In applying the Lyapunov's second method, a continuous, positive definite scalar function, or the Lyapunov function, is defined first. This positive Lyapunov function generally represents a distance, or error, between an instant state and the desired state. Then by formulating the system controller properly, the Lyapunov function produces a negative rate as long as it remains a positive value, which means that the distance and hence the state error is reduced continuously until a zero error is met. Since this method studies stability problem in $\mathcal{R}^1$ space, it is very useful to solve multi-degree-of-freedom control problems like the control of compliant manipulators. It will be shown later that by using the Lyapunov's second method the control design becomes solving the redundant problem instead of the overdetermined problem in the direct acceleration assignment.

Several controller structures will be introduced in the following sections. Before presenting these control algorithms, it should be noticed that to maintain generality of the results the nonlinear nature of system dynamics is considered in the design process. No linearization or ignoring nonlinear term is assumed to simplify the control design problem. However, two assumptions are used in building the following controllers, which are: (1) system parameters including payload are well-known, but in later example simulations payload uncertainty are added to test controller robustness, and (2) all

nominal and vibrational displacement and velocity states are available on-line. The first assumption is generally true for a well calibrated system. As for the second assumption, robot joint position and velocity could be read from attached resolver and tachometers directly, and modal amplitudes and velocities could be measured and reconstructed from strain-gauge readings. Therefore the second assumption is technically feasible. Examples of on-line measurement of vibratory states for the motion control of compliant arm have been reported by [Hasting and Book, 1986] and [Cannon and Schmitz, 1984].

5.3 Orthogonal Projection Method

The first controller is designed by the orthogonal projection method. As mentioned in the direct acceleration assignment, finding a proper input to produce the ideal acceleration response is solving a linear equation described by $Ax = y$, where A has more rows than columns. If A has a full column rank then $(A^TA)^{-1}$ exists, and there is a left-inverse $(A^TA)^{-1}A^T$ such that $x = (A^TA)^{-1}A^Ty$ which represents an approximated solution with minimum error

$$\|y - Ax\|_{\min} = \|(\mathcal{I} - A(A^TA)^{-1}A^T)y\|$$

where $\mathcal{I}$ is an identity matrix and $A(A^TA)^{-1}A^T$ is a projection matrix from y to the column space of A. Because

$$A^T(y - Ax) = 0$$

the solution $x = (A^TA)^{-1}A^Ty$ could be considered geometrically as an orthogonal projection of y on the column space of A. Since the orthogonal projection solution x can not generate the exact y, it could not be used alone to construct the input command. However, the orthogonal projection matrix

$(A^TA)^{-1}A^T$ is a linear map from $\mathcal{R}^{n_\theta+n_\beta}$ space to $\mathcal{R}^{n_\theta}$ space, which allows us to transfer control design from $\mathcal{R}^{n_\theta}$ space to $\mathcal{R}^{n_\theta+n_\beta}$ space where the ideal acceleration $[u_1^T, u_2^T]^T$ resides. After constructing the controller in $\mathcal{R}^{n_\theta+n_\beta}$ space, the result is projected back to $\mathcal{R}^{n_\theta}$ space to obtain the final input command. Therefore, orthogonal projection will be used as a starting point in the following design process. Since the orthogonal projection does not guarantee the ideal acceleration assignment, the Lyapunov's second method will be employed as the criteria to build a stable controller in $\mathcal{R}^{n_\theta+n_\beta}$ space. First, the dynamic equations defined in Equation 5.3 are rewritten as

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} A_1 \\ C \end{bmatrix} u - \left(\begin{bmatrix} A_1 \\ C \end{bmatrix} f_1 + \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \right) \tag{5.11}$$

then by selecting a composite input command

$$u = f_1 + u_3 \tag{5.12}$$

Equation 5.11 is reduced to

$$\begin{aligned} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} &= \begin{bmatrix} A_1 \\ C \end{bmatrix} u_3 - \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \\ &= \begin{bmatrix} A_1 \\ C \end{bmatrix} u_3 + \gamma_1 \end{aligned} \tag{5.13}$$

where γ_1 is defined as

$$\gamma_1 = - \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \tag{5.14}$$

Since submatrix A_1 has full rank, then the matrix α defined by

$$\begin{aligned} \alpha &= [A_1^T \, C^T] \begin{bmatrix} A_1 \\ C \end{bmatrix} \\ &= A_1^T A_1 + C^T C \end{aligned} \tag{5.15}$$

is invertible. So, it is possible to choose a set of input $[u_4^T\ u_5^T]^T$ defined in $\mathcal{R}^{n_\theta+n_\beta}$ space that produces a u_3 in $\mathcal{R}^{n_\theta}$ by the following relations

$$\begin{aligned} u_3 &\stackrel{\text{def}}{=} \left([A_1^T\ C^T]\begin{bmatrix} A_1 \\ C \end{bmatrix}\right)^{-1}[A_1^T\ C^T]\begin{bmatrix} u_4 \\ u_5 \end{bmatrix} \\ &= \alpha^{-1}[A_1^T\ C^T]\begin{bmatrix} u_4 \\ u_5 \end{bmatrix} \end{aligned} \tag{5.16}$$

By substituting the above u_3 into Equation 5.13, the dynamic equations become

$$\begin{aligned} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} &= \begin{bmatrix} A_1 \\ C \end{bmatrix}\alpha^{-1}[A_1^T\ C^T]\begin{bmatrix} u_4 \\ u_5 \end{bmatrix} + \gamma_1 \\ &= \begin{bmatrix} B_1 & B_2^T \\ B_2 & B_3 \end{bmatrix}\begin{bmatrix} u_4 \\ u_5 \end{bmatrix} + \gamma_1 \end{aligned} \tag{5.17}$$

with

$$\begin{aligned} B_1 &= A_1\alpha^{-1}A_1^T \\ B_2 &= C\alpha^{-1}A_1^T \\ B_3 &= C\alpha^{-1}C^T \end{aligned} \tag{5.18}$$

As mentioned before, the orthogonal projection matrix

$$\alpha^{-1}[A_1^T\ C^T]$$

converts the control input from an n_θ vector, u_3, to an $(n_\theta + n_\beta)$ vector , $[u_4^T\ u_5^T]^T$. After designing the control input $[u_4^T\ u_5^T]^T$, the actual input u_3 is generated by Equation 5.16. In the above definitions, B_1 is an invertible matrix for A_1 has full rank, and B_3 is a symmetric matrix. Provided that C has full row rank then B_3 is also invertible. In case that B_3^{-1} exists, u_4 and u_5 could be further defined as

$$\begin{aligned} u_4 &= B_1^{-1}(u_1 + u_6) \\ u_5 &= B_3^{-1}(u_2 + u_7) \end{aligned} \tag{5.19}$$

where u_1 and u_2 are the desired accelerations given in Equation 5.5, and u_6 and u_7 will be defined later by the Lyapunov's second method. With these new u_4 and u_5, Equation 5.17 becomes

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} u_6 \\ u_7 \end{bmatrix} + \begin{bmatrix} B_2^T B_3^{-1}(u_2 + u_7) \\ B_2 B_1^{-1}(u_1 + u_6) \end{bmatrix} + \gamma_1 \tag{5.20}$$

where the first term in the right-hand side is the desired acceleration, the second term is the input vector to be assigned, and the rest represent nonlinear disturbance to the above control system. Equation 5.20 could be converted into error-driven system dynamics by the following definitions. Let

$$\begin{aligned} e_1 &= \begin{bmatrix} \theta - \theta_r \\ \beta - \beta_r \end{bmatrix} \in \mathcal{R}^{n_\theta + n_\beta} \quad ; \quad e_2 = \dot{e}_1 \\ \dot{e}_3 &= -e_1 \qquad\qquad\qquad ; \quad e = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \in \mathcal{R}^{3(n_\theta + n_\beta)} \end{aligned} \tag{5.21}$$

be the error states and

$$K_p = \begin{bmatrix} K_{p\theta} & 0 \\ 0 & K_{p\beta} \end{bmatrix} \in \mathcal{R}^{(n_\theta + n_\beta) \times (n_\theta + n_\beta)} \tag{5.22}$$

$$K_v = \begin{bmatrix} K_{v\theta} & 0 \\ 0 & K_{v\beta} \end{bmatrix} \in \mathcal{R}^{(n_\theta + n_\beta) \times (n_\theta + n_\beta)} \tag{5.23}$$

$$K_I = \begin{bmatrix} K_{I\theta} & 0 \\ 0 & K_{I\beta} \end{bmatrix} \in \mathcal{R}^{(n_\theta + n_\beta) \times (n_\theta + n_\beta)} \tag{5.24}$$

be the new grouped gain matrices, then Equation 5.20 could be transformed into error-driven system dynamic equations

$$\dot{e} = Ae + Bw \tag{5.25}$$

with

$$A = \begin{bmatrix} 0 & \mathcal{I} & 0 \\ K_p & K_v & K_I \\ -\mathcal{I} & 0 & 0 \end{bmatrix} \in \mathcal{R}^{3(n_\theta + n_\beta) \times 3(n_\theta + n_\beta)} \tag{5.26}$$

$$B = \begin{bmatrix} 0 \\ \mathcal{I} \\ 0 \end{bmatrix} \tag{5.27}$$

and

$$w = \begin{bmatrix} u_6 \\ u_7 \end{bmatrix} + \begin{bmatrix} B_2^T B_3^{-1}(u_2 + u_7) \\ B_2 B_1^{-1}(u_1 + u_6) \end{bmatrix} + \gamma_1 \tag{5.28}$$

where $\mathcal{I}$ is an $(n_\theta + n_\beta) \times (n_\theta + n_\beta)$ identity matrix. For constant feedback gain matrices K_p, K_v, and K_I, Equation 5.25 is a linear time-invariant system with disturbance w. So, a quadratic form Lyapunov function is selected as

$$V = e^T P e \in \mathcal{R}^1 \tag{5.29}$$

whose derivative could be derived from Equation 5.25 as

$$\begin{aligned} \dot{V} &= \dot{e}^T P e + e^T P \dot{e} \\ &= e^T (A^T P + P A) e + 2 e^T P B w \\ &= -e^T Q e + 2 e^T P B w \end{aligned} \tag{5.30}$$

where P and Q are positive definite matrices with P and $Q \in \mathcal{R}^{3(n_\theta+n_\beta)\times 3(n_\theta+n_\beta)}$. Also by Theorem 5.2, for a stable matrix A, P and Q satisfy the following relation

$$A^T P + P A = -Q \tag{5.31}$$

which is often called *the Lyapunov matrix equation.* Notice that A is a stable matrix whose eigenvalues are decided by the gain matrices in Equation 5.26. In Equation 5.30, $\dot{V}$ is composed of a negative quadratic term and a nonlinear disturbance w. Recalling from the definition of w in Equation 5.28 that the control input u_6 and u_7 in w are left to be decided. Therefore, they could be selected such that

$$e^T P B w \leq 0 \tag{5.32}$$

is satisfied. If Equation 5.32 is accomplished then $V > 0$ and $\dot{V} < 0$, for all $e \neq 0$. Consequently, according to Corollary 5.1, $e \rightarrow 0$ which implies that $\theta \rightarrow \theta_r$ and $\beta \rightarrow \beta_r = 0$. Since one of the sufficient conditions for $\dot{V} < 0$ is $e^T PBw = 0$, u_6 and u_7 will be solved based on this criterion. First by dividing P into nine $(n_\theta + n_\beta) \times (n_\theta + n_\beta)$ submatrices

$$P = \begin{bmatrix} P_1 & P_2 & P_3 \\ P_2^T & P_4 & P_5 \\ P_3^T & P_5^T & P_6 \end{bmatrix} \tag{5.33}$$

and following the previous definitions that $e^T = [e_1^T \; e_2^T \; e_3^T]$ and $B = [0 \; \mathcal{I} \; 0]^T$, then

$$\begin{aligned} e^T PB &= e_1^T P_2 + e_2^T P_4 + e_3^T P_5^T \\ &\stackrel{\text{def}}{=} [\eta_1^T \; \eta_2^T] \end{aligned} \tag{5.34}$$

with $\eta_1 \in \mathcal{R}^{n_\theta}$ and $\eta_2 \in \mathcal{R}^{n_\beta}$. By these new notations, the scalar equation $e^T PBw = 0$ could be expressed by

$$\begin{aligned} e^T PBw &= (\eta_1^T + \eta_2^T B_2 B_1^{-1}) u_6 + (\eta_2^T + \eta_1^T B_2^T B_3^{-1}) u_7 \\ &\quad + (\eta_1^T B_2^T B_3^{-1} u_2 + \eta_2^T B_2 B_1^{-1} u_1 + [\eta_1^T \; \eta_2^T] \gamma_1) \\ &\stackrel{\text{def}}{=} \Psi^T u_8 + \gamma_2 \\ &= 0 \in \mathcal{R}^1 \end{aligned} \tag{5.35}$$

with

$$\Psi^T = [(\eta_1^T + \eta_2^T B_2 B_1^{-1}) \, (\eta_2^T + \eta_1^T B_2^T B_3^{-1})] \tag{5.36}$$

$$u_8 = \begin{bmatrix} u_6 \\ u_7 \end{bmatrix} \in \mathcal{R}^{n_\theta + n_\beta} \tag{5.37}$$

and

$$\gamma_2 = (\eta_1^T B_2^T B_3^{-1} u_2 + \eta_2^T B_2 B_1^{-1} u_1 + [\eta_1^T \; \eta_2^T] \gamma_1) \tag{5.38}$$

Since Equation 5.35 is a scalar equation, and u_8 is an $(n_\theta + n_\beta)$ vector, finding a u_8 satisfying Equation 5.35 is solving a redundant problem. Here a solution with a minimum norm $\|u_8\|$ value is selected, which is

$$u_8 = \frac{-\gamma_2}{\Psi^T \Psi} \Psi \tag{5.39}$$

provided that $\Psi \neq 0$. After solving u_8, the controller design is accomplished. Now, $V > 0$ and $\dot{V} < 0$, for all $e \neq 0$, and according to the Lyapunov theory, the error state converges asymptotically to null state. Consequently, both nominal trajectory tracking and vibration elimination are retained. It should be noticed that by using the Lyapunov stability criteria, compliant manipulator control design is transformed from an overdetermined to a redundant problem. To construct the final command input u, the u_8 defined in Equations 5.37 and 5.39 and the ideal accelerations u_1 and u_2 given in Equation 5.5 are substituted into Equation 5.19 to produce u_4 and u_5 which are then projected back to $\mathcal{R}^{n_\theta}$ space to generate u_3 by Equation 5.16. The final command input u is the sum of u_3 and nonlinear term f_1 as stated in Equation 5.12. Then according to the above analyses, the input u will fulfill the Lyapunov stability criterion and hence stabilize the nominal tracking motion and structural oscillations.

5.4 Restrictions of the Projection Method

In developing the control law by the projection method, two analytic assumptions are proposed. First, it is assumed that

$$B_3 = C\alpha^{-1}C^T$$

is invertible, which means that C must maintain full row rank during the control process. Since C is an $(n_\beta \times n_\theta)$ matrix, to have a full row rank

implies that $n_\beta \leq n_\theta$. So the projection method is restricted to the case that the number of modeled vibratory modes is not greater than that of nominal joints. Also, inaccessible vibratory modes must be avoided during operation, since a row or rows of zero in C will make B_3 not invertible.

Another assumption applied on Equation 5.39 is that $\Psi \neq 0$. According to Equation 5.35, when $\Psi = 0$, the derivative of the Lyapunov function in Equation 5.30 becomes

$$\dot{V} = -e^T Q e + 2\gamma_2 \tag{5.40}$$

where γ_2 is given by Equation 5.38. In that case, $\dot{V}$ is affected by the nature of γ_2. If γ_2 remains negative then $\dot{V} < 0$ and the stability proof remains valid. Otherwise, $\dot{V}$ becomes positive when

$$2\gamma_2 > e^T Q e > 0 \tag{5.41}$$

which means that when e is inside a spherical *ball*, $e^T Qe$, bounded by $2\gamma_2$, the asymptotical convergence of the error state is not ensured. Geometrically, an error state outside the spherical ball will be driven toward the ball continuously by control input, once the error state enters the ball it will be confined inside the ball but the destination is uncertain. Obviously, by reducing the size of the spherical ball, the uncertain error state will set closer to zero. In the following, we will discuss how to reduce the size of uncertain spherical ball. Recalling that the definition of γ_2 is given by

$$\gamma_2 = (\eta_1^T B_2^T B_3^{-1} u_2 + \eta_2^T B_2 B_1^{-1} u_1 + [\eta_1^T \; \eta_2^T]\gamma_1)$$

where B_1, B_2, B_3, and γ_1 are system properties which can not be manipulated directly during a task, and u_1 and u_2 are the ideal accelerations which can

be reduced only by using small gains. Hence, the major reduction must be accomplished by η_1 and η_2. By Equation 5.34

$$[\eta_1^T \; \eta_2^T] = e_1^T P_2 + e_2^T P_4 + e_3^T P_5^T$$

where e_i, $i \in \{1, 2, 3\}$, are the on-line state errors, so η_1 and η_2 can be reduced by choosing small P_2, P_4, and P_5. However, positive definite matrix P is generally solved numerically from the Lyapunov matrix equation

$$A^T P + PA = -Q$$

for a given stable matrix A and a positive definite matrix Q. So the selection of P is not arbitrary because the Lyapunov matrix equation relationship must be maintained. In the next section, an explicit solution of P for a specific set of A and Q will be presented. These results will show how to obtain a desirable P structure from the adjustment of Q values. Hence a small P could be constructed in terms of Q to reduce the uncertain ball in the above stability analyses. Case studies of using the orthogonal projection method to control a six-link manipulator modeled with four joint or four link compliances are reported in [Tosunoglu, Lin, and Tesar, 1990, a].

5.5 Solution of the Lyapunov Matrix Equation

According to Theorem 5.2, for a stable matrix A and a given positive definite matrix Q, there exits a unique positive definite matrix P such that

$$A^T P + PA = -Q$$

Hence in general control design, P is not chosen directly but solved from the Lyapunov matrix equation for a given pair of A and Q. Generally, numerical methods are used to solve the Lyapunov matrix equation. However, Q is

practically chosen to be diagonal for high dimensional systems, and the gain matrices K_p, K_v, and K_I are usually diagonal to decouple the final state equations. For these particular A and Q matrices, P can be solved explicitly in terms of the submatrices of A and Q. In Equation 5.26, A is given as

$$A = \begin{bmatrix} 0 & \mathcal{I} & 0 \\ K_p & K_v & K_I \\ -\mathcal{I} & 0 & 0 \end{bmatrix} \in \mathcal{R}^{3(n_\theta+n_\beta)\times 3(n_\theta+n_\beta)}$$

and in Equation 5.33, P is divided evenly into nine $(n_\theta + n_\beta) \times (n_\theta + n_\beta)$ submatrices

$$P = \begin{bmatrix} P_1 & P_2 & P_3 \\ P_2^T & P_4 & P_5 \\ P_3^T & P_5^T & P_6 \end{bmatrix}$$

For a diagonal Q defined as

$$Q = \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & Q_2 & 0 \\ 0 & 0 & Q_3 \end{bmatrix} \tag{5.42}$$

where Q_i, $i \in \{1,2,3\}$, are $(n_\theta + n_\beta) \times (n_\theta + n_\beta)$ diagonal submatrices, the submatrices of P have the following explicit solutions:

$$\begin{aligned}
2P_1 &= (K_v^2 - K_p)DQ_1 + (K_p^2 - K_vK_I)DQ_2 + (I - K_v^3K_I^{-1})DQ_3 \\
2P_2 &= -K_vDQ_1 + K_IDQ_2 + K_v^2K_I^{-1}DQ_3 \\
2P_3 &= -K_IDQ_1 + K_pK_IDQ_2 + K_pK_v^2K_I^{-1}DQ_3 \\
2P_4 &= DQ_1 - K_pDQ_2 - K_vK_I^{-1}DQ_3 \\
2P_5 &= -K_I^{-1}Q_3 \\
2P_6 &= -K_vK_IDQ_1 + K_I^2DQ_2 + (K_v^2 + K_p - K_p^2K_vK_I^{-1})DQ_3
\end{aligned} \tag{5.43}$$

with

$$D = (K_pK_v - K_I)^{-1} \tag{5.44}$$

This solution is obtained by using the facts that diagonal matrices remain commutative under multiplication, and P is the unique solution of the Lyapunov matrix equation. Notice that P_i, $i \in \{1,2,3,4,5,6\}$, are diagonal

matrices and linear functions of Q_1, Q_2, and Q_3. Since A is a stable matrix, and K_p, K_v, and K_I are diagonal matrices, it could be shown by the Routh-Hurwitz criterion that $(K_pK_v - K_I) > 0$ which automatically guarantees the existence of D in Equation 5.44. In the above explicit solutions, P is a linear function of Q_i for a given A; this relationship allows a direct regulation of Q values in order to affect the structure of P. For example, in the orthogonal projection method, in order to reduce the η_1 and η_2 defined in Equation 5.34, small Q_1, Q_2, and Q_3 can be employed to produce small P_2, P_4, and P_5 according to Equation 5.43 results. Application of the above explicit P will be demonstrated later in example simulations.

5.6 Modified Controller Design

In the orthogonal projection method, the control law requires the computation of B_1, B_2, B_3, B_1^{-1}, and B_3^{-1}, which creates burden on real-time operation. Also the controller is limited to compliant systems with $n_\theta \geq n_\beta$. To remove such restrictions, new control algorithms are proposed in this section. The system dynamics in Equation 5.11 are given here again as

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} A_1 \\ C \end{bmatrix} u - \left(\begin{bmatrix} A_1 \\ C \end{bmatrix} f_1 + \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \right) \tag{5.45}$$

Now, the composite input u is defined as

$$u = f_1 + A_1^{-1} u_1 + C^* u_2 + u_3 \tag{5.46}$$

where f_1 represents the feedforward component, u_1 and u_2 are the ideal accelerations given in Equation 5.5, and u_3 will be defined in stability analysis. Since A_1 has full rank, its inverse exists and is introduced in the above equation. The $(n_\theta \times n_\beta)$-matrix C^* in Equation 5.46 represents a general matrix

whose exact structure is case dependent and will be discussed below. Substituting the composite u into Equation 5.45 results in the familiar form

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} A_1 \\ C \end{bmatrix} u_3 + \gamma \tag{5.47}$$

where

$$\gamma = \begin{bmatrix} 0 \\ (CC^* - \mathcal{I}')u_2 \end{bmatrix} + \begin{bmatrix} A_1 C^* u_2 \\ CA_1^{-1} u_1 \end{bmatrix} - \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \tag{5.48}$$

and $\mathcal{I}' \in \mathcal{R}^{n_\beta \times n_\beta}$ is an identity matrix. Depending on the dimension of C, different C^* could be selected to simplify the structure of γ. Since C is an $(n_\beta \times n_\theta)$ matrix, when $n_\theta \geq n_\beta$, i.e., the number of modeled vibratory modes is not greater than the number of inputs, and C has a row rank of n_β and no inaccessible vibratory mode occurs, then the right-inverse of C, C^+, exists. Hence C^* is selected as $C^* = C^+$, this selection causes the γ expression in Equation 5.48 to become

$$\gamma = \begin{bmatrix} A_1 C^+ u_2 \\ CA_1^{-1} u_1 \end{bmatrix} - \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \tag{5.49}$$

Otherwise, when the right-inverse C^+ does not exit due to inaccessibility problems or $n_\beta > n_\theta$, or the computation of C^+ requires an unacceptable overhead in controller implementation, a simple way to reduce γ is to select a null matrix $C^* = 0$ so that γ becomes

$$\gamma = \begin{bmatrix} 0 \\ CA_1^{-1} u_1 - u_2 \end{bmatrix} - \begin{bmatrix} C^T \\ A_2 \end{bmatrix} (f_2 + K\beta) \tag{5.50}$$

Both simplified γ structures will be used in the later example studies. However, despite various possible γ forms introduced above, the controller will be designed for the general form in Equation 5.47. Following the error-state definitions in Equation 5.21 and the stable matrix formulation in Equation 5.26,

the error-driven system dynamic equations for Equation 5.47 are expressed by

$$\dot{e} = Ae + B\left(\begin{bmatrix} A_1 \\ C \end{bmatrix} u_3 + \gamma\right) \tag{5.51}$$

where the control component u_3 becomes input of the error-driven system. Again, the Lyapunov function candidate V is selected in a quadratic form

$$V = e^T P e \in \mathcal{R}^1 \tag{5.52}$$

whose time derivative together with Equation 5.51 yields

$$\dot{V} = -e^T Q e + 2e^T P B\left(\begin{bmatrix} A_1 \\ C \end{bmatrix} u_3 + \gamma\right) \tag{5.53}$$

where P and Q are $3(n_\theta + n_\beta) \times 3(n_\theta + n_\beta)$ positive definite matrices and

$$A^T P + PA = -Q \tag{5.54}$$

Although there are many possible selections of u_3 to cause $\dot{V} < 0$ for all $e \neq 0$, here u_3 is selected to generate $\dot{V} = -e^T Q e$. That is, letting η and μ denote the following quantities

$$\begin{aligned} \eta &= [A_1^T \; C^T] B^T P e \in \mathcal{R}^{n_\theta} \\ \mu &= e^T P B \gamma \in \mathcal{R}^1 \end{aligned} \tag{5.55}$$

u_3 is solved from the scalar equation

$$\eta^T u_3 + \mu = 0 \tag{5.56}$$

Since u_3 is an n_θ vector, the above scalar equation is a redundant problem hence more than one solution exist. Therefore additional criterion could be introduced to assist the selection of u_3. Two such criteria are presented here. First, Equation 5.47 is restated as follows

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \left(\begin{bmatrix} A_1 \\ C \end{bmatrix} u_3 + \gamma\right)$$

Since the first term in the right-hand side is the desired acceleration, by minimizing the disturbance caused by the last two terms, the acceleration response will have the best approximation to the ideal result. Hence, the problem becomes finding an optimal u_3 that minimizes

$$\left\| \begin{bmatrix} A_1 \\ C \end{bmatrix} u_3 + \gamma \right\| \tag{5.57}$$

while u_3 is subject to the constraint of Equation 5.56. The optimal solution to this problem is given by

$$u_3 = -Z \left\{ [A_1^T \; C^T]\gamma + \frac{\mu - \eta^T Z [A_1^T \; C^T]\gamma}{\eta^T Z \eta} \eta \right\} \tag{5.58}$$

where

$$Z = \left([A_1^T \; C^T] \begin{bmatrix} A_1 \\ C \end{bmatrix} \right)^{-1} \tag{5.59}$$

Due to the full rank of A_1, Z is positive definite and hence nonsingular. Apparently, u_3 in this design demands some computational effort which makes it unattractive for real-time implementation. Therefore, another criterion is proposed, whose result will be used in the following example studies. Now, the object is to minimize $\|u_3\|$ which is subject to the Equation 5.56 constraint, and the solution is simply given by

$$u_3 = -\left(\frac{\mu}{\eta^T \eta} \right) \eta \tag{5.60}$$

where η is assumed to be nonzero. The situations when a null vector η occurs for $e \neq 0$ will be discussed later. For the designed u_3 values, $V > 0$ and $\dot{V} < 0$ hold for every $e \neq 0$, which means that asymptotic stability of error state is ensured and consequently tracking of desired trajectories and elimination of oscillations are obtained since the error states are defined as the difference between plant and reference values. Notice that the modified control algorithms have much simpler structures than that of the orthogonal projection method, therefore, the former control laws demand less computational effort.

5.7 Effect of Matrix P on System Response

In the orthogonal projection method, a P matrix with relatively small magnitude entries is suggested to improve system stability. Here, the effect of P on controller performance will be analyzed. To investigate system response, the u_3 designed in Equation 5.60 is substituted back to Equation 5.47 which results in a compact form

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + (\mathcal{I} - S)\gamma \tag{5.61}$$

with

$$\begin{aligned} S &= \frac{1}{v^T R v}(R v v^T) \\ v &= B^T P e \\ R &= [A_1^T \; C^T]^T [A_1^T \; C^T] \end{aligned} \tag{5.62}$$

where γ is defined in Equation 5.48 and $\mathcal{I}$ is an $(n_\theta + n_\beta) \times (n_\theta + n_\beta)$ identity matrix. Note that S is idempotent; that is, $S^2 = S$, and $(\mathcal{I} - S)$ is also idempotent. However, since S is not a symmetric matrix, $(\mathcal{I} - S)$ is not a projection matrix. Another property is that v^T is a left nullvector of $(\mathcal{I} - S)$, i.e., $v^T(\mathcal{I} - S) = 0$, which is a result of Equation 5.56. In Equation 5.61, when $(\mathcal{I} - S)\gamma$ approaches to zero, the controlled system approaches the ideal acceleration. According to Equation 5.62, P is a constituent of S, so the selection of P can affect the value of $(\mathcal{I} - S)\gamma$; hence, improve system motion response. Furthermore, recalling the definition of B as $B = [0 \; \mathcal{I} \; 0]^T$, v in Equation 5.62 takes on the form $v = P_2 e_1 + P_4 e_2 + P_5 e_3$, which indicates that only P_2, P_4, and P_5 submatrices are involved in the computation of v, therefore, only these submatrices need to be monitored to affect S and hence control response. Once the preferable P_2, P_4, and P_5 structures are decided,

they could be generated by adjusting the Q_1, Q_2, and Q_3 submatrices defined in Equation 5.43. Unfortunately, the nonlinear nature of Equation 5.62 hinders the effort of finding a constant P matrix analytically that minimizes the disturbance $(\mathcal{I} - S)\gamma$. However, according to our case study results, a large difference between the first n_θ and the next n_β diagonal elements in P_2, P_4, and P_5 submatrices will enhance controller performance. This characteristic behavior will be demonstrated later in the case studies, but an explanation is given here. Since the v in Equation 5.62 is defined as $v = P_2 e_1 + P_4 e_2 + P_5 e_3$, the first n_θ diagonal elements of submatrices P_2, P_4, and P_5 are the gains of nominal state errors in e_1, e_2, and e_3, while the next n_β diagonal elements are the gains of vibratory state errors. A large difference between these gain elements will emphasize the errors of nominal states but suppress the effect of vibratory state errors. Since the nominal states are smooth and comparatively slow-moving in contrast to the high frequency oscillations, the large difference arrangement on the P submatrices will reduce the high frequency vibrational disturbance on S and consequently produce a better system response.

5.8 Effect of Matrix P on System Stability

One assumption used in the above derivation of Lyapunov stability is that η is not a null vector for all $e \neq 0$. If $\eta = 0$ for some $e \neq 0$ then by Equations 5.53 and 5.55 the asymptotical stability is uncertain when

$$2\mu > e^T Q e > 0 \tag{5.63}$$

To choose a matrix P to improve the stability region, the structure of η will be analyzed from the inverse identity defined in Equation 4.7 which gives C

as

$$\begin{aligned} C &= -\Lambda_2^{-1}\Sigma\left(\Lambda_1 - \Sigma^T\Lambda_2^{-1}\Sigma\right)^{-1} \\ &= -\Lambda_2^{-1}\Sigma A_1 \end{aligned} \tag{5.64}$$

A substitution of the above C expression into the η defined in Equation 5.55 results in

$$\begin{aligned} \eta &= [A_1^T\, C^T]B^T Pe = [A_1^T\,(-A_1^T\Sigma^T\Lambda_2^{-T})]v \\ &= A_1^T[\mathcal{I}\,(-\Sigma^T\Lambda_2^{-T})]v \end{aligned} \tag{5.65}$$

where $\mathcal{I}$ is an $(n_\theta \times n_\theta)$ identity matrix. In the above equation, A_1 is a non-singular matrix, and $[\mathcal{I}\,(-\Sigma^T\Lambda_2^{-T})] \in \mathcal{R}^{n_\theta \times (n_\theta + n_\beta)}$ which has a nullity of n_β. Therefore, η becomes zero when v is in the null space of $[\mathcal{I}\,(-\Sigma^T\Lambda_2^{-T})]$. However, due to the geometric dependence of Λ_2 and Σ on θ and β, identification of this null space is a demanding task especially for a moving manipulator. Hence, a qualitative analysis is given to this problem. As the submatrices of a generalized inertia matrix, Σ and Λ_2 generally have a similar order of magnitude, therefore, the submatrix $(-\Sigma^T\Lambda_2^{-T})$ has entries with a small order of magnitude perhaps around 1, which means that $[\mathcal{I}\,(-\Sigma^T\Lambda_2^{-T})]$ could be roughly represented by

$$\begin{bmatrix} 1 & \cdots & 0 & \pm 1 & \cdots & \pm 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & \pm 1 & \cdots & \pm 1 \end{bmatrix} \tag{5.66}$$

so when the entries of $(n_\theta + n_\beta)$-vector v has a large difference between the first n_θ and the last n_β elements, v will be away from the null space of $[\mathcal{I}(-\Sigma^T\Lambda_2^{-T})]$ hence $[\mathcal{I}(-\Sigma^T\Lambda_2^{-T})]v \neq 0$, which further implies that $\eta \neq 0$ for all $e \neq 0$. This result supports the assumption used in the stability analysis. Recalling that $v = P_2 e_1 + P_4 e_2 + P_5 e_3$, in order to create a v with a large difference between

the first n_θ and the last n_β elements, P_2, P_4, and P_5 could be constructed with a relatively large difference between the first n_θ and the next n_β diagonal elements. This provides us a guide on the selection of P values. Coincidentally, such P structures are also suggested to enhance the controller performance in the last section. Of course, the above analyses only give a qualitative solution to avoid a null η which is also affected by the on-line error-state values.

5.9 Numerical Simulations on a Six-Link Manipulator

In order to test controller performance, numerical simulations are conducted on three flexible manipulators which have the same kinematic structure but different compliant components. All three cases use the model of a six-degree-of-freedom Cincinnati Milacron T3-776 industrial robot shown in Figure 5.1. The first example considers six joint compliances of the robot, the second and the third models contain three compliant joints and two flexible links. The compliant joints modeled in the second example are the first three joints, also the forearm and upper arm are considered flexible whose lateral deflections are approximated by two orthogonal translational springs located at the end of each link. To test the generality of the proposed controller, a third example is added to the case studies. The third model has the same number of vibratory modes as the second example except that the wrist is compliant instead of the first three joints. Figure 5.2 shows the spring model depicting the joint compliance modeled in the simulations. Figure 5.3 presents the orthogonal linear springs used in approximating the lateral deflections of each flexible link. The spring stiffnesses used in the case studies are arbitrarily chosen for illustration purposes, however, they are approximately 2 to 5 times softer than the actual values measured by [Sklar, 1988]

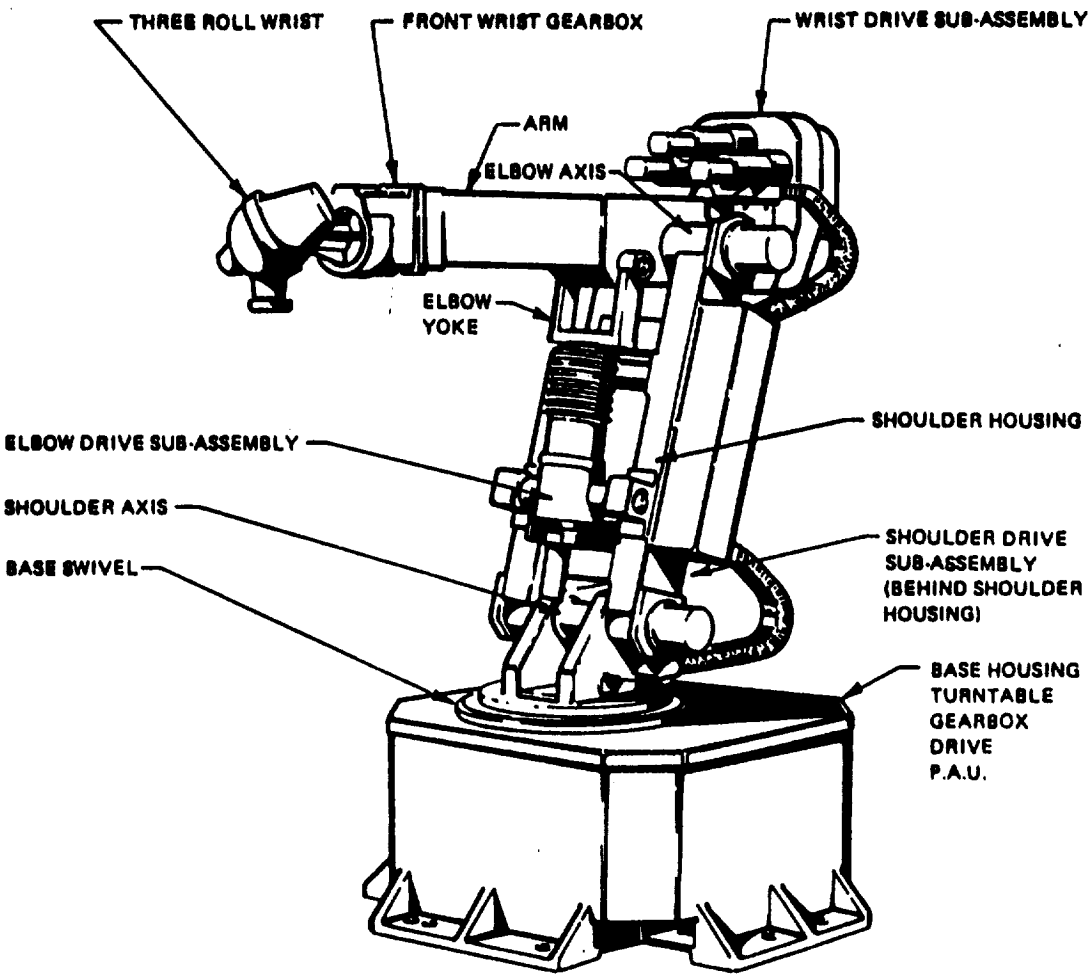


Figure 5.1: Cincinnati Milacron T3-776 Robot

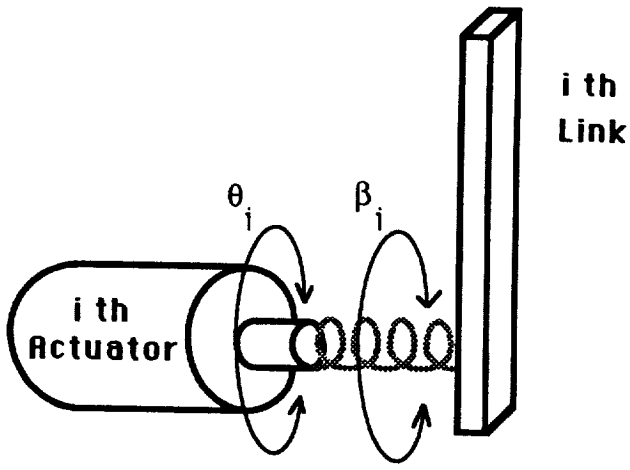


Figure 5.2: Joint Compliance Model

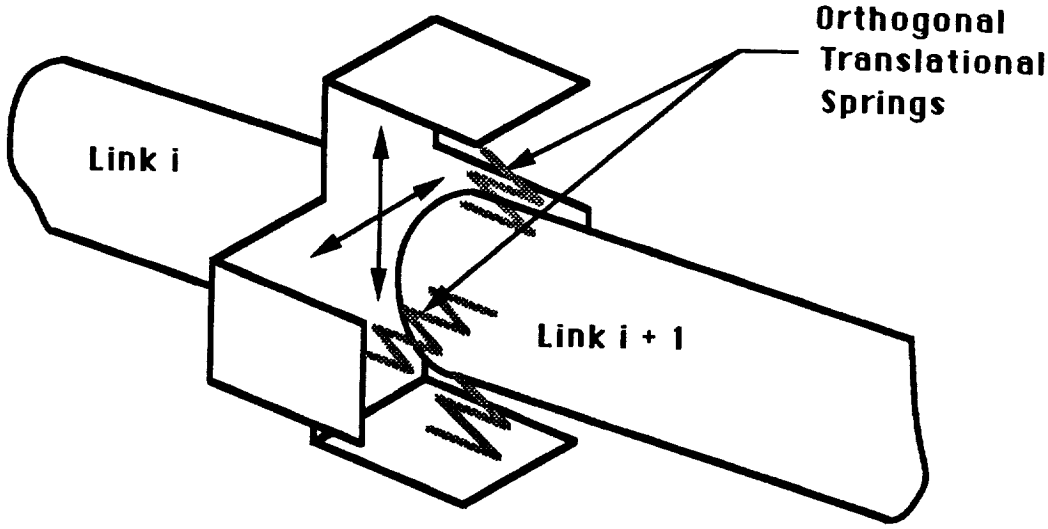


Figure 5.3: Lumped Link Compliance Model
of Upper Arm and Forearm

and [Behi, 1985]. The system parameters of the T3-776 robot are listed in the following tables.

Table 5.1 T3-776 Kinematic Parameters			
	Link Length (m) (x, y, z)	Center of Mass (m) (x, y, z)	Offset Angle (deg)
Link 1	(0, 0, 0.8128)	(0, 0, -0.4318)	0
Link 2	(1.1776, 0, 0)	(0.508, -0.0254, 0)	90
Link 3	(0, 0, 0)	(0.1016, -0.1778, 0)	0
Link 4	(0, 0, 1.397)	(0, 0, -0.508)	90
Link 5	(0, 0, 0)	(0, 0, 0)	-60
Link 6	(0, 0, 0.1524)	(0, 0, -0.1016)	60
Payload		(0, 0, 0.0254)	

Table 5.2 T3-776 Inertial Parameters		
	Mass (kg)	Moment of Inertia ($kg.m^2$) (I_{xx}, I_{yy}, I_{zz})
Link 1	317.5	(0, 0, 29.3)
Link 2	680.4	(5.9, 52.7, 43.9)
Link 3	453.6	(49.7, 7.61, 49.7)
Link 4	68	(0.59, 0.59, 0.35)
Link 5	36.3	(0.23, 0.23, 0.06)
Link 6	27.2	(0.12, 0.12, 0.06)
Payload	91.	(0.06, 0.06, 0.06)

Table 5.3 T3-776 Actuator Parameters						
Joint	1	2	3	4	5	6
Inertia ($10^{-3} \cdot kg \cdot m^2$)	4.2	2.1	2.1	1.3	1.3	0.8
Damping ($N \cdot m/(rad/s)$)	0.4	0.4	0.3	0.4	0.3	0.3
Resistance (ohm)	0.8	0.8	0.8	0.8	0.8	0.8
Torque Constant ($volt/(rad/s)$)	20	20	14	11	8	8
Gear ratio	100	100	100	80	30	10
Back emf Const. ($N \cdot m/amp$)	0.5	0.5	0.4	0.3	0.3	0.2

In these case studies, the numerical integration step is selected as 0.1 msec, a larger step has been attempted but was abandoned due to numerical instability.

5.9.1 Case 1: Six Compliant Joints

Figure 5.4 illustrates the six-link robot used in the first example where $\beta_1, \cdots, \beta_6$ denote local joint deflections. In this example, each joint compliance is represented by a constant torsional spring acting across the joint as sketched in Figure 5.2. Torsional springs used in this example are assumed massless, therefore, each nominal joint θ_i and the accompanying vibratory mode β_i, $i \in \{1, 2, \cdots, 6\}$, is subjected to the same structural inertial effect. This means that the submatrices of the generalized inertia in Equations 3.48 have the same values, that is,

$$\Lambda_2 = \Sigma = \Sigma^T = \Lambda_1' \tag{5.67}$$

Substitution of this particular property and $\Lambda_1 = \Lambda_1' + J$ into the inverse identity cited below

$$\begin{bmatrix} A_1 & C^T \\ C & A_2 \end{bmatrix}$$

$$= \begin{bmatrix} \left(\Lambda_1 - \Sigma^T \Lambda_2^{-1} \Sigma\right)^{-1} & -\left(\Lambda_1 - \Sigma^T \Lambda_2^{-1} \Sigma\right)^{-1} \Sigma^T \Lambda_2^{-1} \\ -\Lambda_2^{-1} \Sigma \left(\Lambda_1 - \Sigma^T \Lambda_2^{-1} \Sigma\right)^{-1} & \Lambda_2^{-1} + \Lambda_2^{-1} \Sigma \left(\Lambda_1 - \Sigma^T \Lambda_2^{-1} \Sigma\right)^{-1} \Sigma^T \Lambda_2^{-1} \end{bmatrix}$$

produces

$$\begin{aligned} A_1 &= -C^T = -C = J^{-1} \\ A_2 &= (\Lambda_1')^{-1} + J^{-1} \end{aligned}$$

where J is the diagonal actuator inertia matrix defined in Equation 3.56 and whose values are given in Table 5.3. Since in this case $n_\theta = n_\beta = 6$ and joint oscillations always remain accessible as discussed in the last chapter, the right-inverse of C exists and is simply given by $C^+ = -J$. The structure of γ in Equation 5.49 here becomes

$$\gamma = \begin{bmatrix} -u_2 \\ -u_1 \end{bmatrix} - \begin{bmatrix} -J^{-1} \\ (\Lambda_1')^{-1} + J^{-1} \end{bmatrix} (f_2 + K\beta) \tag{5.68}$$

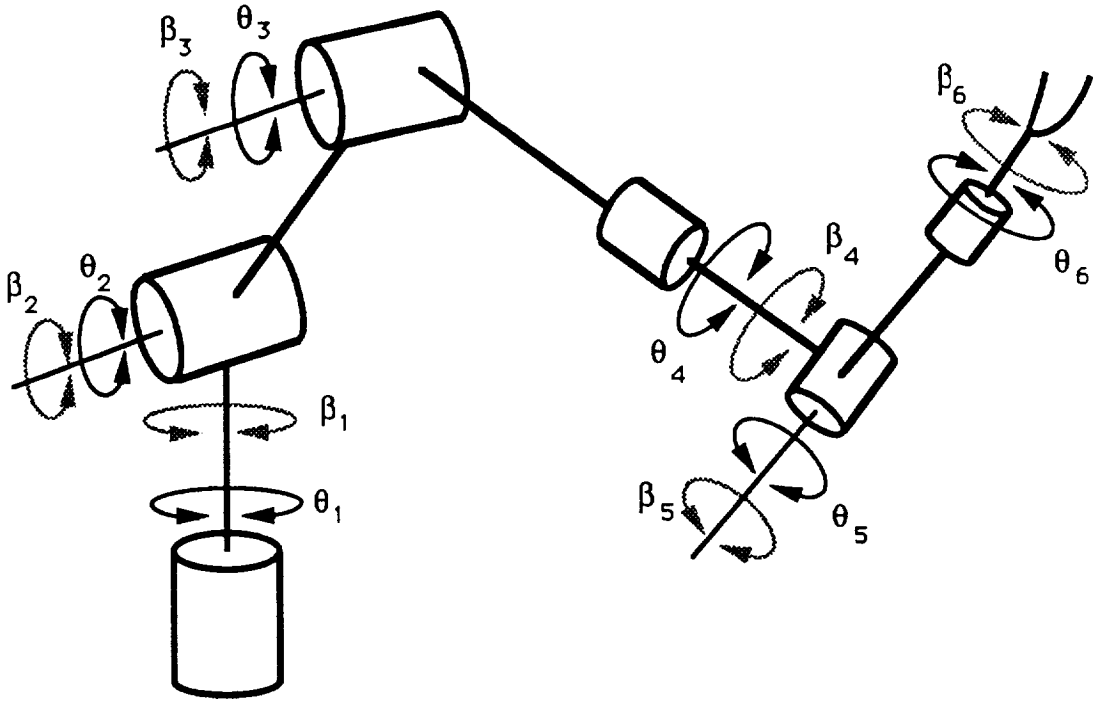


Figure 5.4: A Six-Link Robot Modeled with Six Joint Compliances

The resultant control input in Equation 5.46 takes on the following expression

$$u = f_1 + Ju_1 - Ju_2 + u_3 \tag{5.69}$$

where u_1 and u_2 are the ideal accelerations given in Equation 5.5, and the u_3 defined in Equation 5.60 is used in the above controller. The above results are valid for any n-link robot modeled with n compliant joints. Notice form Equation 5.67 that without adding the actuator inertia J the generalized inertial matrix defined in Equation 3.48 is positive semidefinite in this model.

For illustration purposes, the stiffness matrix for the six compliant joints are chosen as

$$K = \text{diag}[452000\ 339000\ 226000\ (8500)_3] \quad \text{Nm/rad}$$

where $(\cdot)_j$ represents a diagonal entry repeating consecutively j times, and the six diagonal elements correspond to the stiffness values of joints 1 through 6. In this example, eigenvalues $(-3-6-7)$ are assigned to the ideal nominal and vibratory mode accelerations, which produce diagonal feedback gain matrices

$$K_p = [(-81)_{12}]\ K_v = [(-16)_{12}]\ K_I = [(126)_{12}]$$

where the first six diagonal elements are associated with nominal motion and the last six diagonal elements are with vibrational modes. The robot is controlled to follow nominal trajectories defined by

$$\begin{aligned} \theta_r(t, t_f) &= \Delta\theta f(t, t_f) + \theta_{r0} \\ f(t, t_f) &= 6\frac{t^5}{t_f^5} - 15\frac{t^4}{t_f^4} + 10\frac{t^3}{t_f^3} \end{aligned} \tag{5.70}$$

where $f(t, t_f)$ is a normalized fifth-order polynomial with $f(0, t_f) = \dot{f}(0, t_f) = \ddot{f}(0, t_f) = \dot{f}(t_f, t_f) = \ddot{f}(t_f, t_f) = 0$ and $f(t_f, t_f) = 1$, in which t_f is the

termination time for reference motion. In the above equation, $\Delta\theta$ is the amount of joint movement, and θ_{r0} is the starting position of reference trace. The reference trajectory is chosen to have smooth start and stops in order to avoid shocks that might trigger structural resonance. In the simulation, actual joints will start from a position θ_0 distinct from the desired initial θ_{r0}. This difference is deliberately created to test the tracking capability of the designed controller. The trajectory parameters used in this example are tabulated in the next table.

Table 5.4 Case 1 Nominal Trajectory Parameters

	$t_f(sec)$	$\Delta\theta(deg)$	θ_{r0}	θ_0	$\theta_0 - \theta_{r0}$
Joint 1	8	50	0	-10	-10
joint 2	8	40	35	30	-5
Joint 3	8	-40	-55	-45	10
Joint 4	8	40	0	-5	-5
Joint 5	8	50	0	-5	-5
Joint 6	8	20	0	-5	-5

Besides the initial position errors, a discrepancy in payload description is deliberately introduced to test robustness of the controller to parameter variations. The controller presumes a payload of 68 kg while the system actually carries a 91 kg payload which represents a 30 % error.

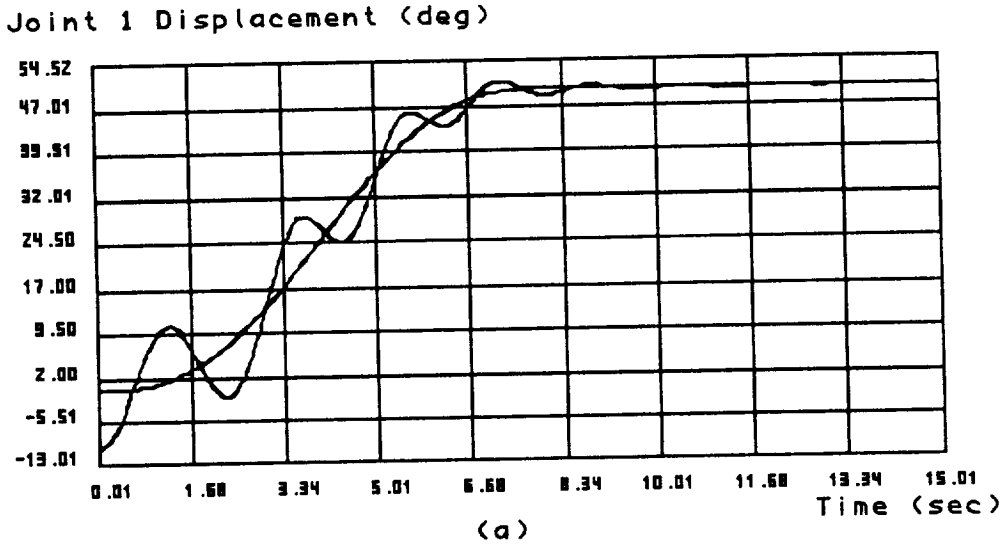
In order to avoid a null η vector for $e \neq 0$ as discussed in the previous section, P needs to have a relatively high difference between the part associated with nominal joints and vibrational modes. Accordingly, the Q values selected for this case are

$$Q_1 = [(100000)_6\,(200)_6]\; Q_2 = [(1000)_6\,(2)_6]\; Q_3 = [(100000)_6\,(100)_6]$$

and for the above given feedback gain matrices this selection generates significantly different P values for nominal and vibratory modes, which are

$$P_1 = [(19499)_6\,(37.57)_6] \quad P_2 = [(824.4)_6\,(1.56)_6] \quad P_3 = [(-16779)_6\,(-26.52)_6]$$
$$P_4 = [(82.78)_6\,(.16)_6] \quad P_5 = [(-396.8)_6\,(-.40)_6] \quad P_6 = [(136022)_6\,(299)_6]$$

For these selected parameters, a simulation is conducted and the results are displayed in the following figures. Figures 5.5(a) to 5.5(f) give the actual and reference joint displacements. Note that an initial error separates these two traces in each figure. Figures 5.5(g) to 5.5(l) display the joint deflections of the controlled system, while Figures 5.5(m) to 5.5(r) show input voltages for each actuator. These results indicate that nominal displacements show asymptotically stable path tracking capability while residual vibrations settle to static deflection values. In Figure 5.5(n), the high voltage surge at the starting point is due to the sudden release of the system, which might physically exceed actuator saturation voltage values. In order to study this problem, the same simulation is repeated with a ± 50 volt bound on each actuator. The results are presented in Figures 5.6(a) to 5.6(r). A comparison of both simulation results suggests that the system response remains almost identical, which implies that for this simulation, controller performance is not sensitive to voltage saturation. Note that in Figures 5.5(m) to 5.5(r) and Figures 5.6(m) to 5.6(r) the high frequency, small magnitude vibrations in control voltages at steady state are to counteract spring torques due to residual oscillations which are considerably small as shown in Figures 5.5(g) to 5.5(l) and Figures 5.6(g) to 5.6(l). Actually, such residual oscillations will quite possibly be dissipated by structural damping not included in the dynamic model.



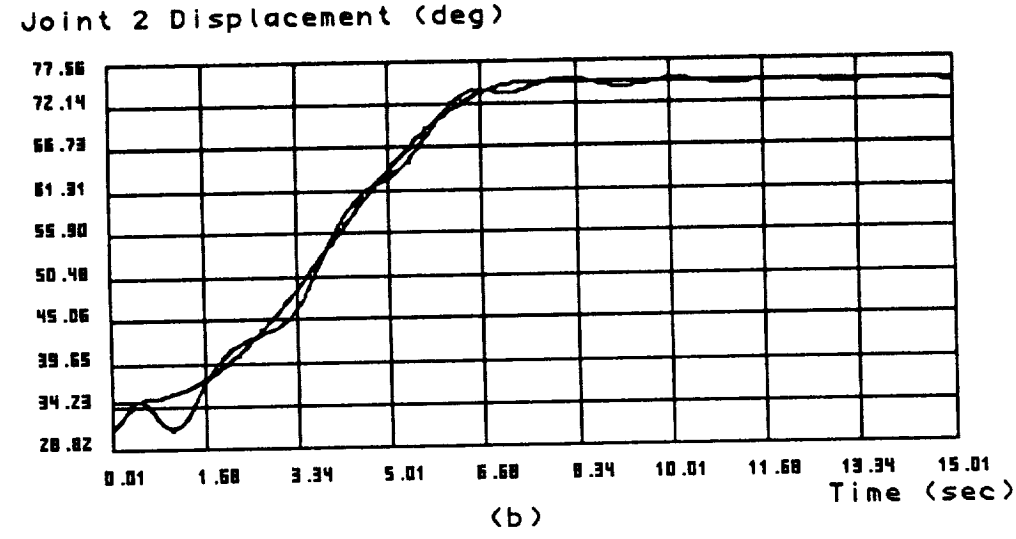
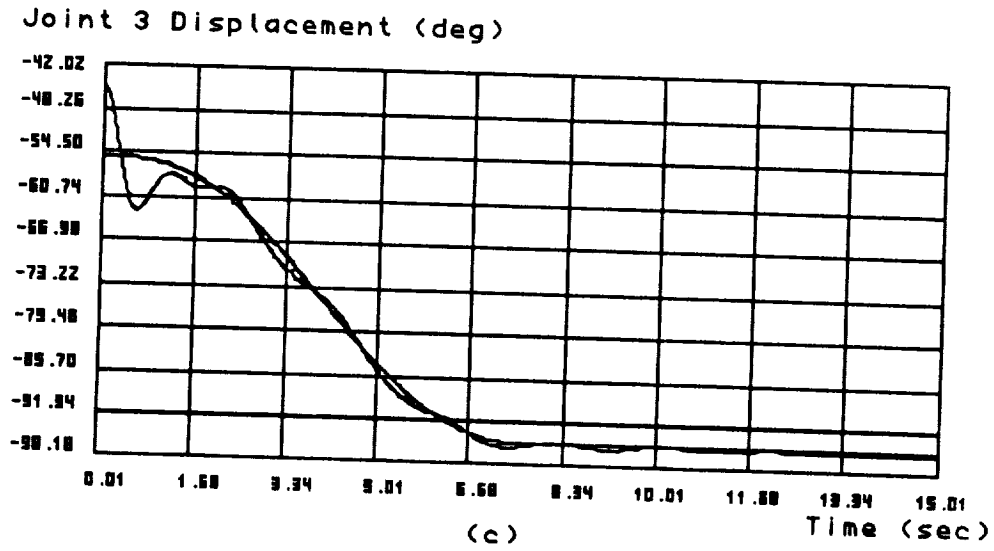


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



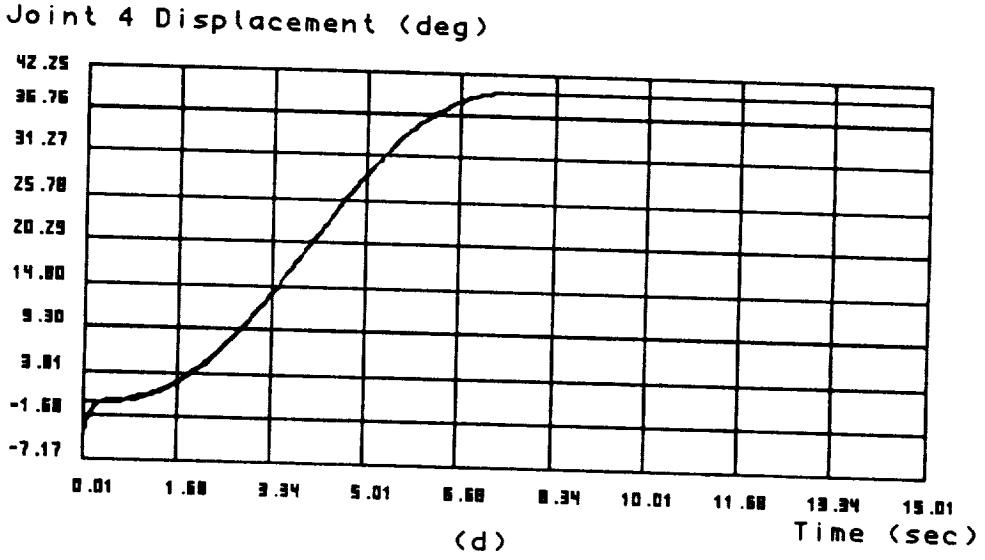
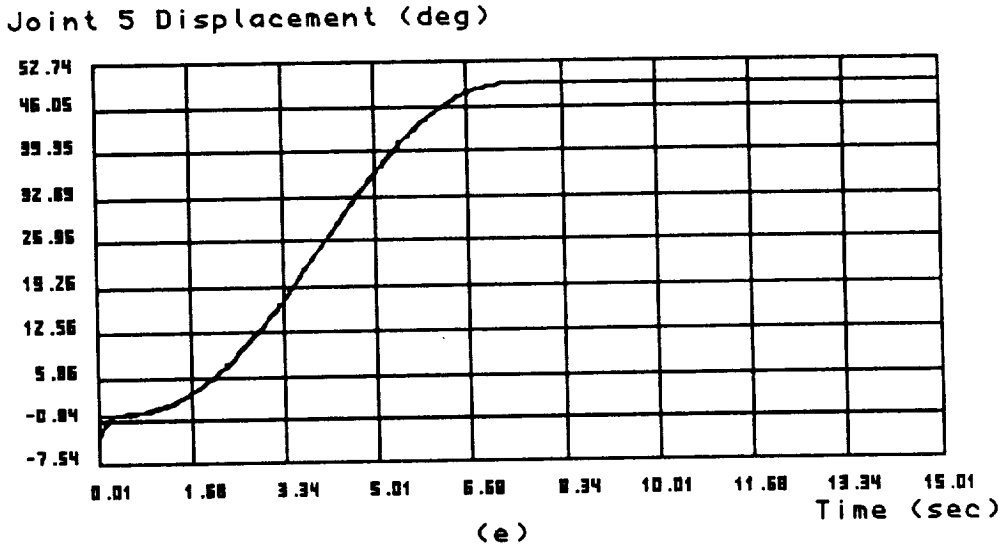


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



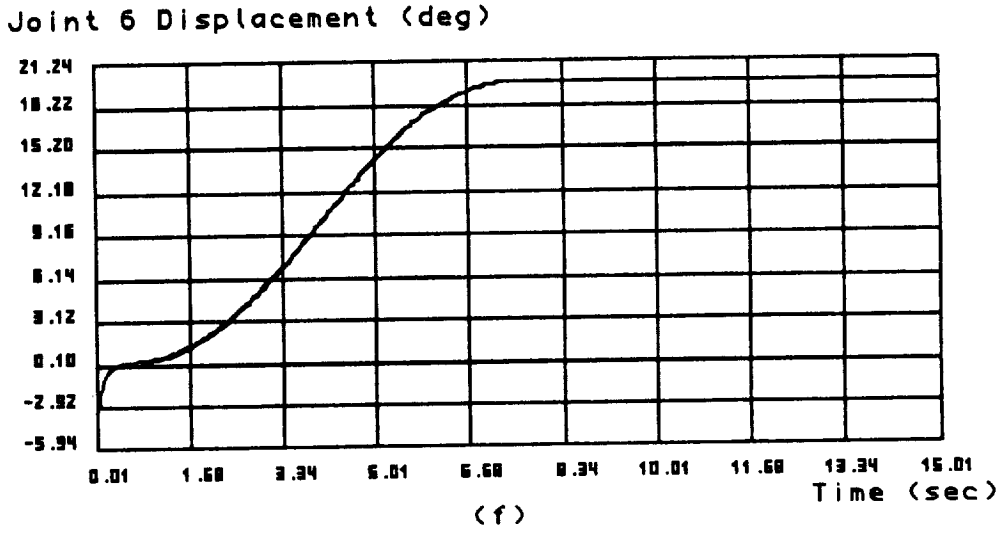
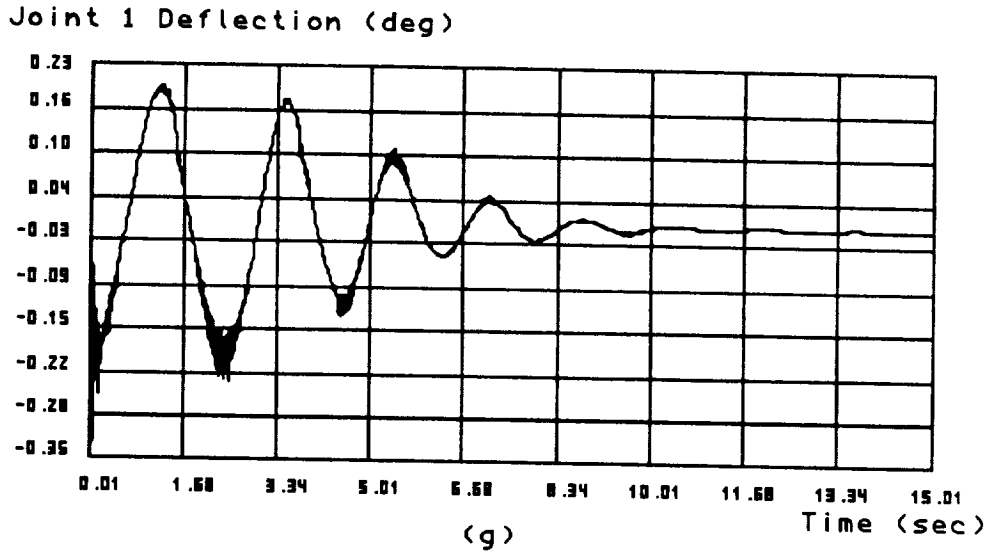


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



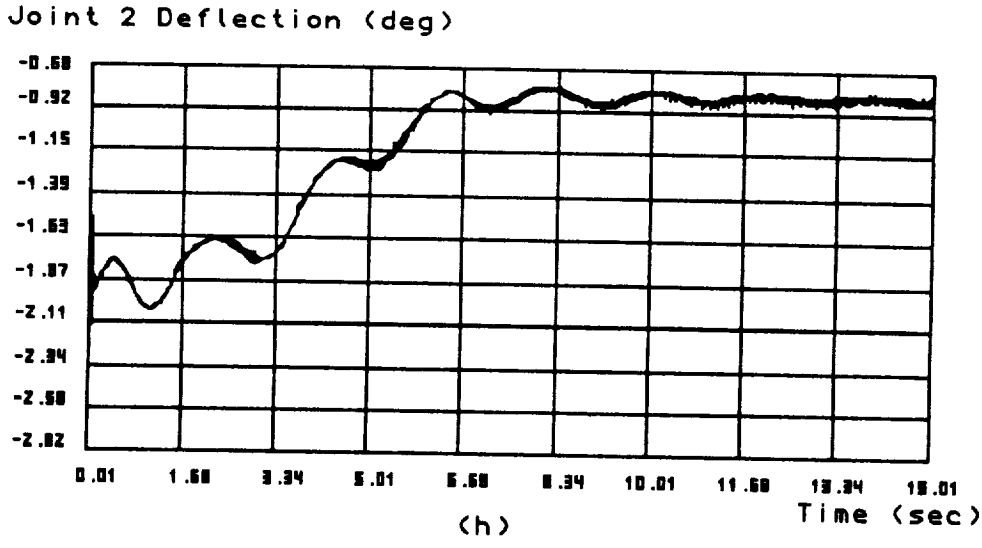
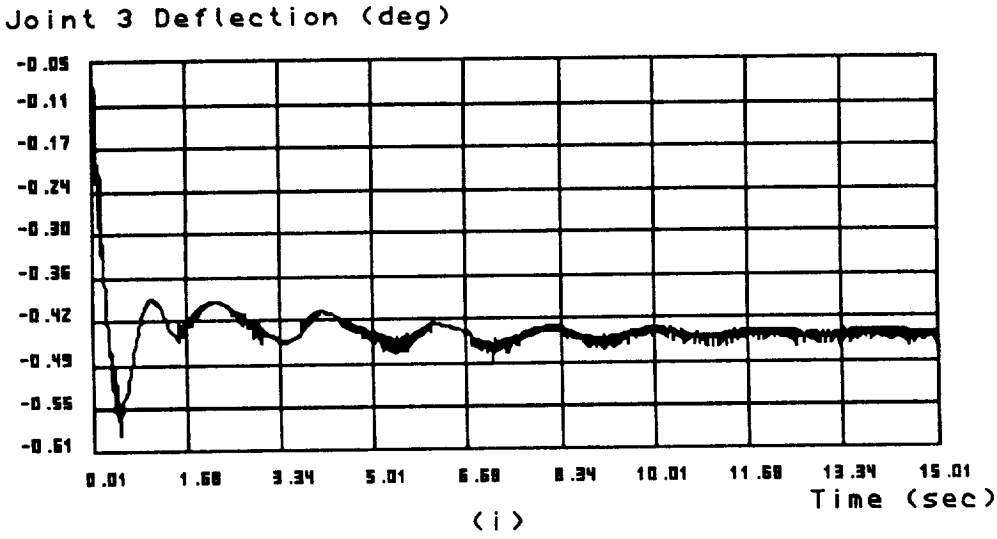


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



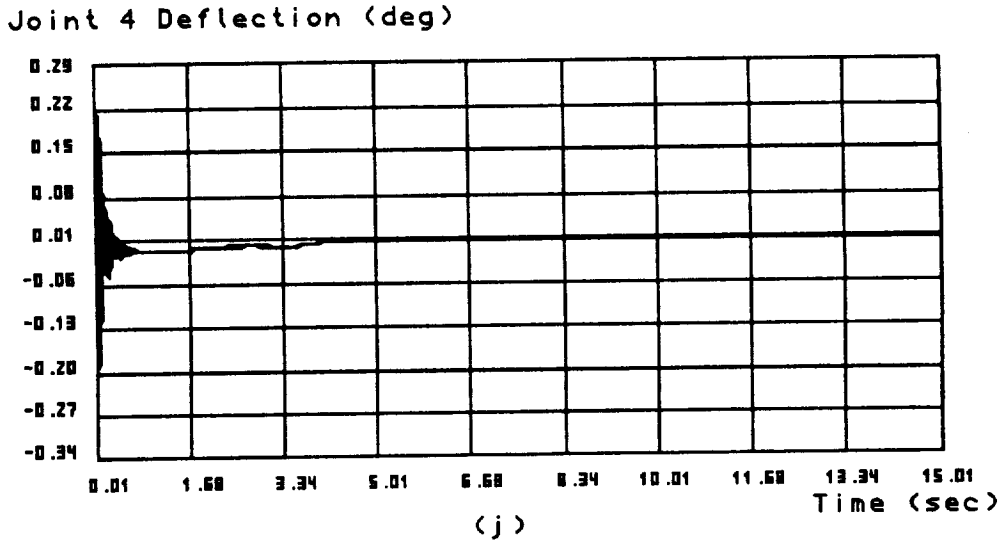
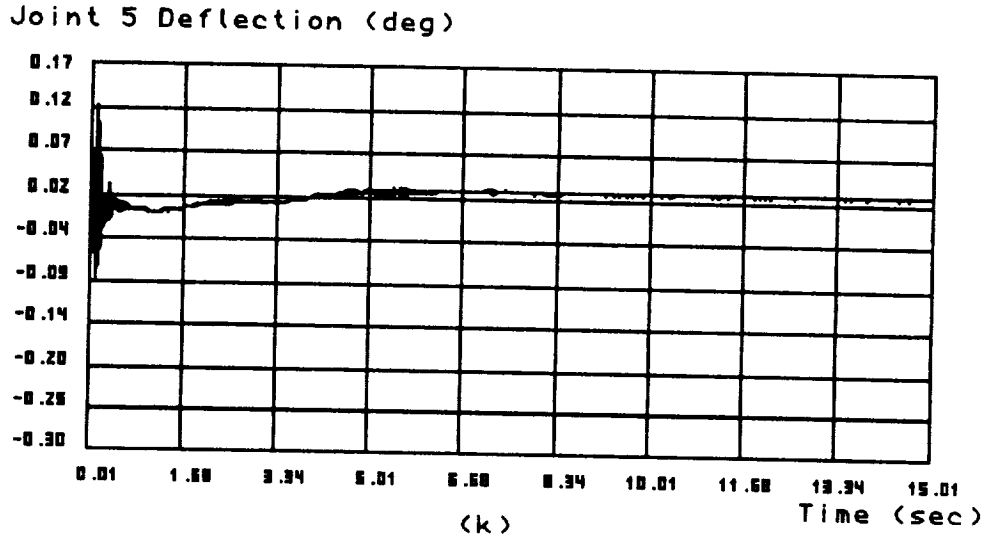


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



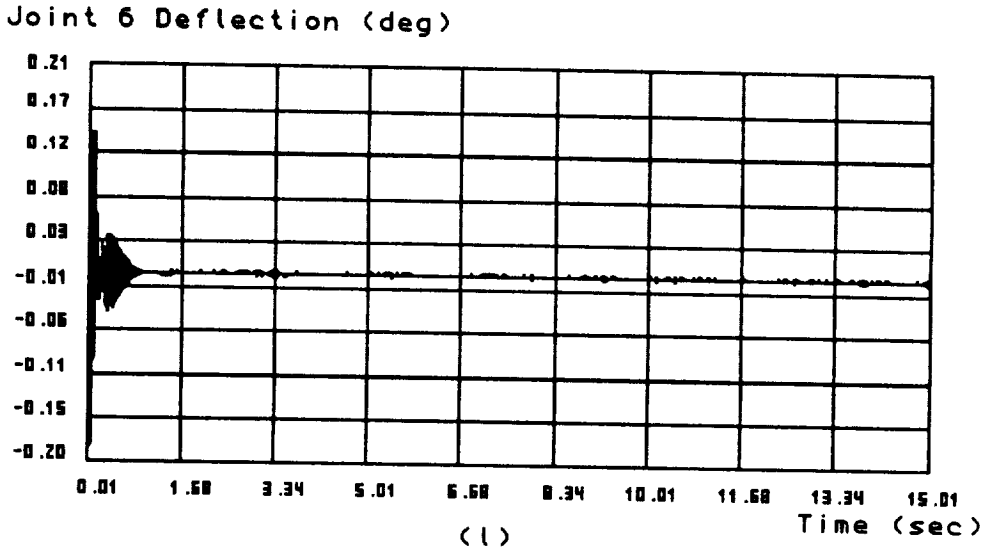
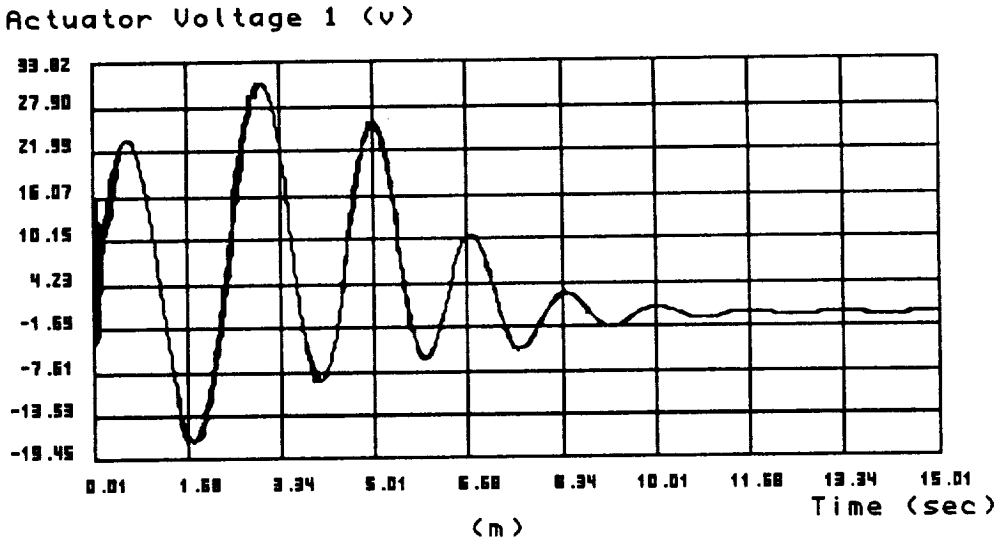


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



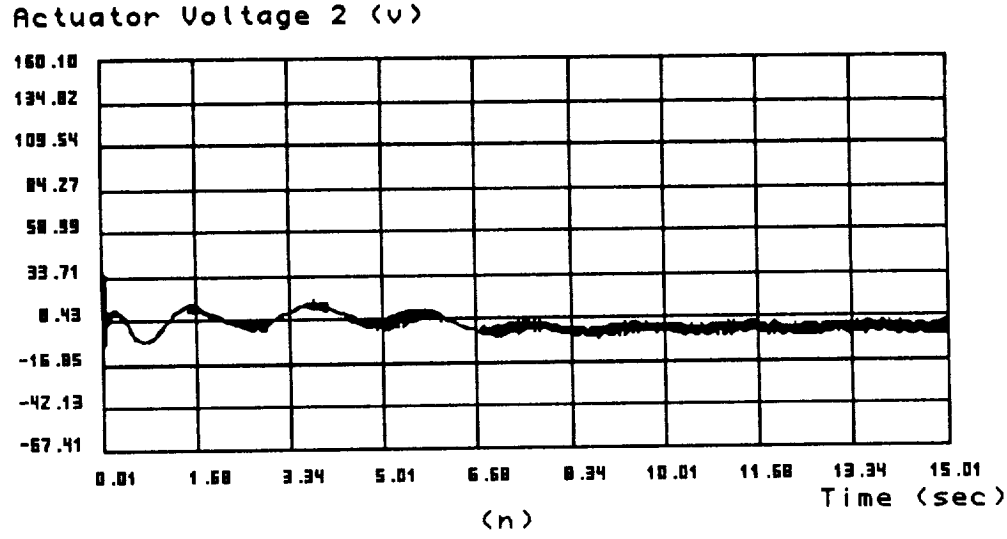
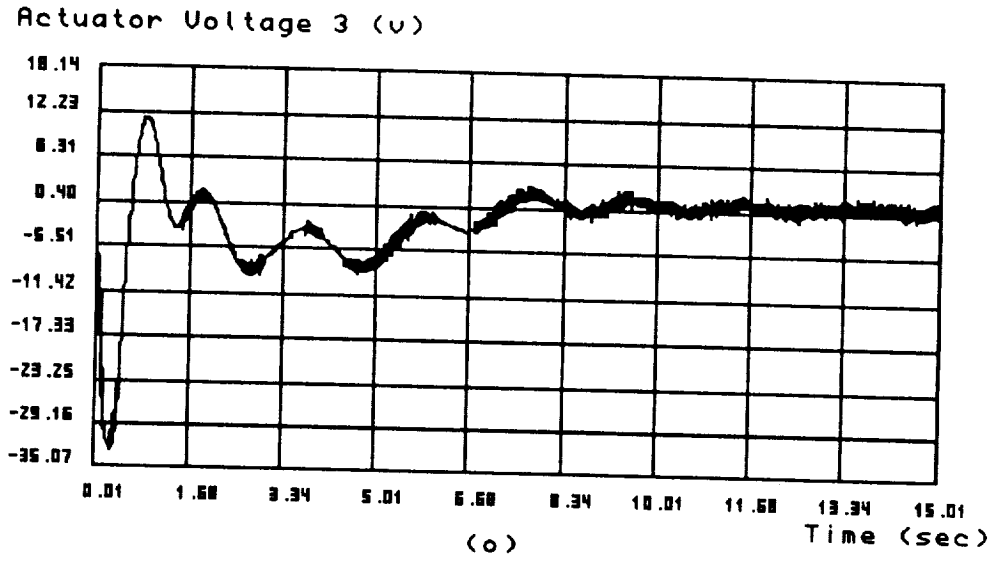


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



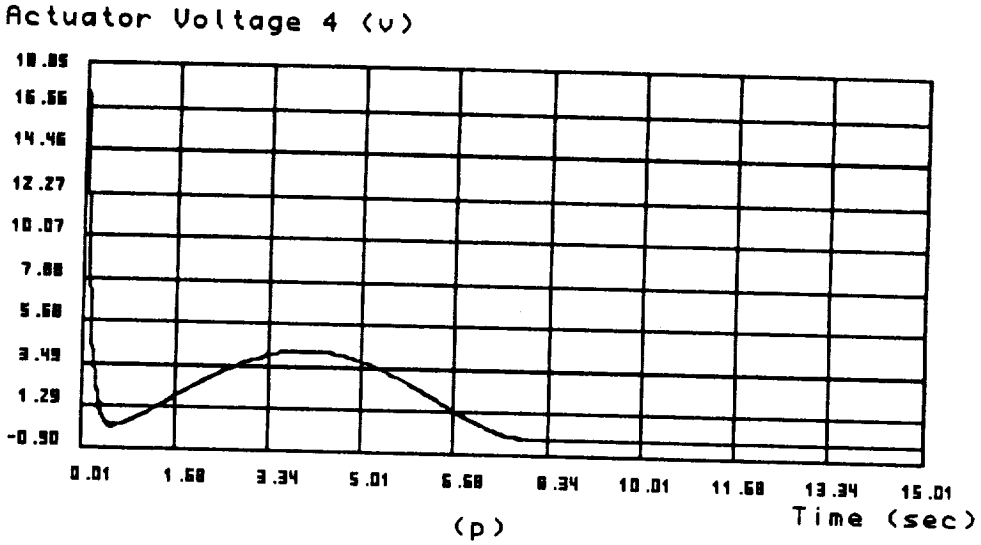
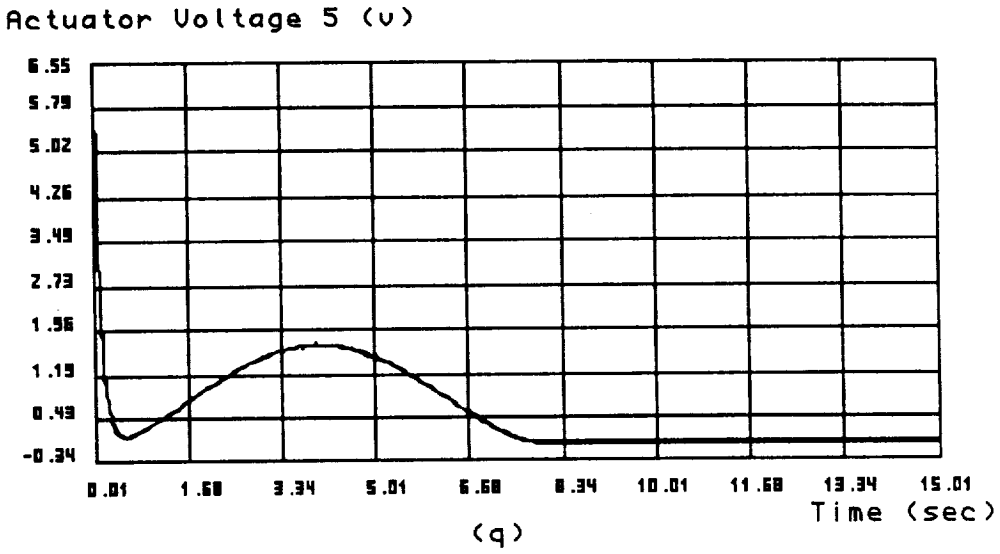


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



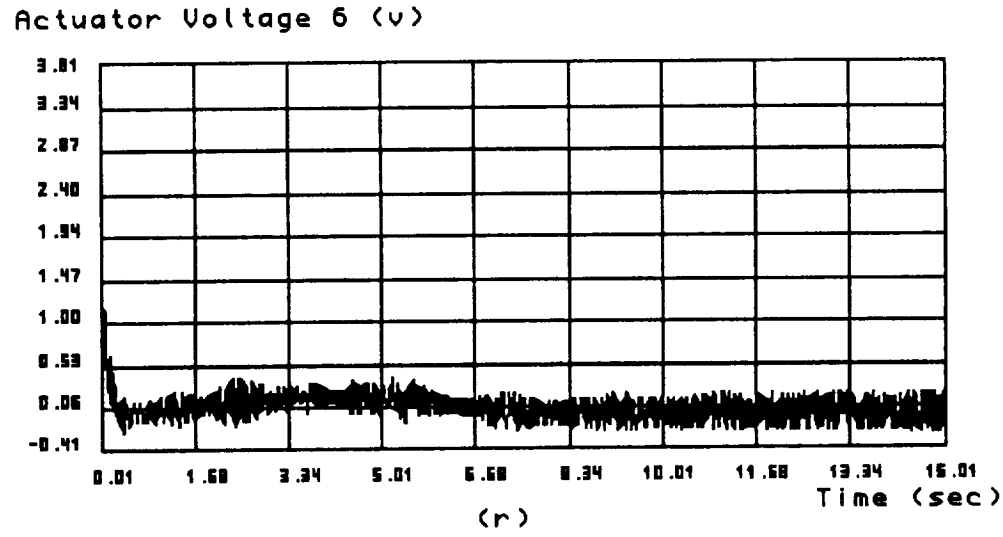
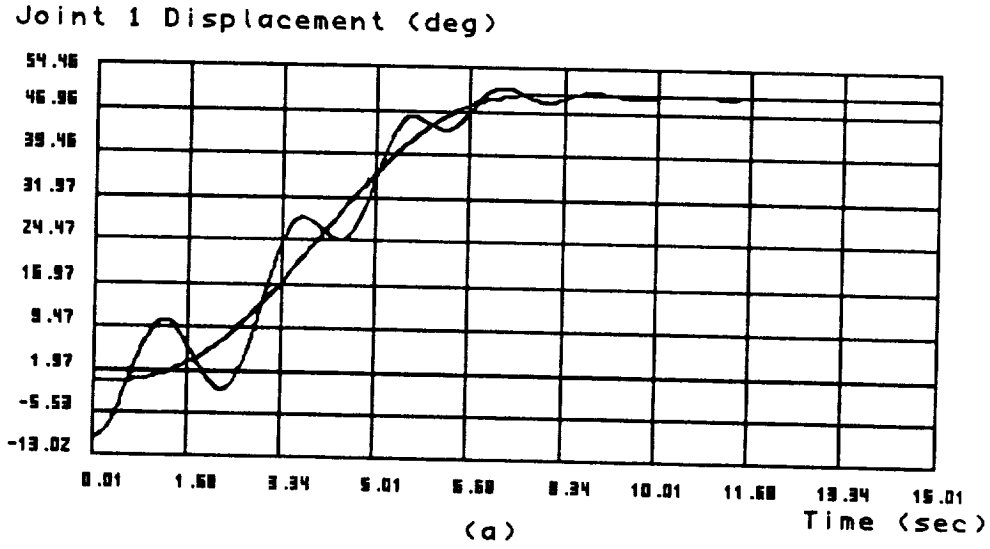


Figure 5.5: Six-Compliant-Joint Model Simulation Results (Without Voltage Bound)



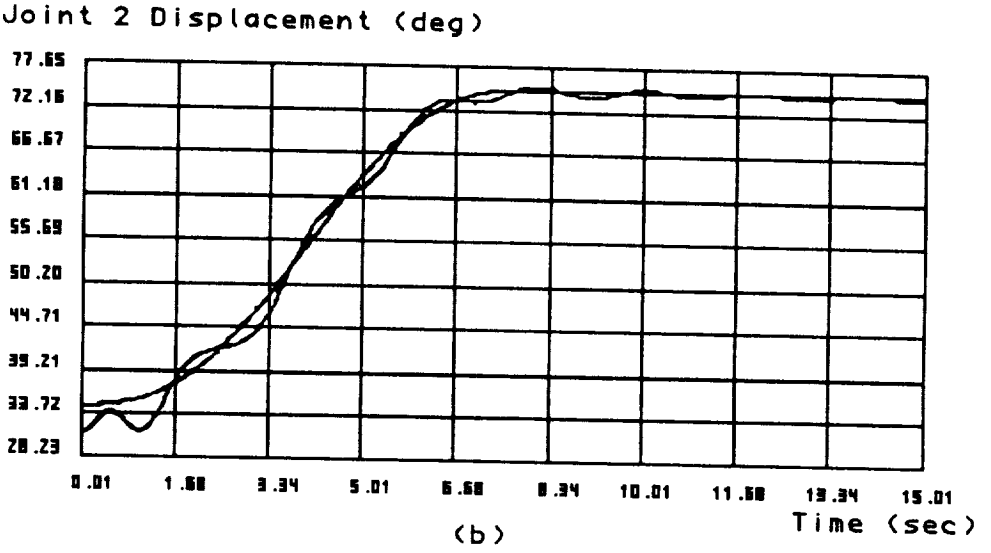
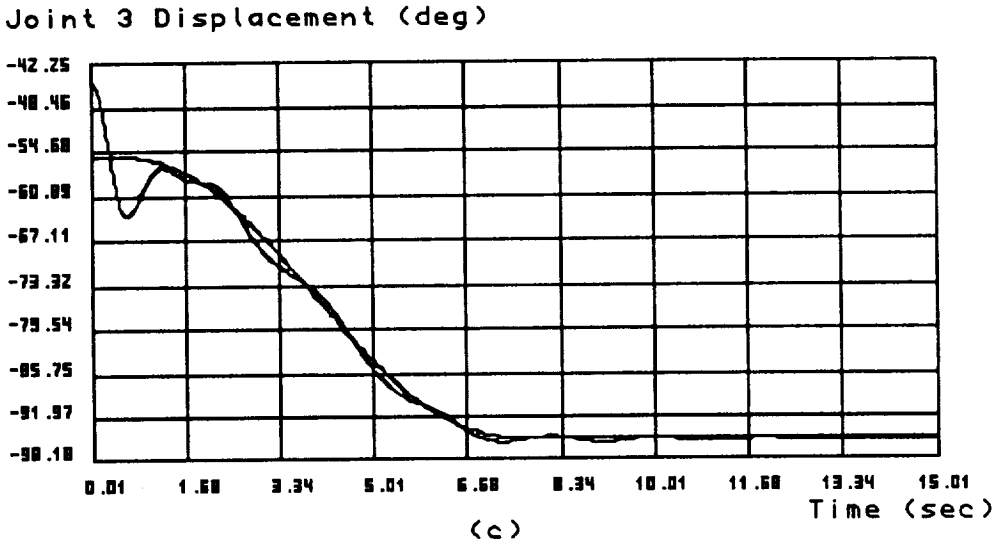


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



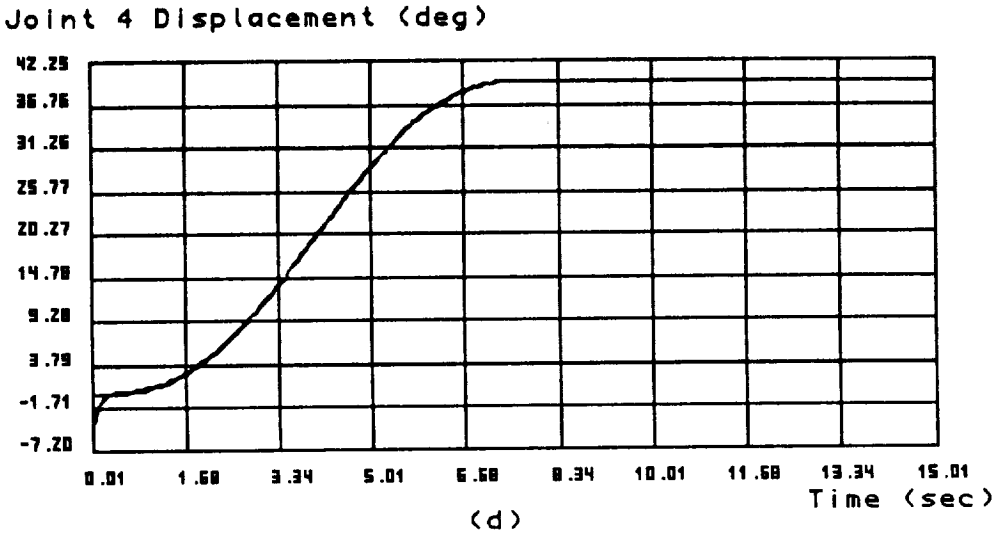
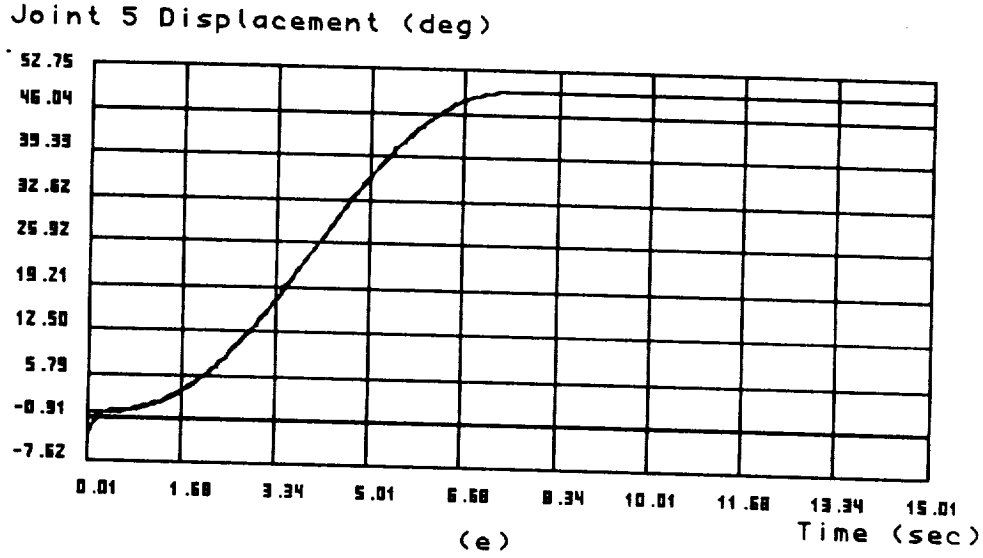


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



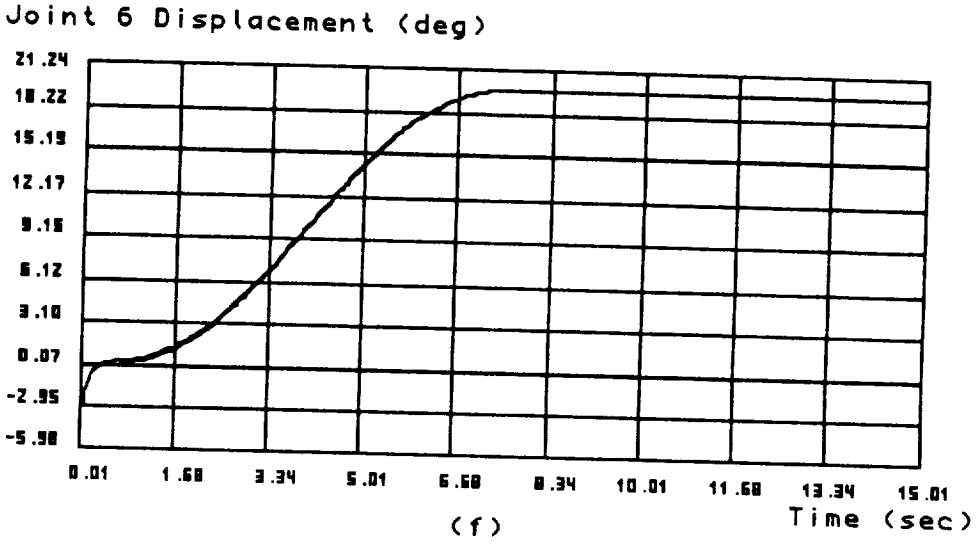
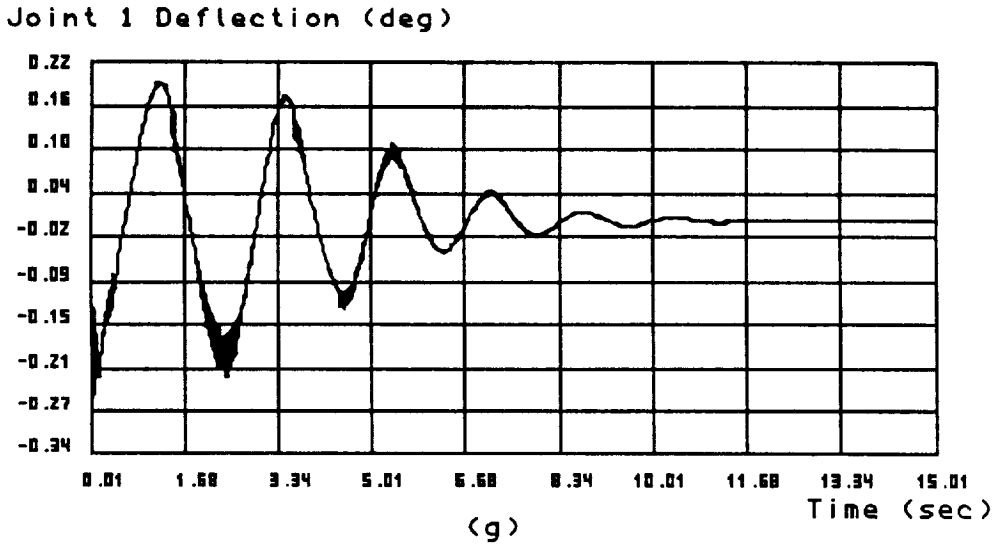


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



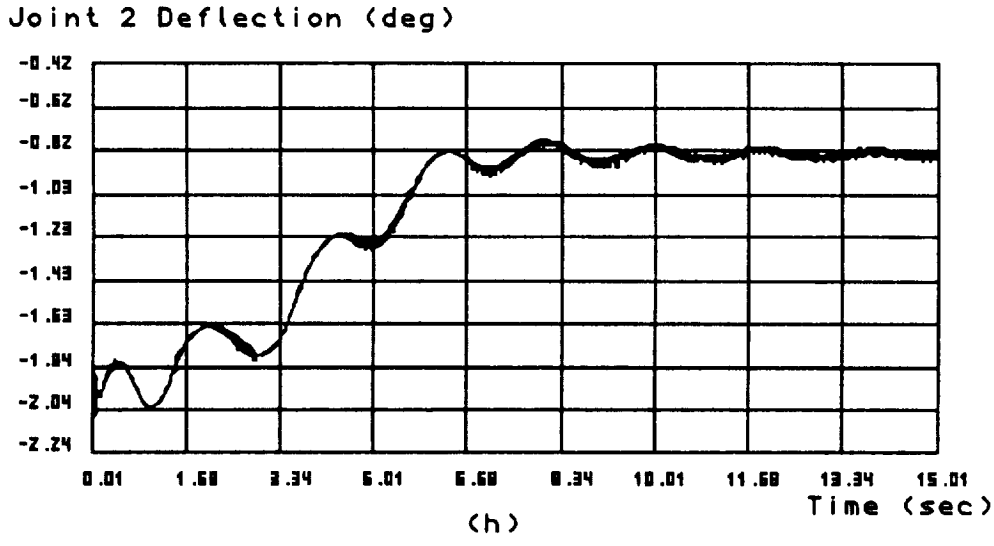
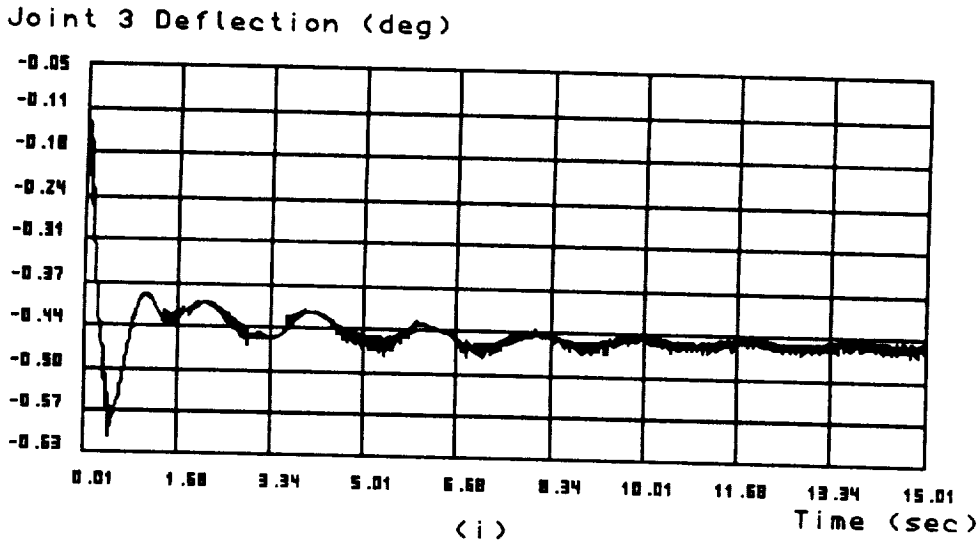


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



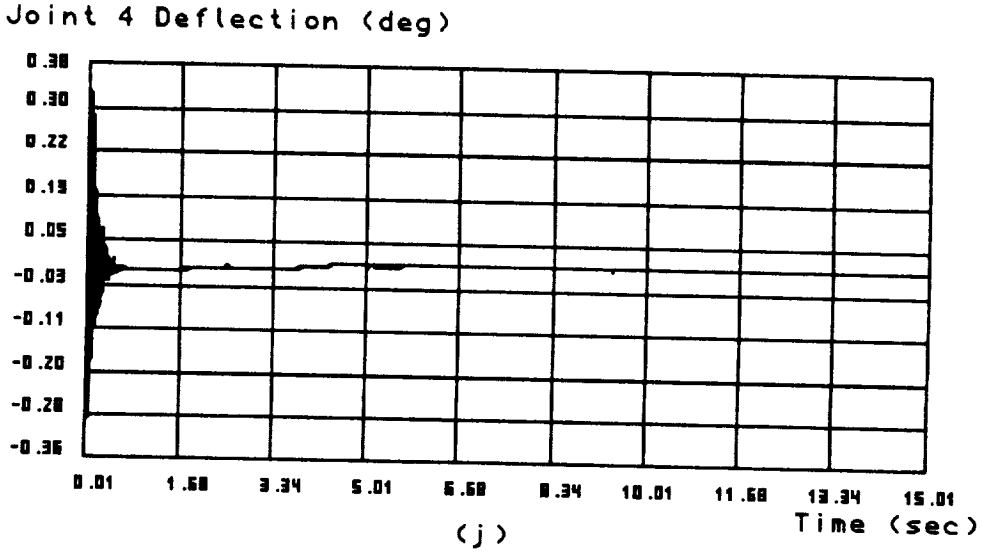
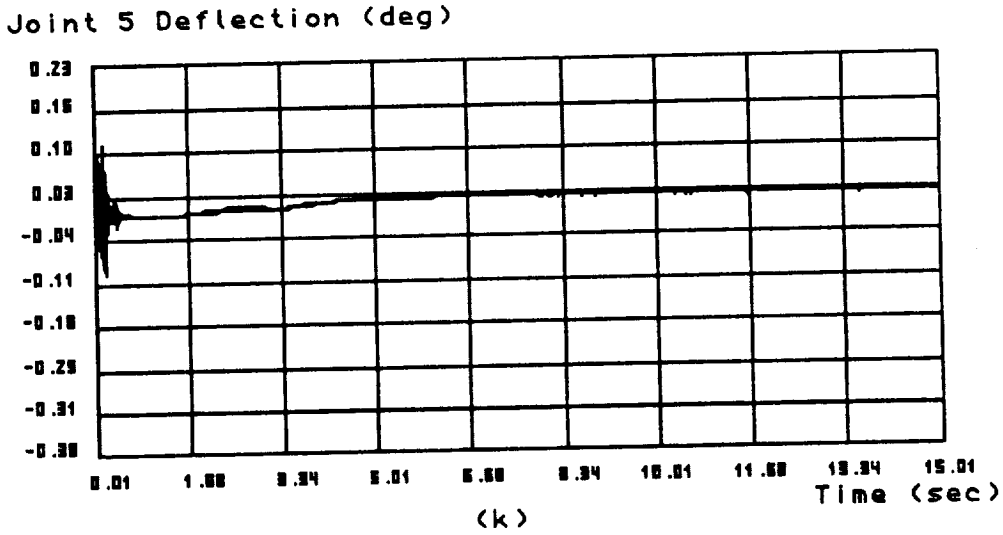


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



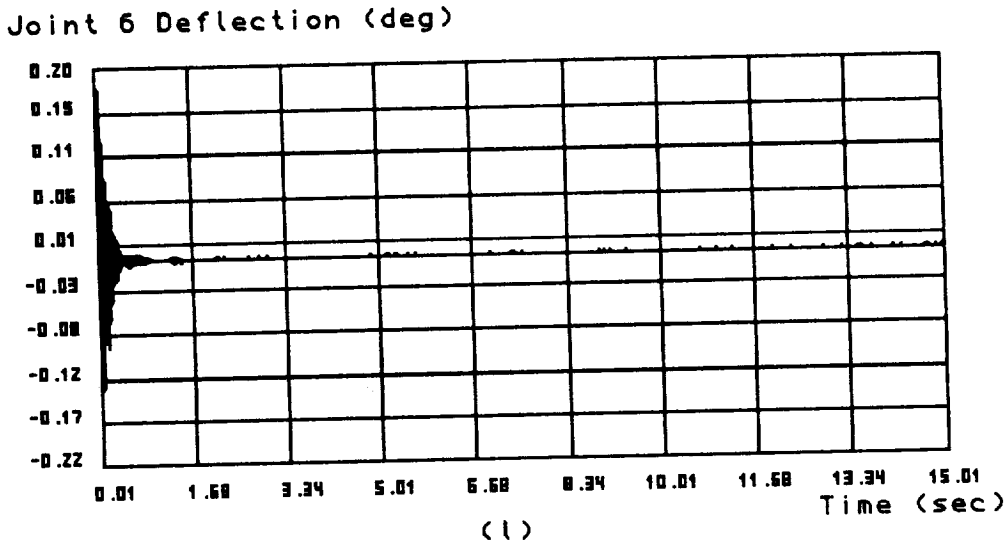
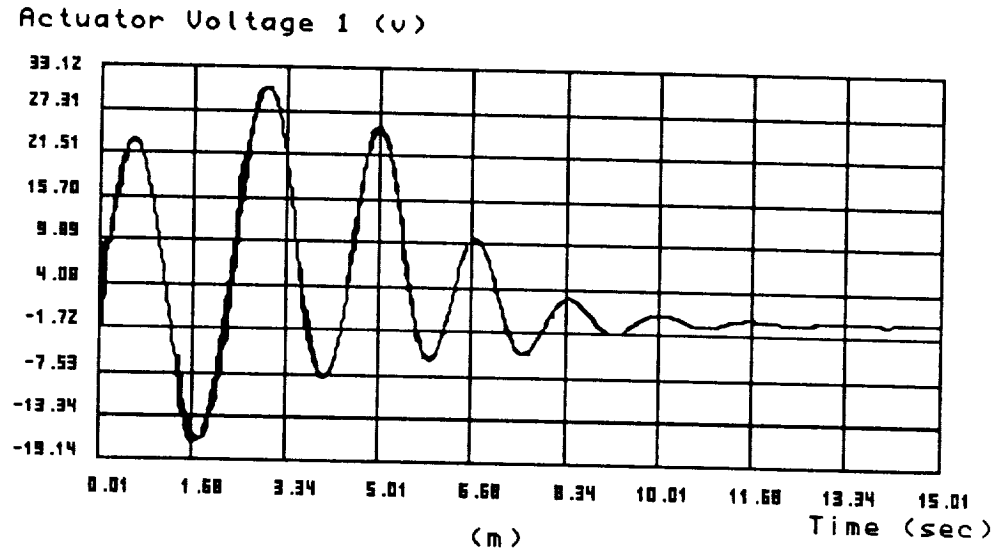


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



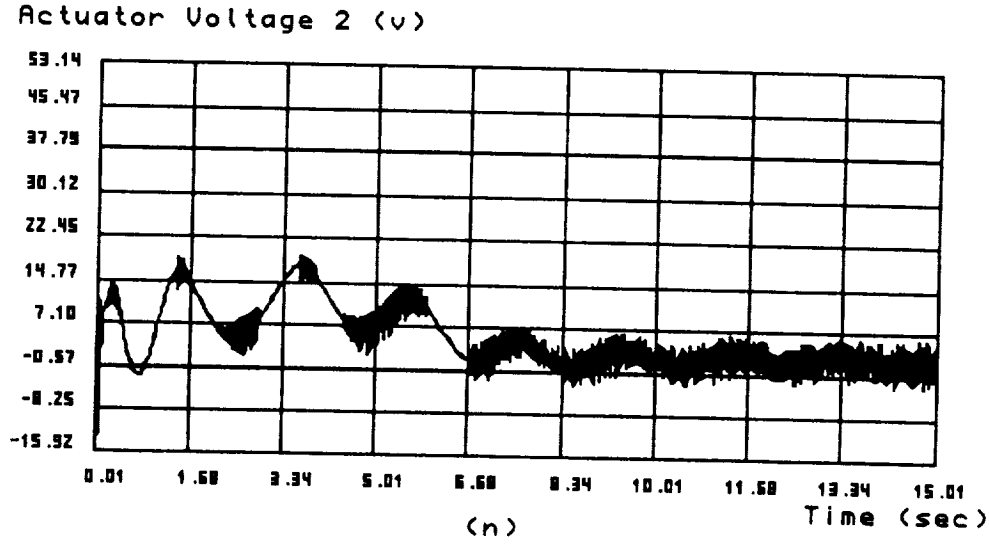
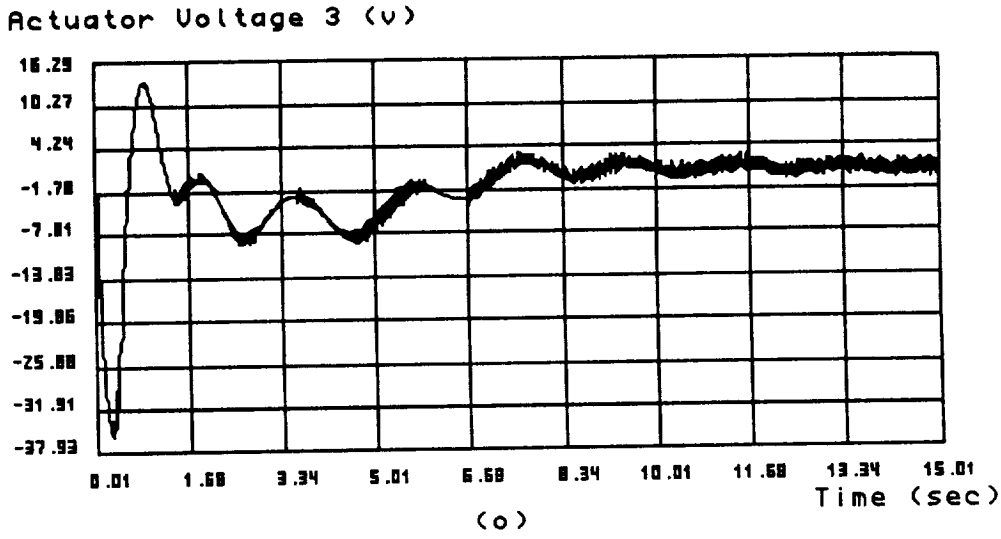


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



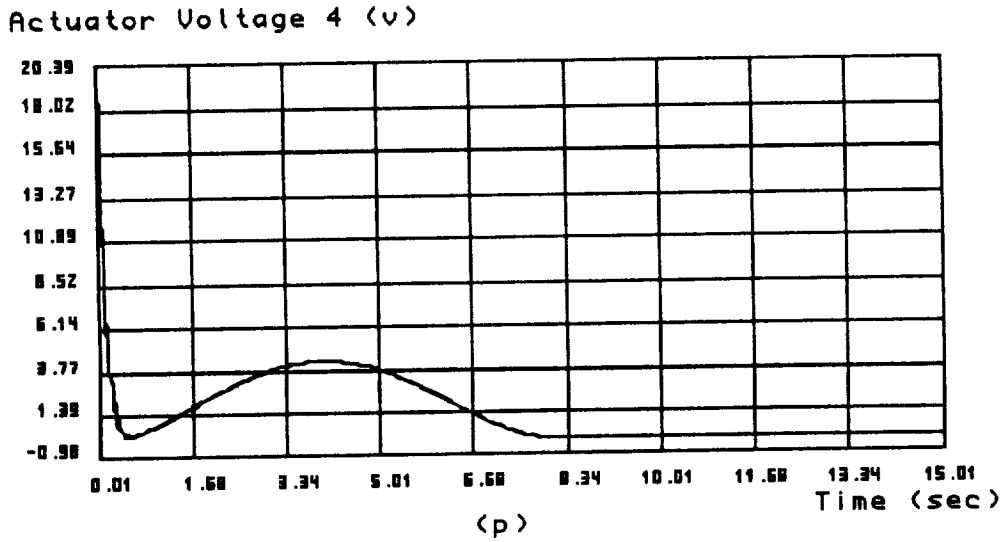
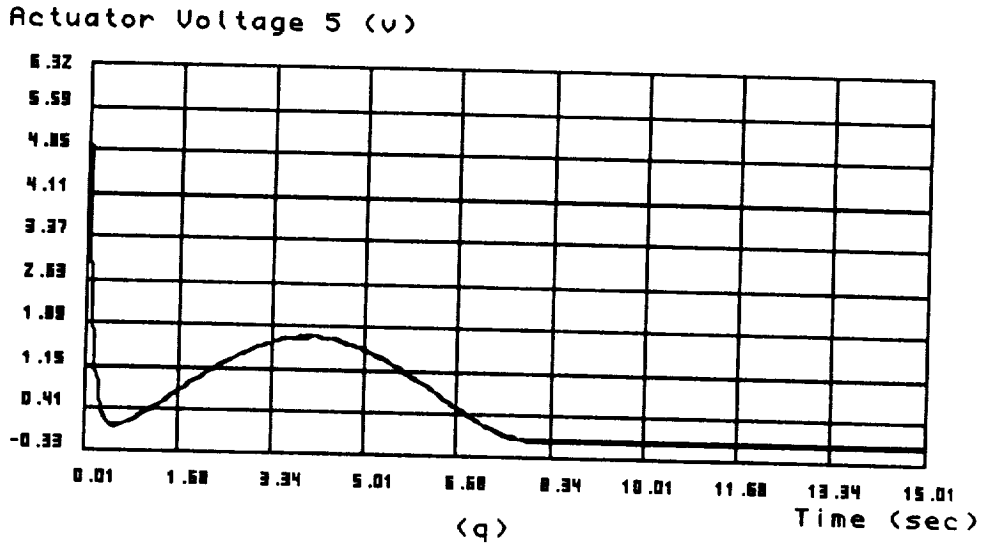


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)



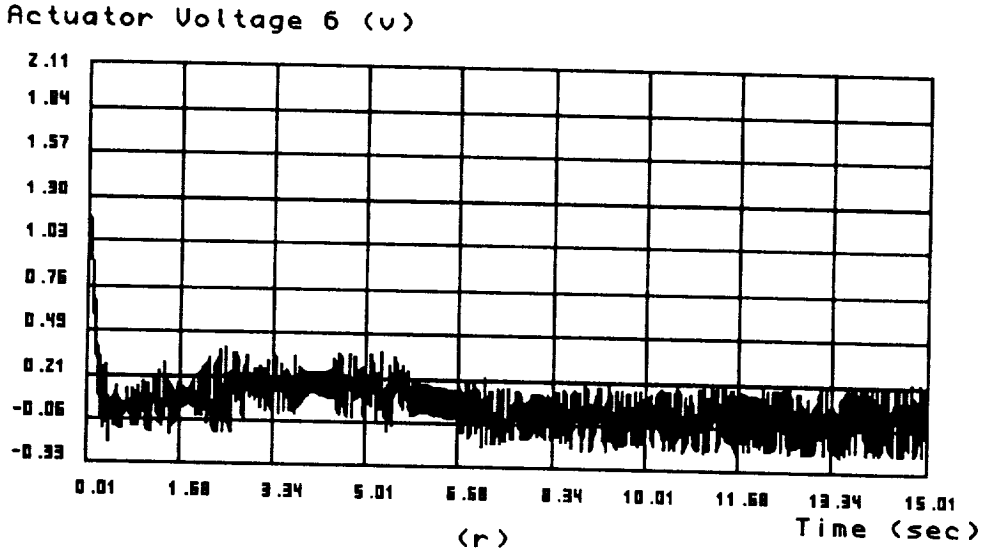


Figure 5.6: Six-Compliant-Joint Model Simulation Results (With Voltage Bound)

5.9.2 Case 2: Three Joint and Four Link Compliances

Figure 5.7 shows the model of the second example manipulator whose first three joints are compliant and modeled by the torsional spring depicted in Figure 5.2. Besides the joint compliances, the forearm and upper arm are modeled as flexible links whose flexibilities are approximated by orthogonal springs shown in Figure 5.3. Therefore, seven vibratory modes are modeled in this example, which are denoted by $\beta_1, \cdots, \beta_7$ separately as indicated in Figure 5.7. Since $n_\beta > n_\theta$ in this example, the γ structure defined in Equation 5.50 is employed to construct the control command. The stiffness matrix values for this case are assigned as

$$K = \text{diag}[452000\ 339000\ 226000\ ;\ (3500000)_4]\ \text{Nm/rad}\,;\text{N/m}$$

where the same stiffness value is assigned to all translational springs. The first three joint stiffnesses are in Nm/rad, while the last four are link stiffnesses in N/m. The reference trajectories used in the last example are used again in this simulation. Similar payload error is also tested here. In order to emphasize nominal motion tracking, the eigenvalues for all nominal joint displacements are assigned to $(-3\ -4\ -5)$, and to $(-1\ -2\ -3)$ for all vibratory modes. The corresponding diagonal feedback gain matrices are given by

$$K_p = [(-47)_6\,(-11)_7]\quad K_v = [(-12)_6\,(-6)_7]\quad K_I = [(60)_6\,(6)_7]$$

where the first six diagonal elements are the feedback gains associated with nominal motion, and the last seven diagonal elements are that of vibratory modes. The Q and P submatrices used in this case are

$$\begin{array}{lll} Q_1 = [(10000)_6\,(1)_7] & Q_2 = [(5000)_6\,(1)_7] & Q_3 = [(100000)_6(1)_7] \\ P_1 = [(19380)_6\,(2)_7] & P_2 = [(654.8)_6\,(.15)_7] & P_3 = [(-25774)_6\,(-1.15)_7] \\ P_4 = [(262.9)_6\,(.11)_7] & P_5 = [(-833.3)_6\,(-.08)_7] & P_6 = [(78452)_6\,(1.82)_7] \end{array}$$

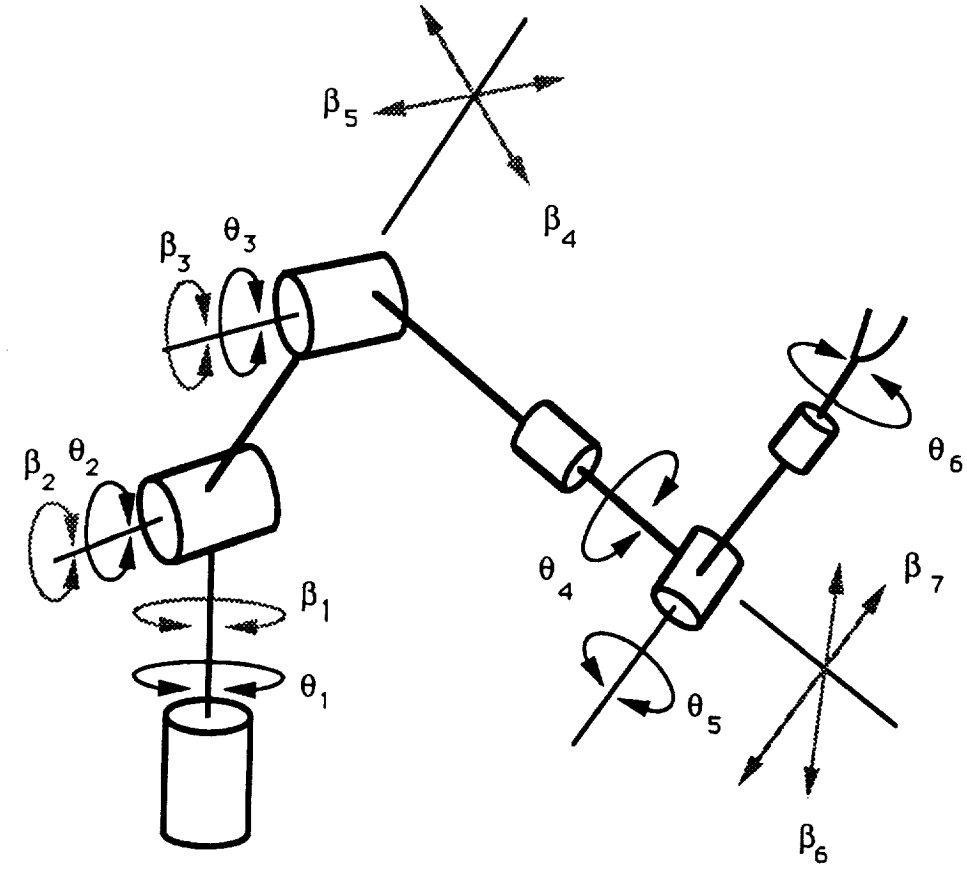
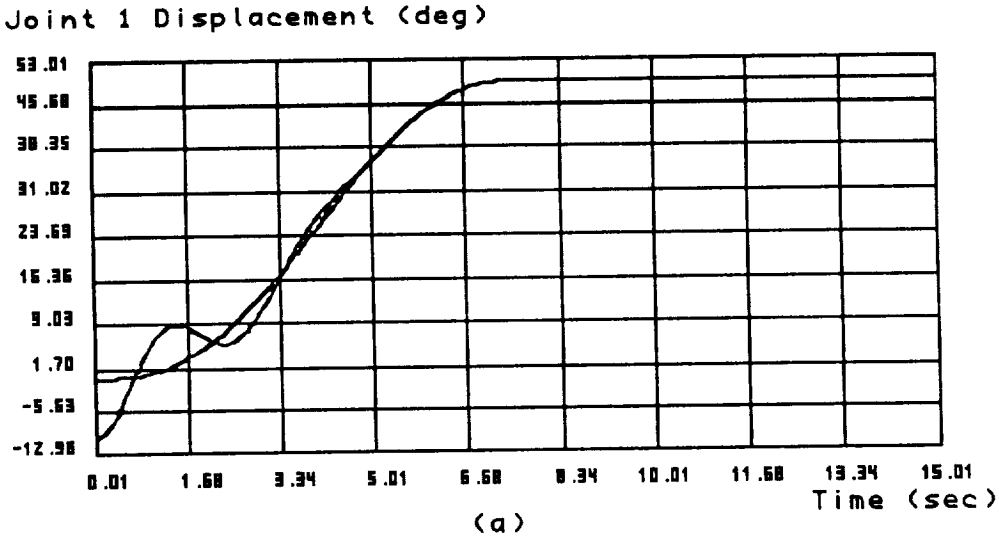


Figure 5.7: A Six-Link Robot Modeled with First Three Joints, Upper Arm, and Forearm Compliances

which have a large difference between the first six and the last seven diagonal elements to avoid a null η. The simulation results are displayed in the following figures. Figures 5.8(a) to 5.8(f) are the nominal displacements of joints 1 to 6. Notice that each joint starts with a positional error away from the reference trajectory. Figures 5.8(g) to 5.8(m) show the vibratory mode deflections along the actual trajectory. Figures 5.8(n) to 5.8(s) display the implemented voltages produced by the designed controller. Because of the emphasis on precise nominal tracking by choosing advantageous eigenvalues and P matrix, the nominal displacements converges satisfactorily to the desired trajectories even under initial position error and inaccurate payload information. In Figures 5.8(j) to 5.8(m), the link oscillations are bounded by decaying envelopes around the trace of static deflection. It is noticed that all vibratory modes are stabilized to negligible residual oscillations at the end of motion. The nonzero, steady state static deflections in Figures 5.8(h), (i), (j), and (l) are due to gravitational loading.

In order to test robustness of the proposed controller, a 30 % payload error was created in the above simulations. To constrict the test conditions, the payload error is further increased in this simulation. Now, the controller constructs control voltage for a 68 Kg payload while carrying an actual payload of 227 Kg which is more than three times of the assumed value. With this new payload change, the last simulation presented above is repeated and the results are shown in Figures 5.9(a) to (s). It appears that stability is maintained in this case but joint 1, 2, and 3 displacements in Figures 5.9(a) to (c) are affected by the large payload difference. This phenomenon could be explained from a comparison of the new payload error (159 Kg) with the system parameters listed in Table 5.2. Since the error is of the same order of magnitude as the mass of links 1, 2, and 3, it creates an impact on the

motion of the first three joints. Notice that the wrist motion shown in Figures 5.9(d) to (f) is seldom affected by the payload error in this case. That is because the wrist motion is affected primarily by the moment of inertia of a given payload. In this example, mass is the only payload error and due to a short moment arm between the gripper and the wrist, payload error creates little disturbance on wrist control. The effect of payload uncertainty on wrist motion control will be presented in the next chapter.



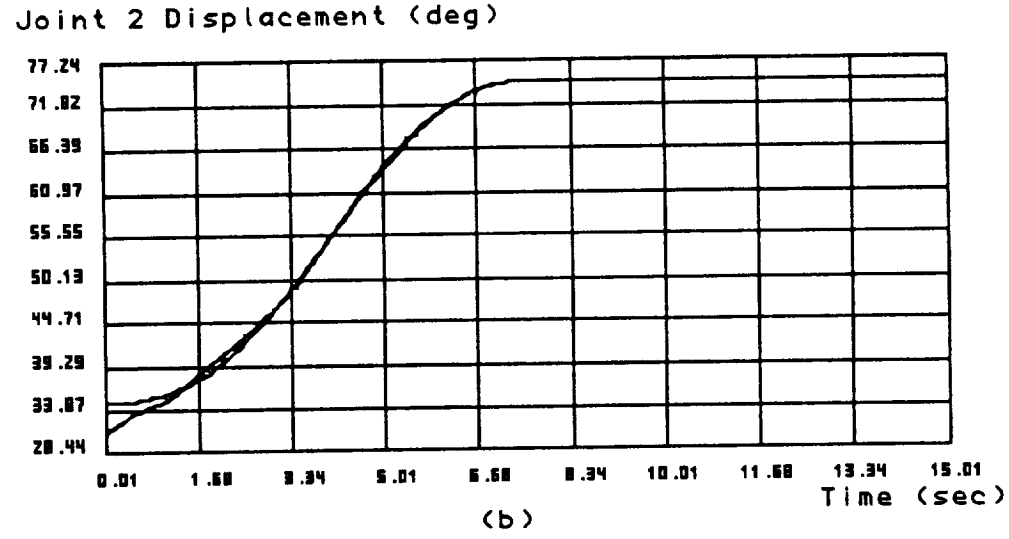
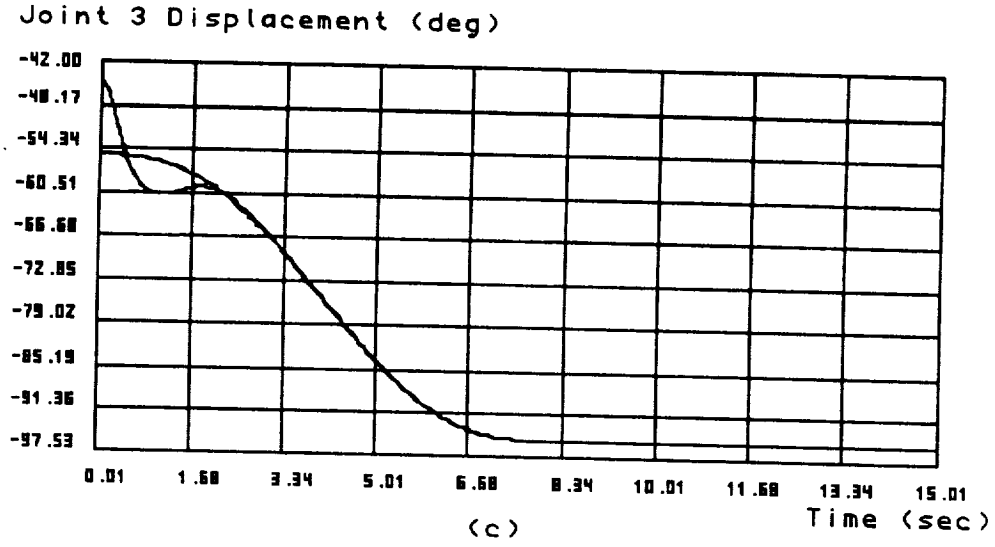


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



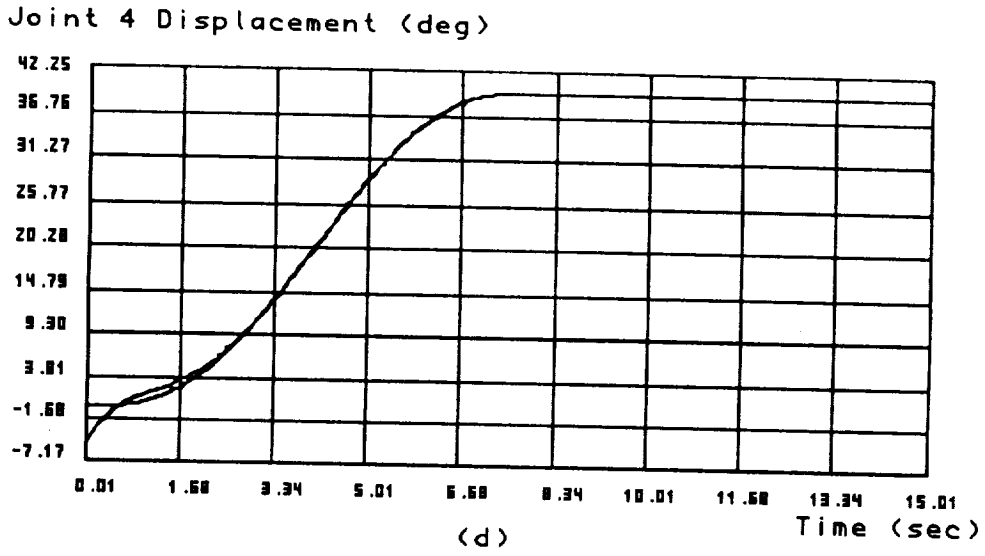
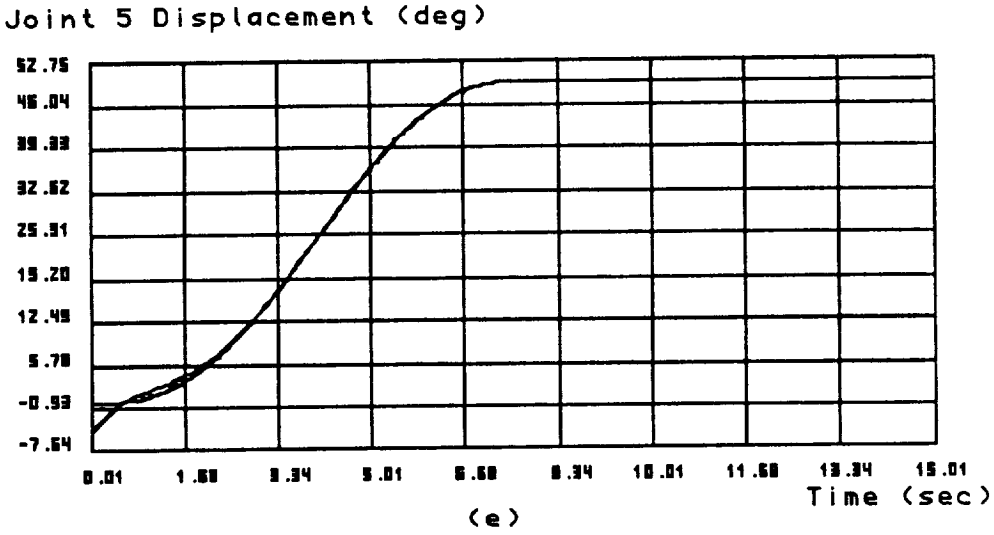


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



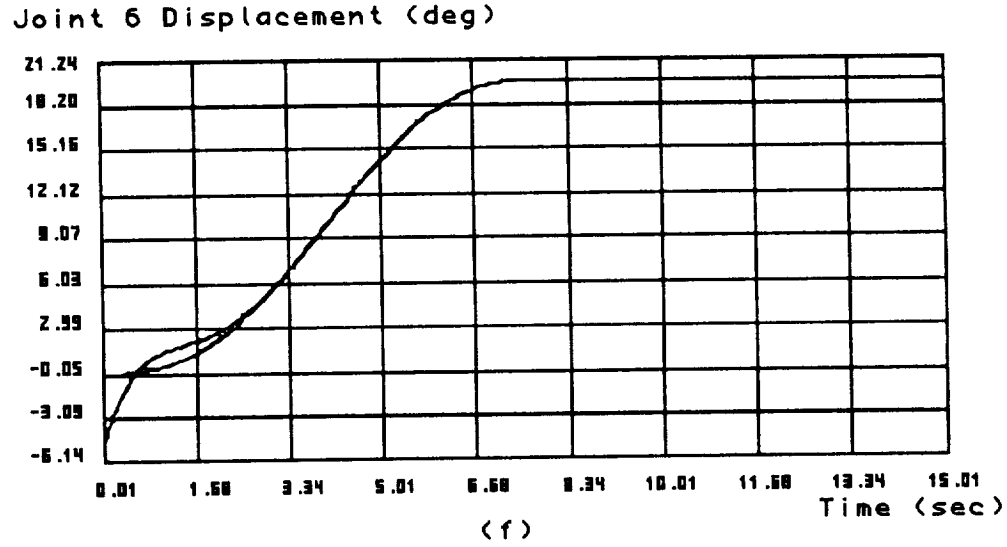
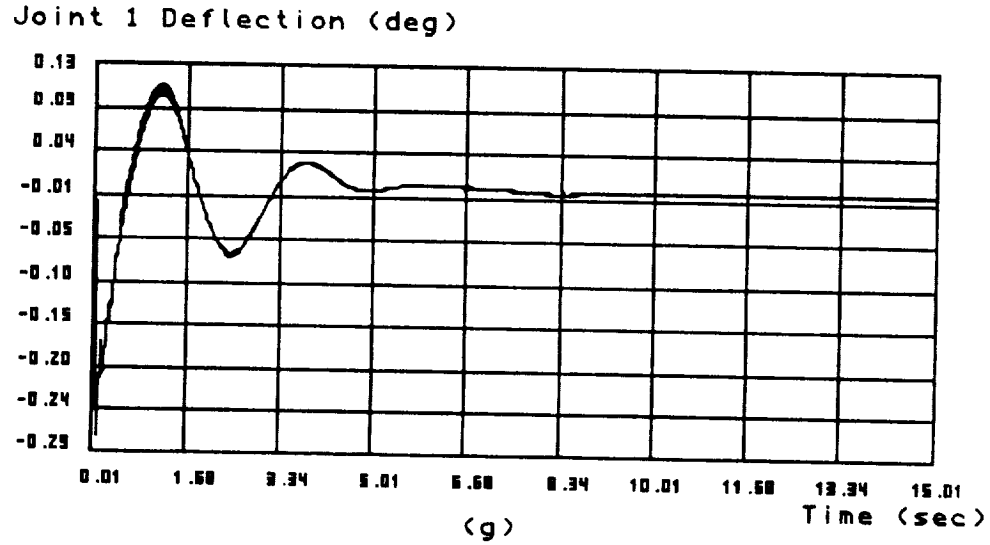


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



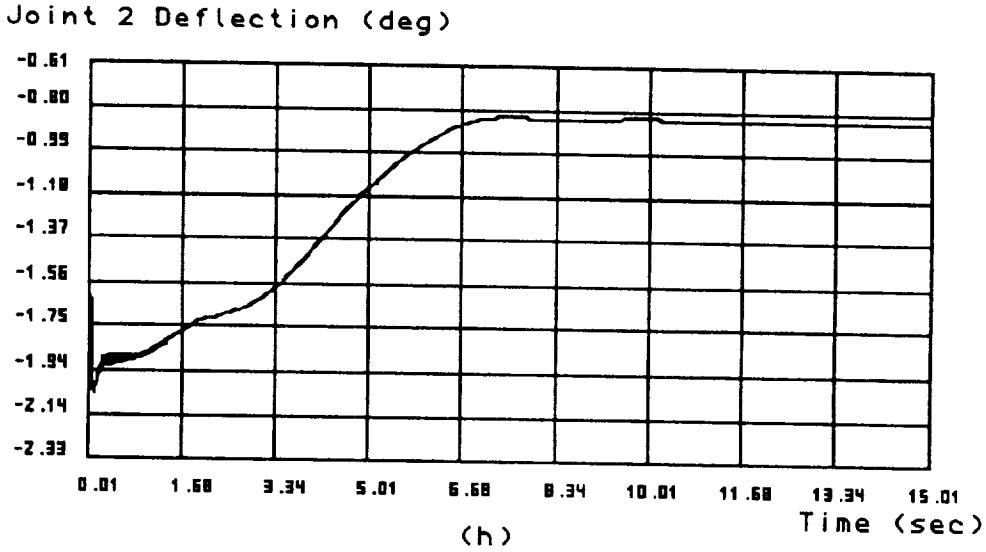
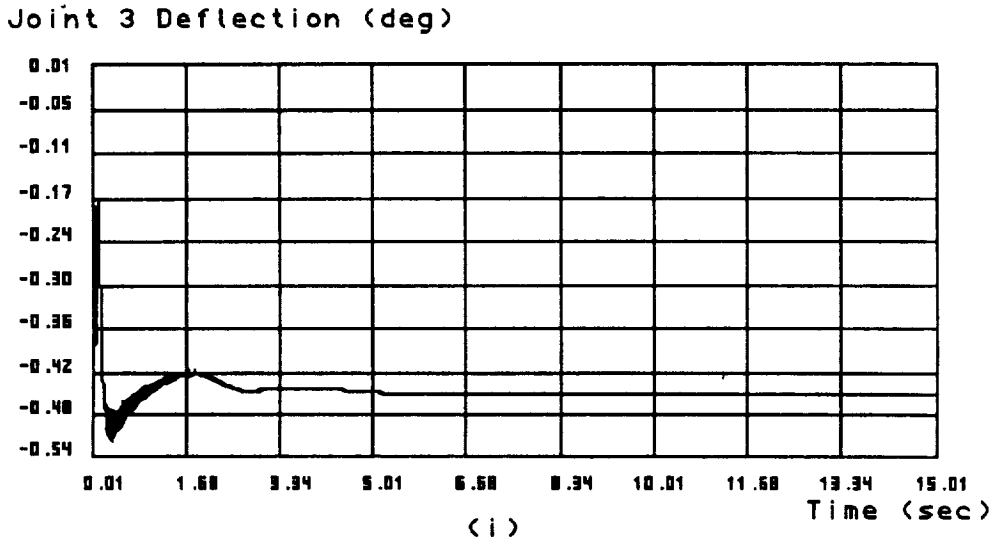


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



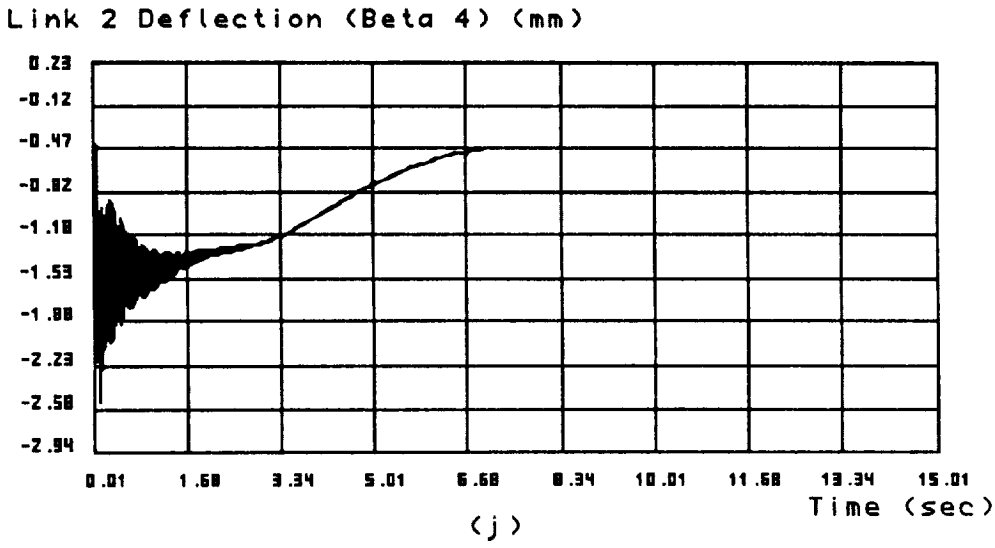
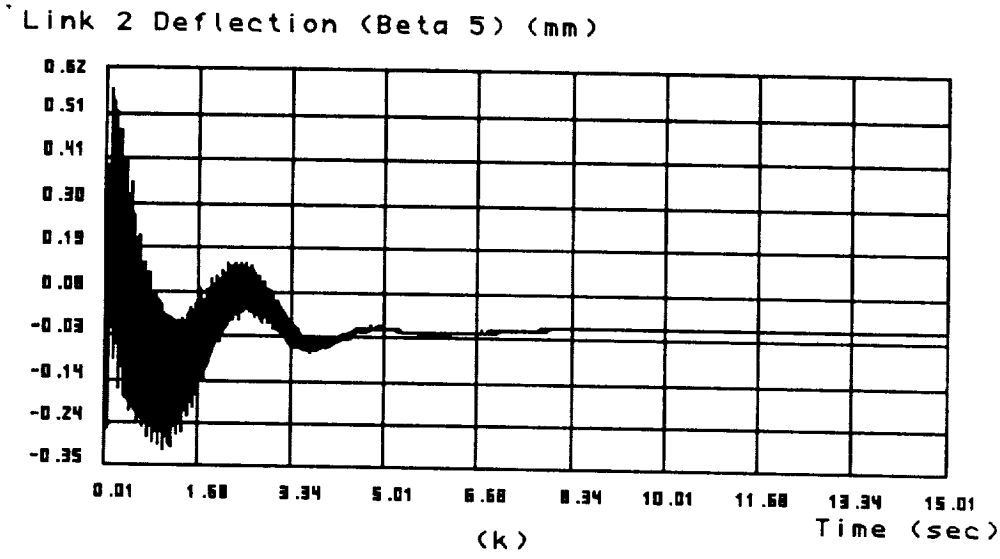


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



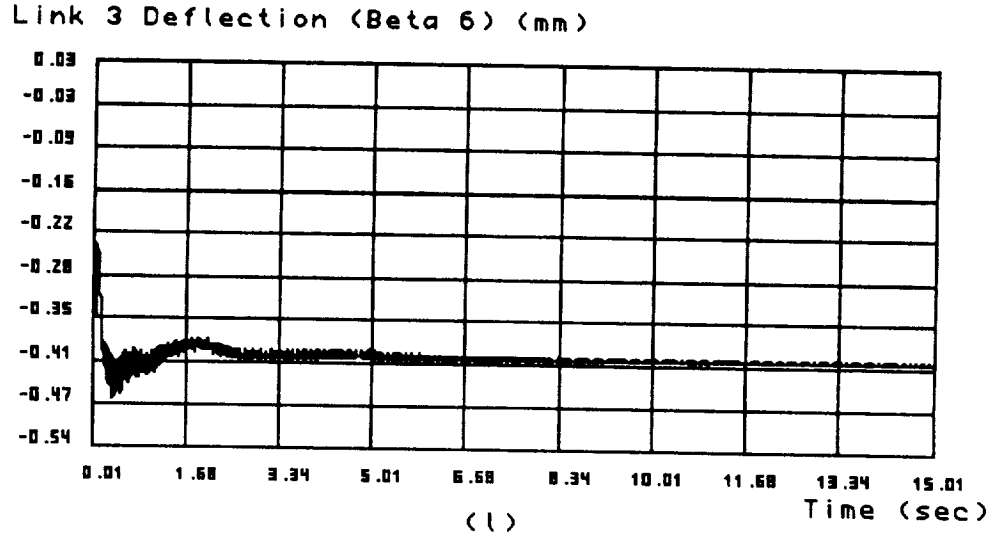
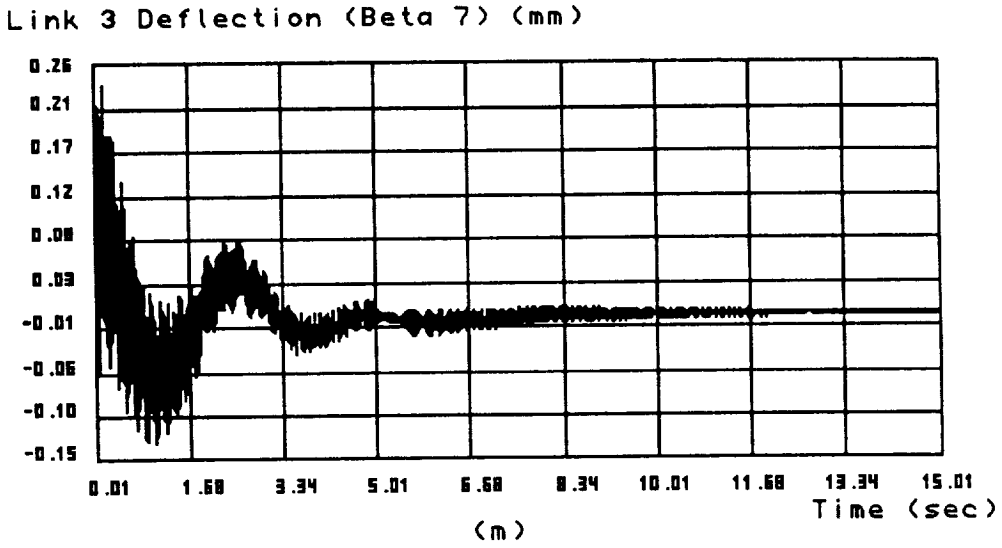


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



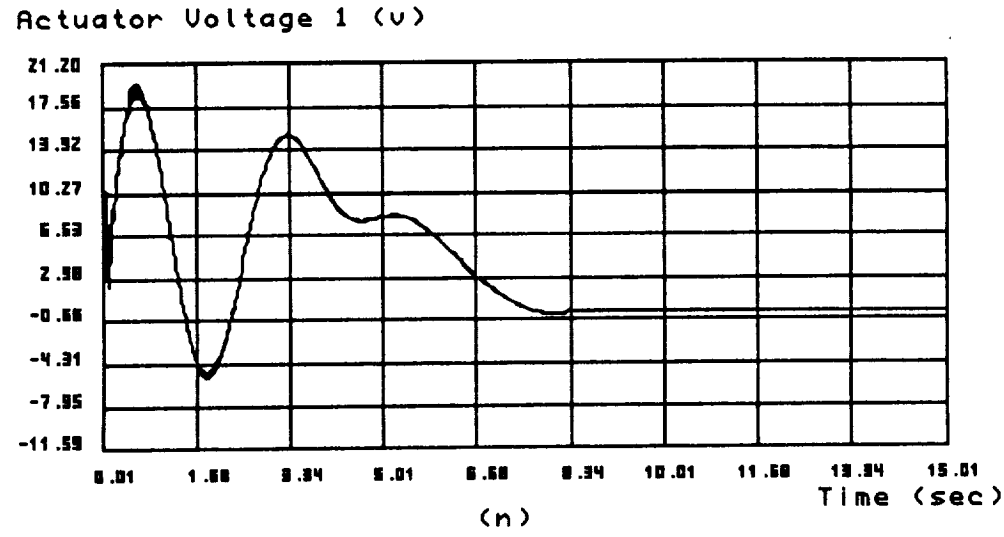
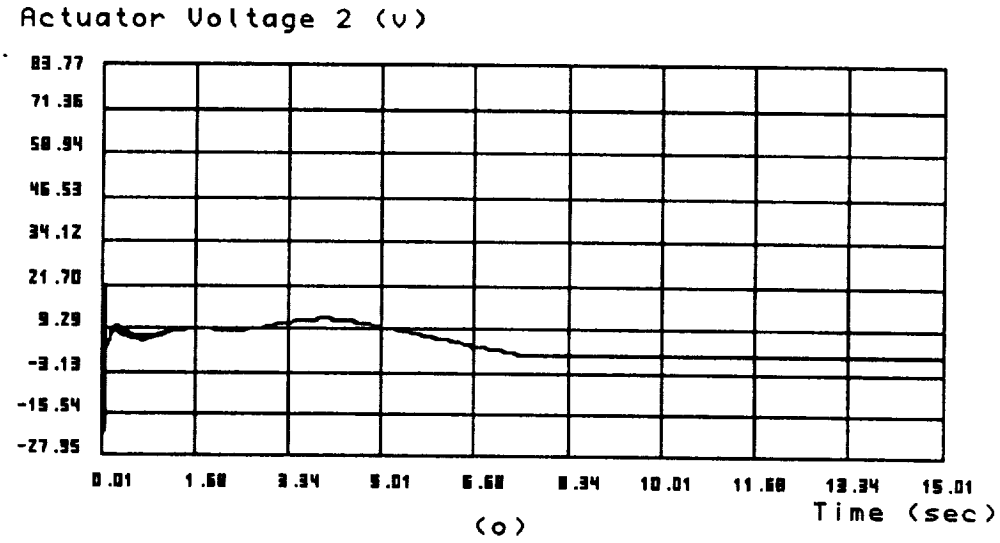


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



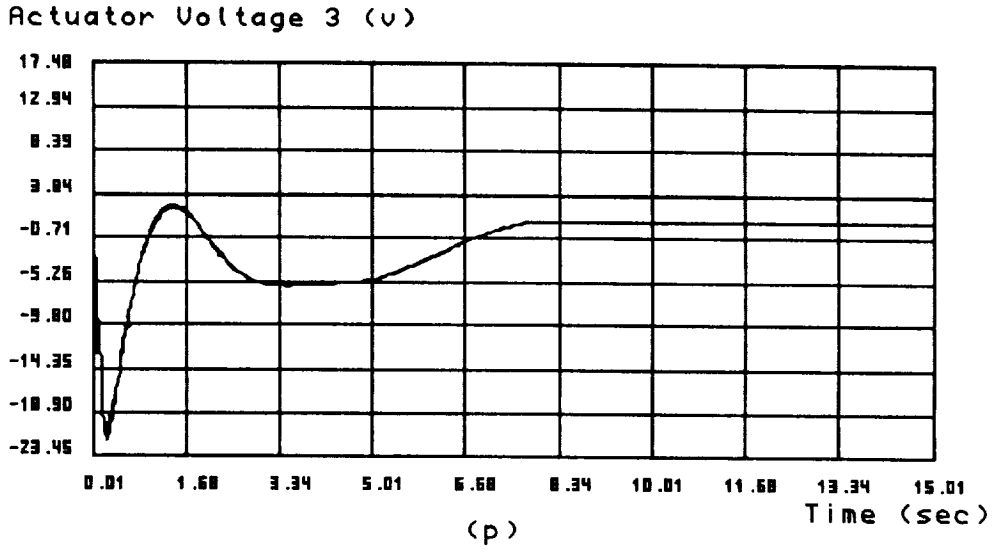
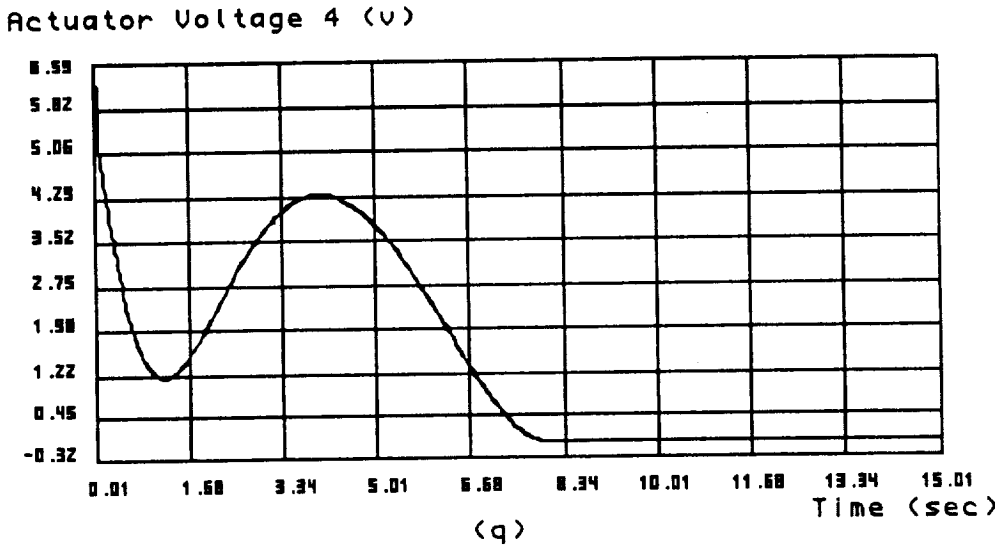


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



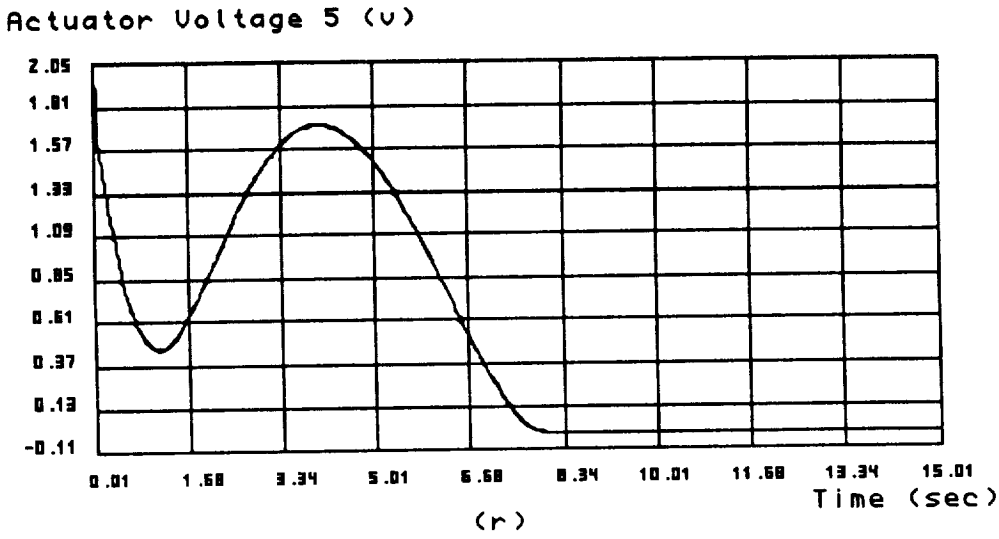


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)

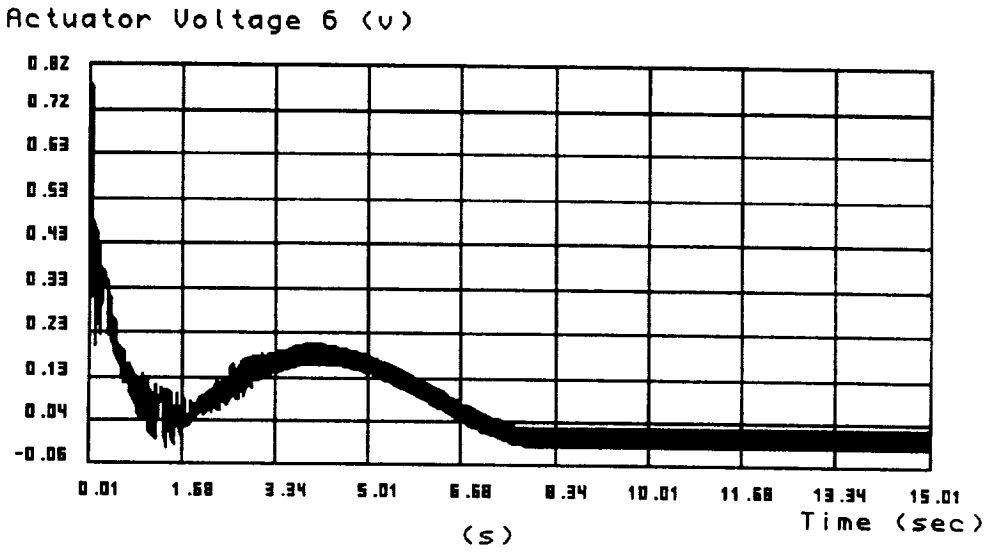
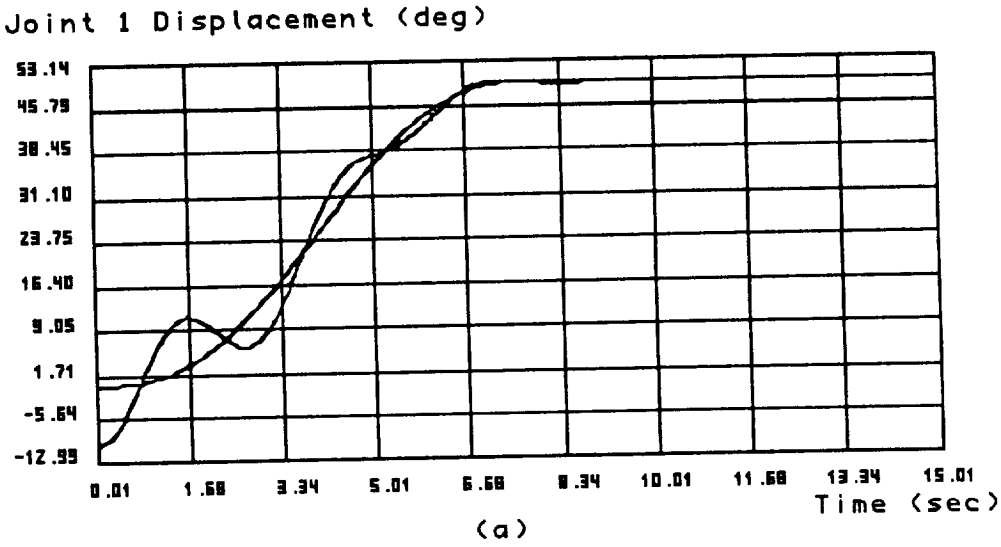


Figure 5.8: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (30% Payload Error)



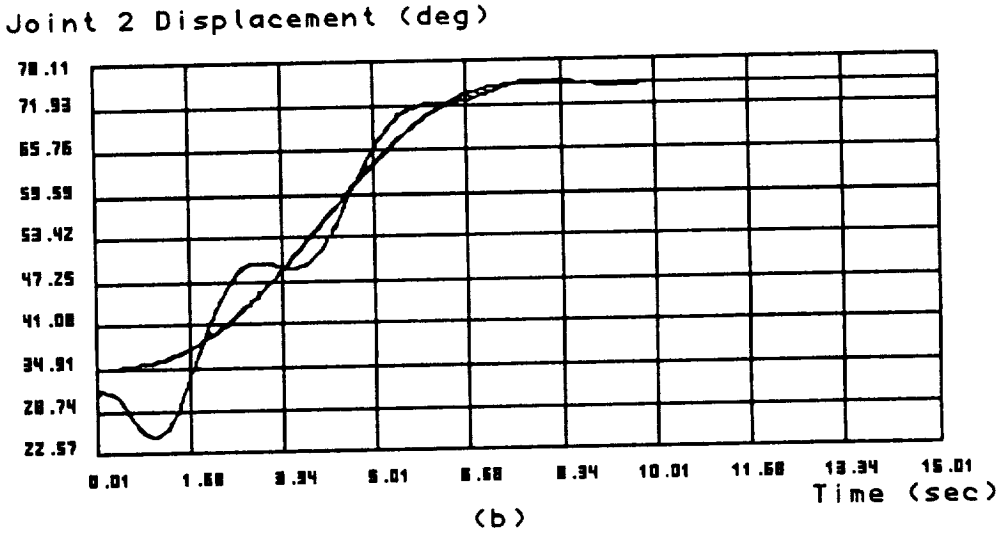
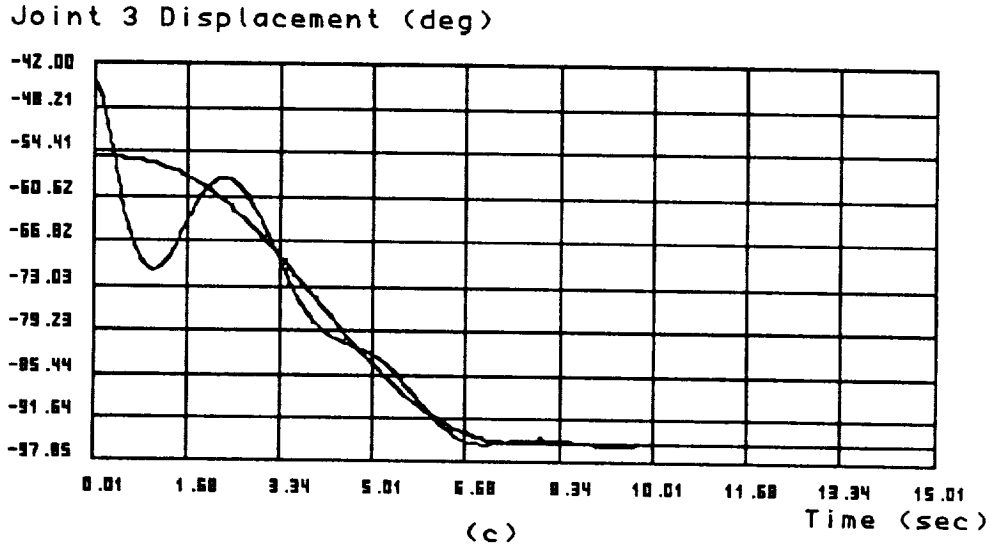


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



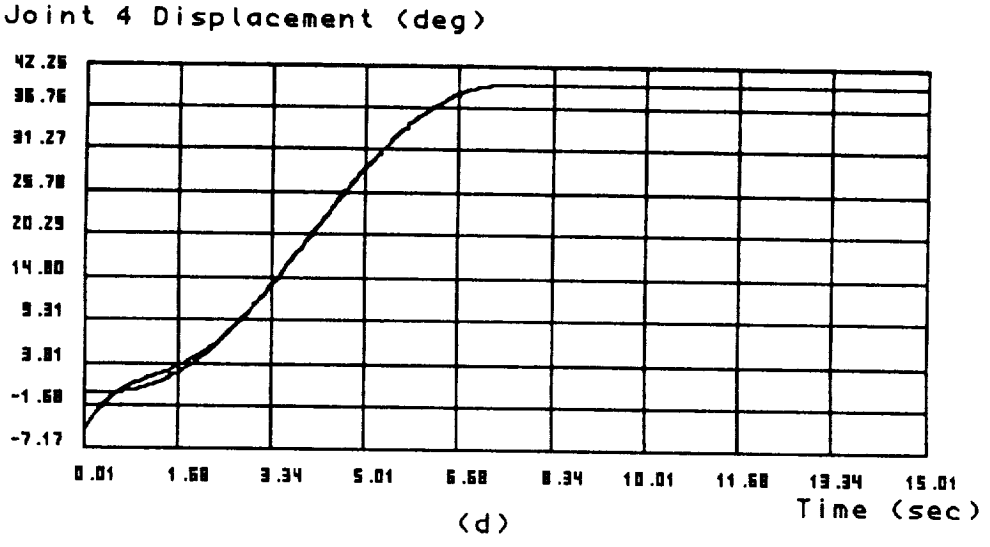
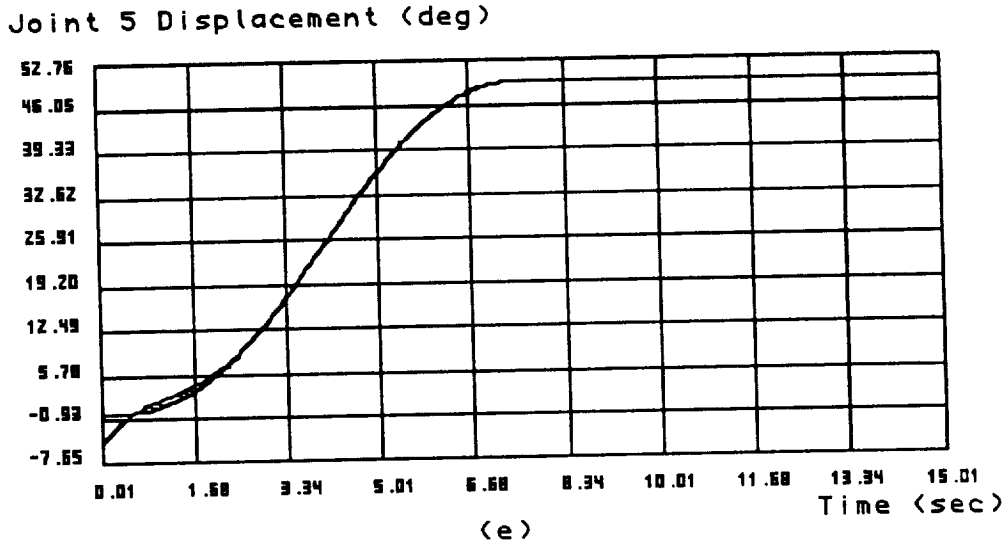


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



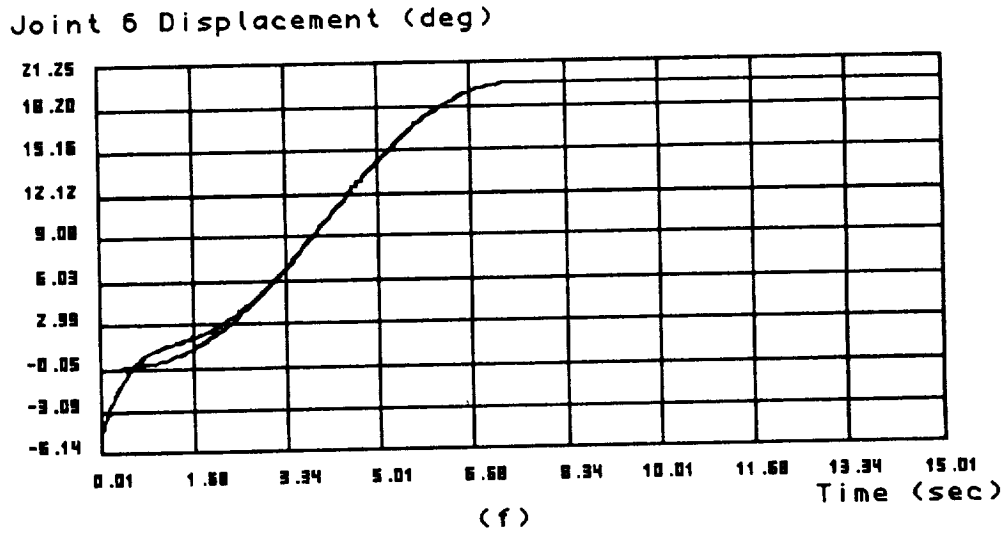
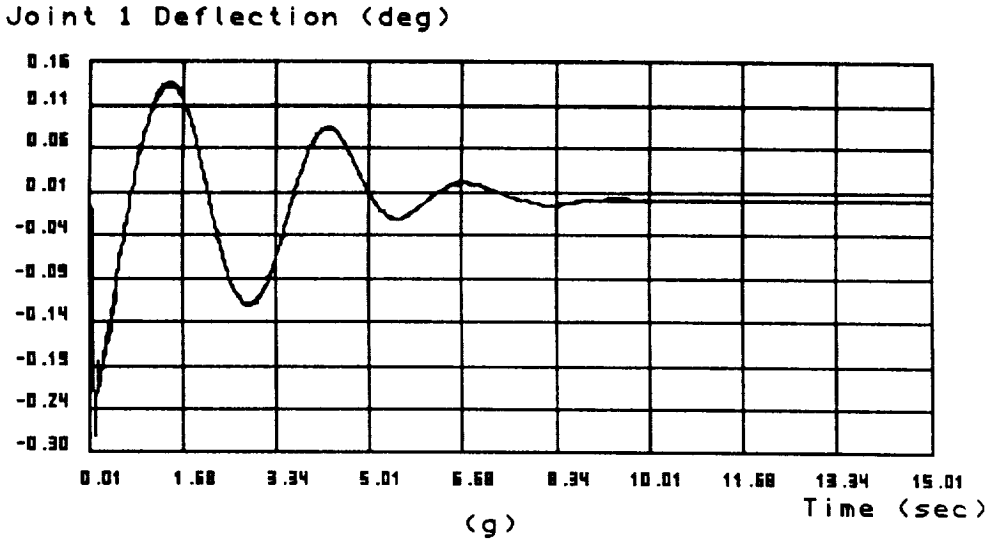


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



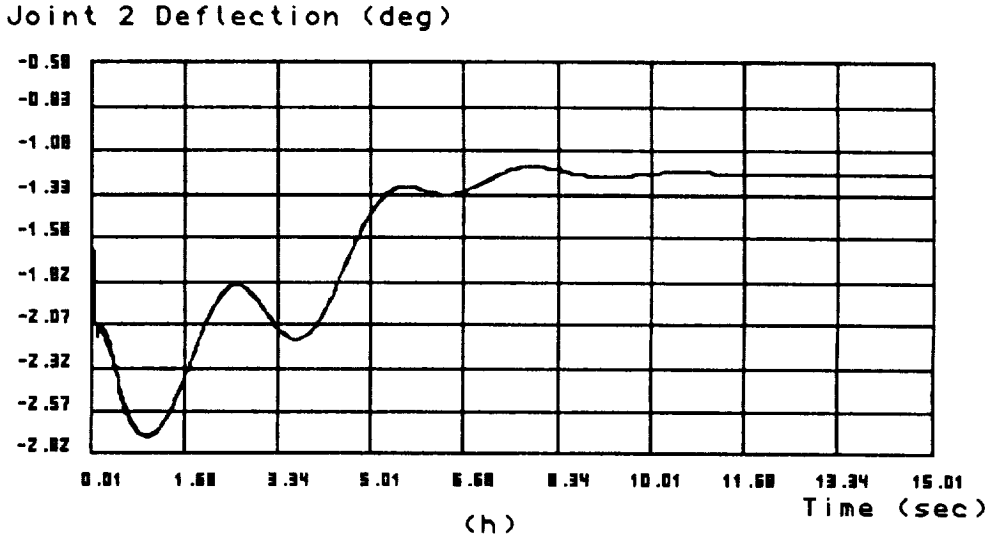
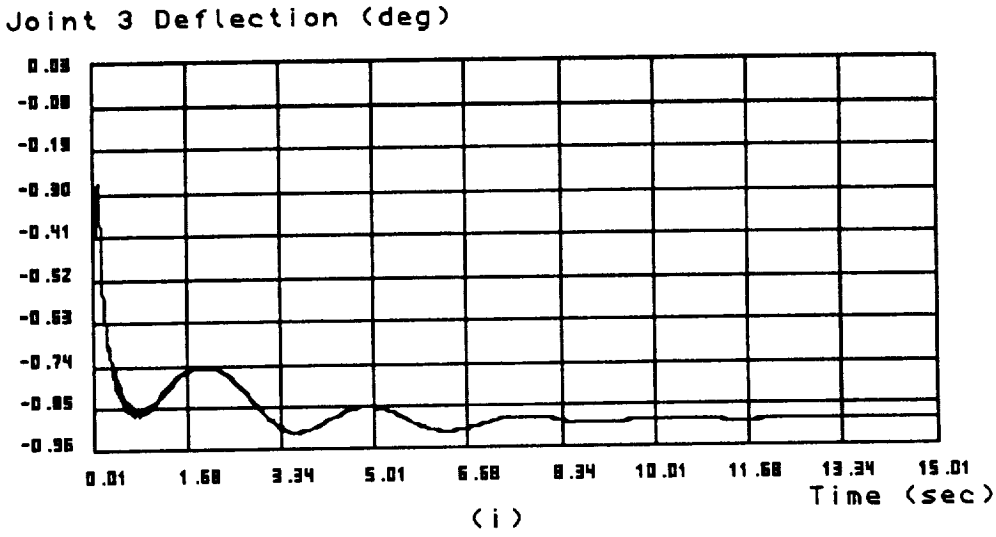


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



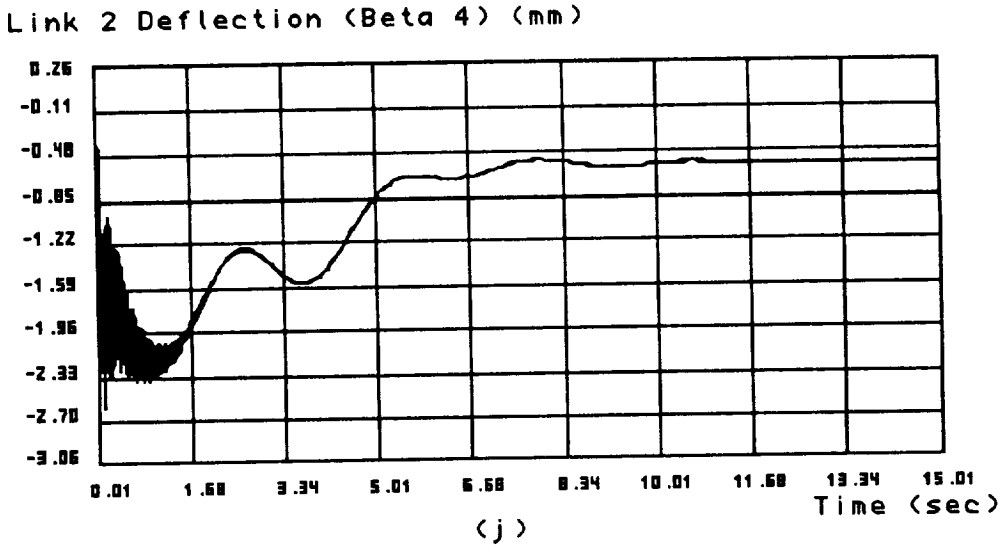
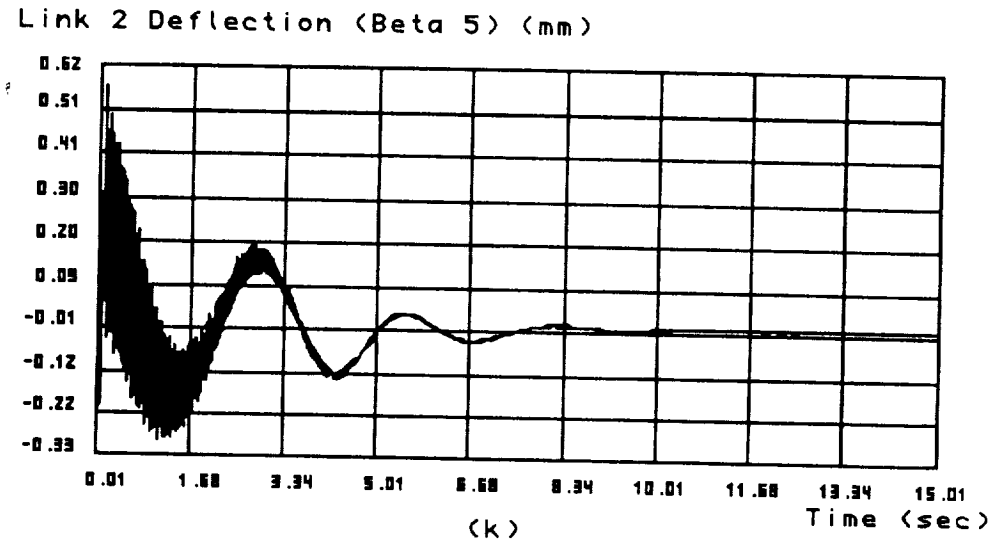


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



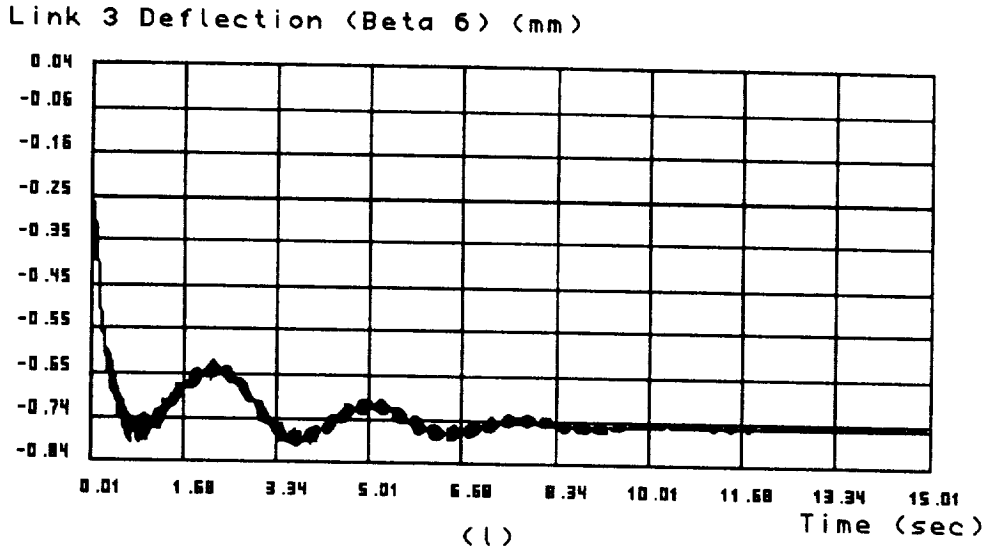
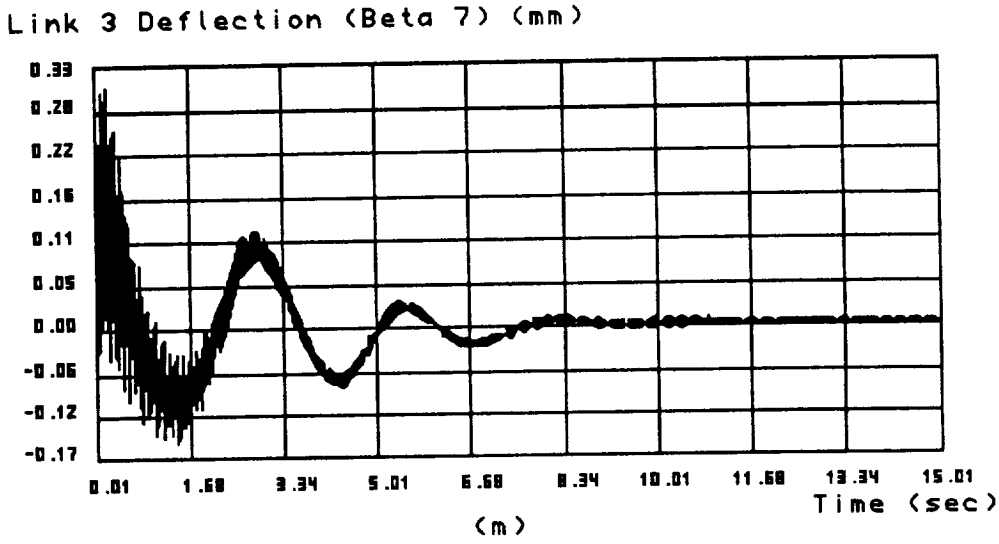


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



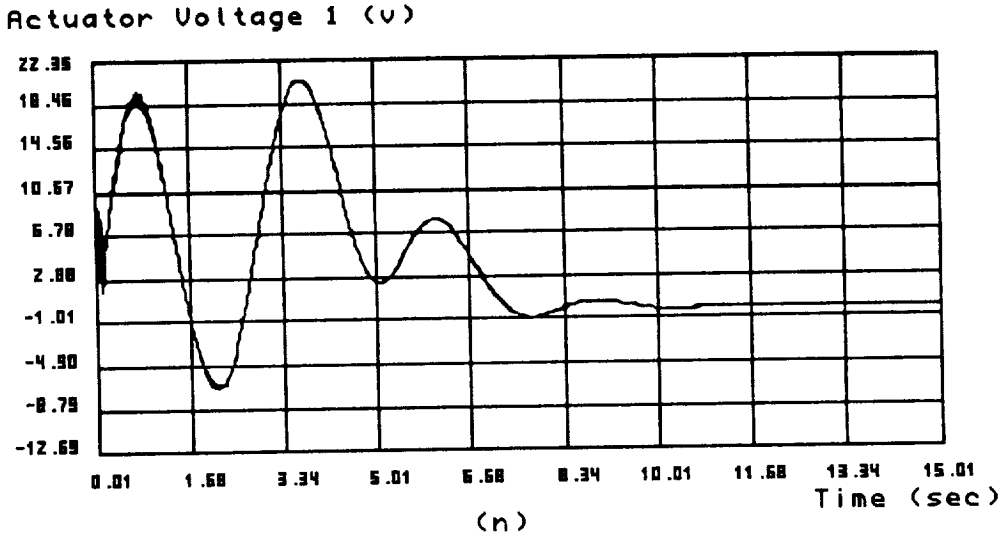
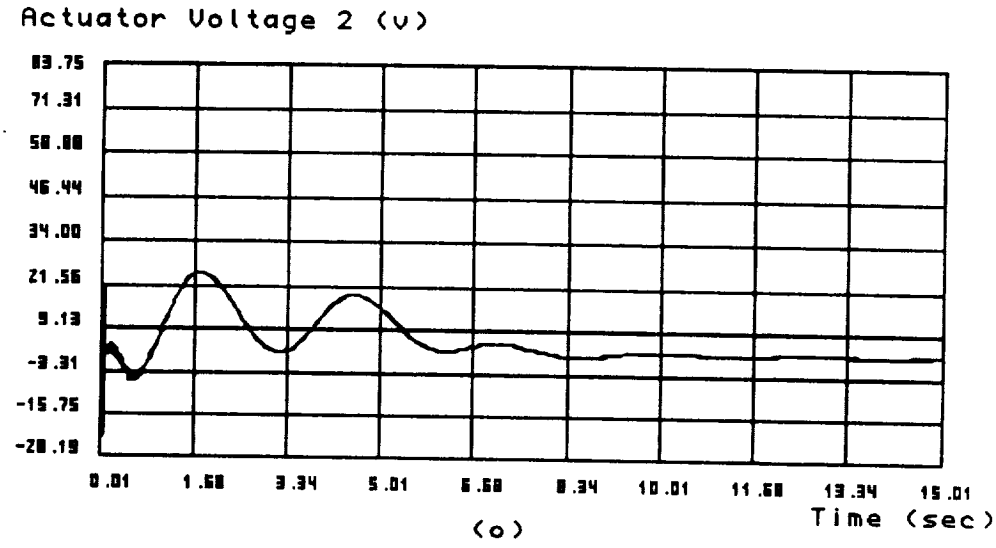


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



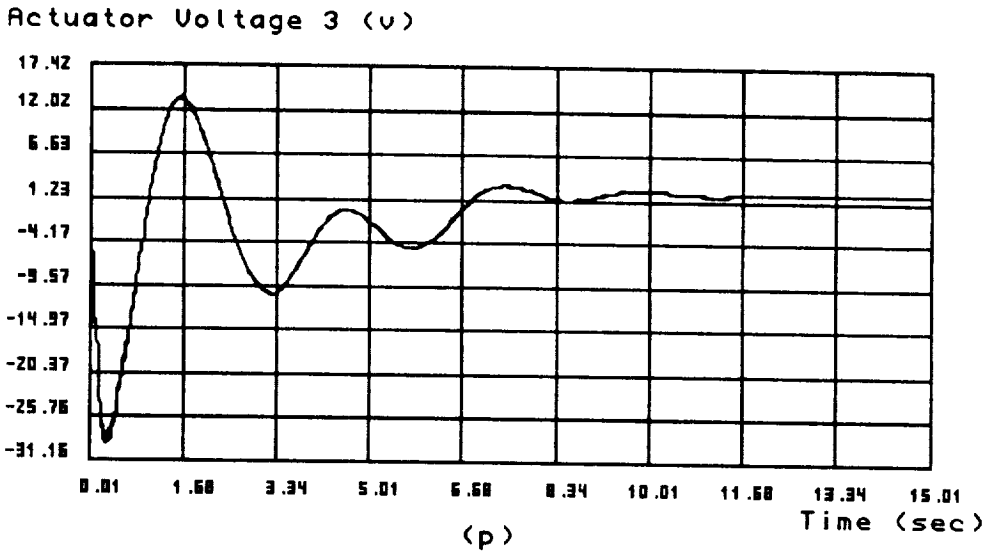
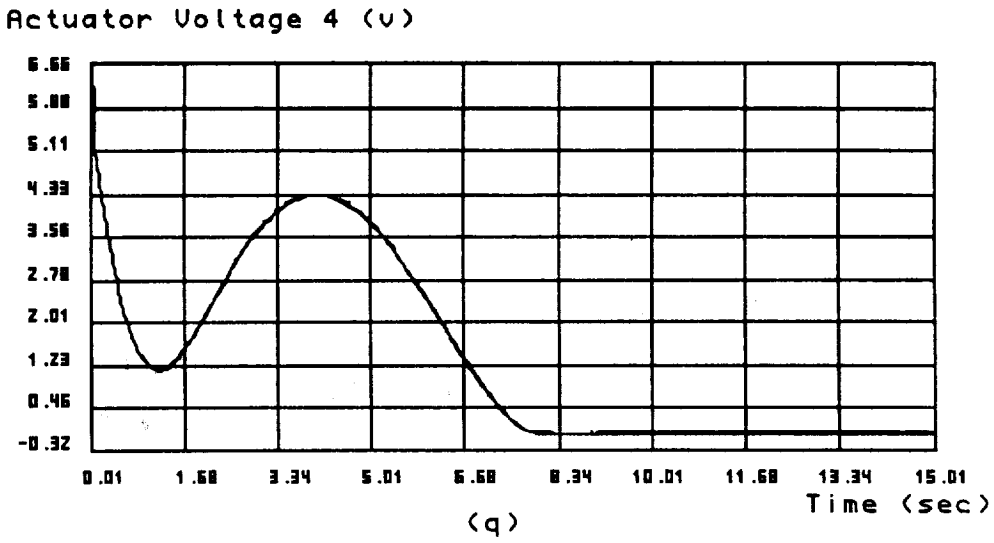


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)



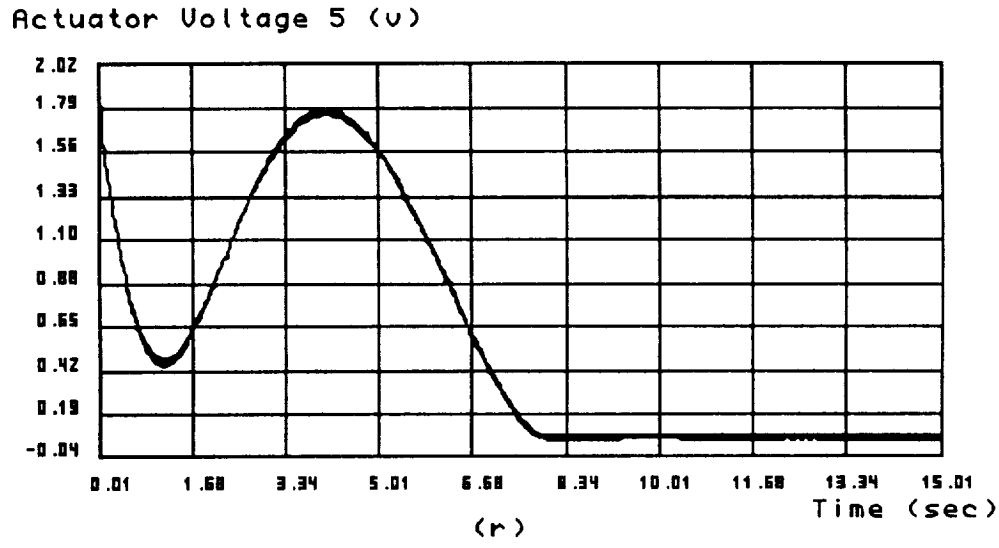


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)

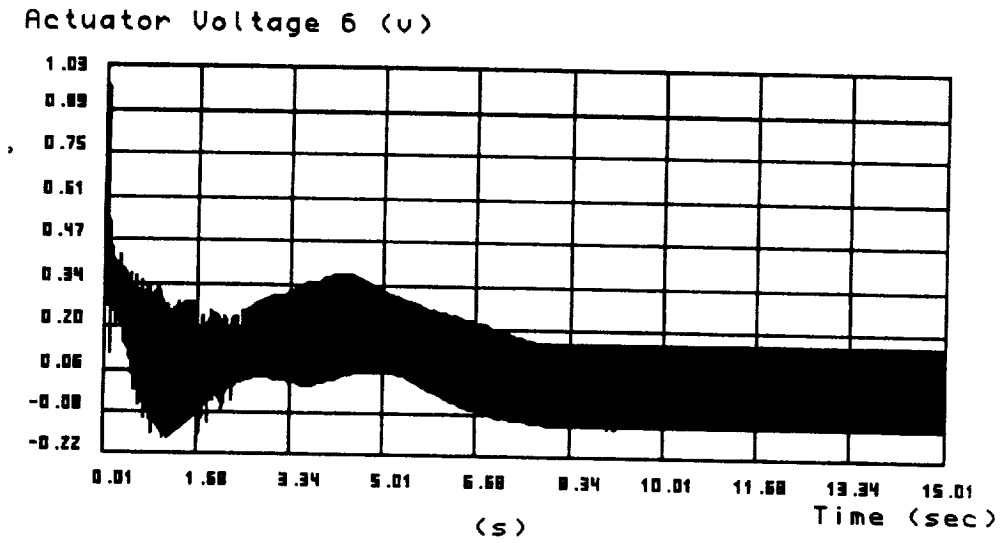


Figure 5.9: Simulation Results of the Six-Link Robot Modeled with First Three Joint, Upper Arm, and Forearm Compliances (Large Payload Error)

5.9.3 Case 3: A Compliant Wrist and Four Link Compliances

In the T3-776 robot shown in Figure 5.1, the forearm is designed as a lever rotating around the third joint, or elbow. The wrist and wrist actuators are located at opposite ends of the forearm. This kinematic arrangement has the benefit of using wrist actuators to counterbalance the inertia of wrist and forearm. Inside the forearm, three slender, coaxial torsion tubes serve as driving shafts connecting the wrist to its actuators as shown in Figure 5.10. Due to the slenderness of the torsion tubes, the wrist is softer than the first three joints. The third example will model the torsion tube stiffnesses by three torsional springs and also include the forearm and upper arm flexibility in controller design. The example model is sketched in Figure 5.11, where the first three joints are rigid and $\beta_1, \cdots, \beta_7$ are the modeled joint and link vibratory displacements. In this example, the influence of P values on system response as mentioned in the previous section will be demonstrated. The stiffness matrix is selected as

$$K = \text{diag}[(8500)_3\,;(3500000)_4]\ \text{Nm/rad}\,;\text{N/m}$$

where the first three values are wrist joint stiffness in Nm/rad and the last four values are lateral link stiffnesses in N/m. Similarly, the robot is controlled to follow the reference trajectories specified in the last example under the same initial position and 30 % payload errors. First, the same set of eigenvalues, feedback gain matrices, and P and Q matrices used in the last section are repeated in this case, which are

$$\begin{array}{lll} K_p = [(-47)_6\,(-11)_7] & K_v = [(-12)_6\,(-6)_7] & K_I = [(60)_6\,(6)_7] \\ Q_1 = [(10000)_6\,(1)_7] & Q_2 = [(5000)_6\,(1)_7] & Q_3 = [(100000)_6(1)_7] \\ P_1 = [(19380)_6\,(2)_7] & P_2 = [(654.8)_6\,(.15)_7] & P_3 = [(-25774)_6\,(-1.15)_7] \\ P_4 = [(262.9)_6\,(.11)_7] & P_5 = [(-833.3)_6\,(-.08)_7] & P_6 = [(78452)_6\,(1.82)_7] \end{array}$$

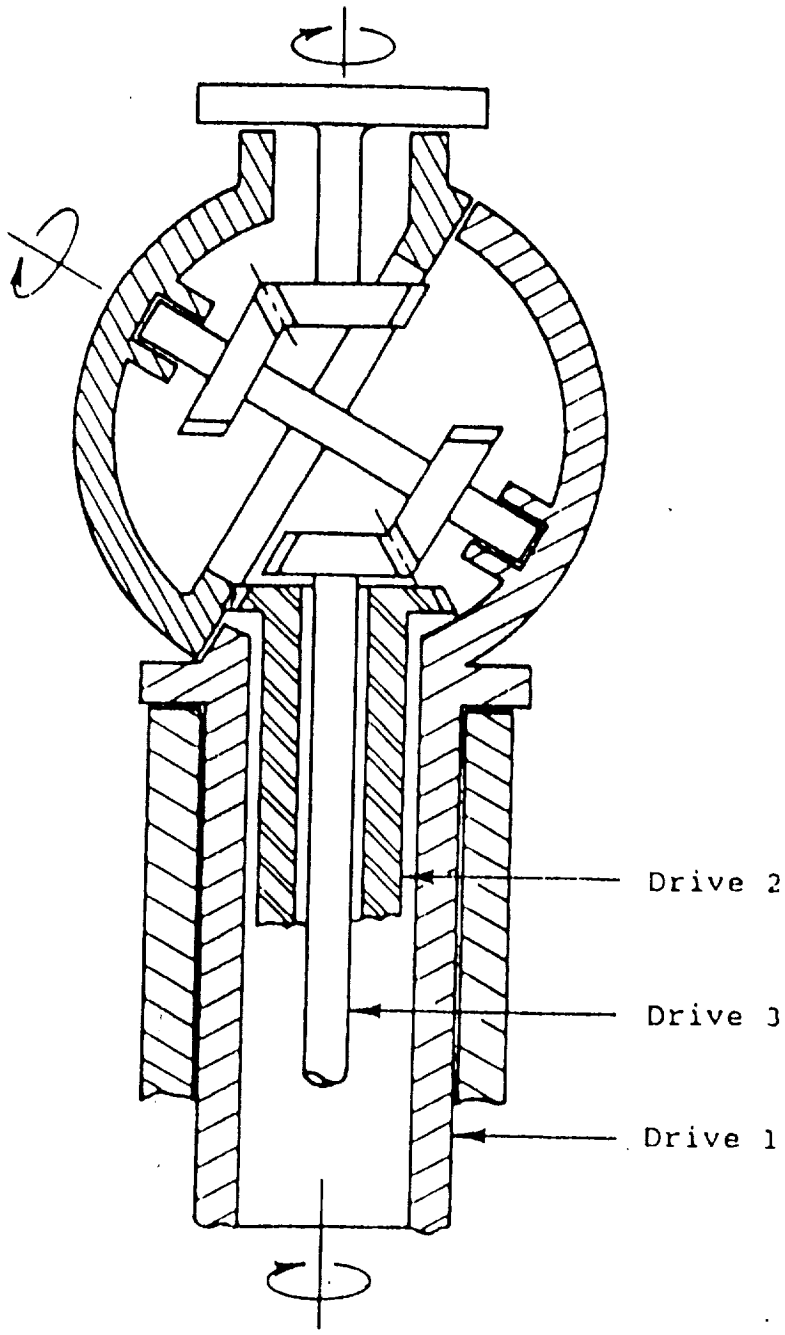


Figure 5.10: Cincinnati Milacron Three Degree-of-Freedom Wrist

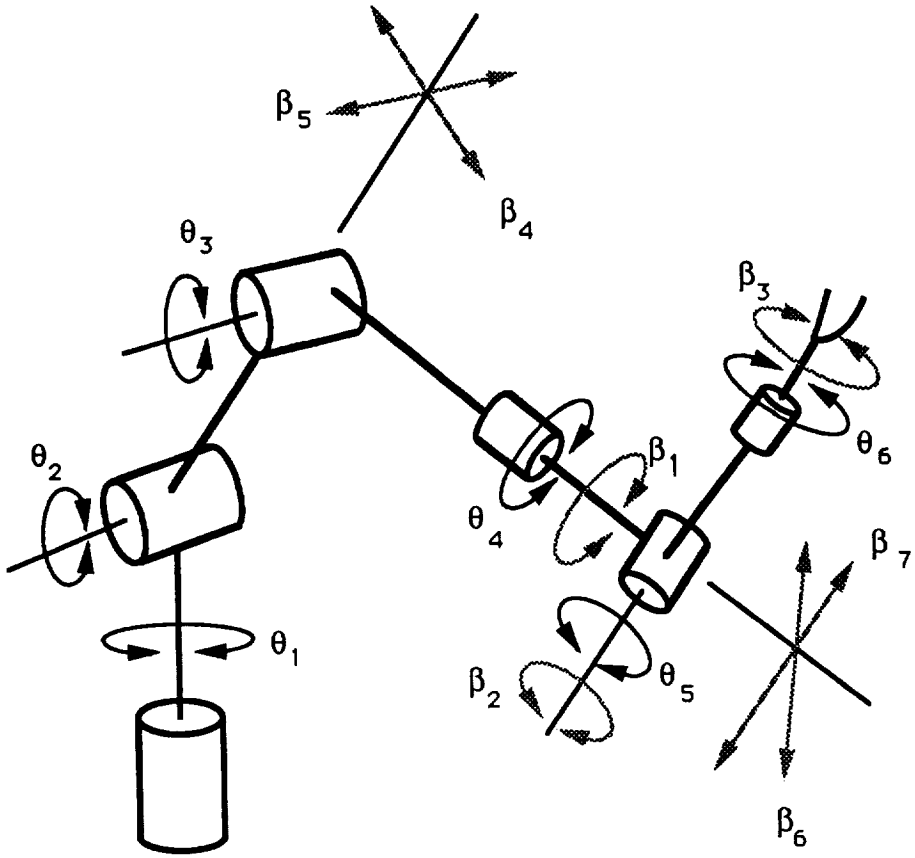


Figure 5.11: A Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances

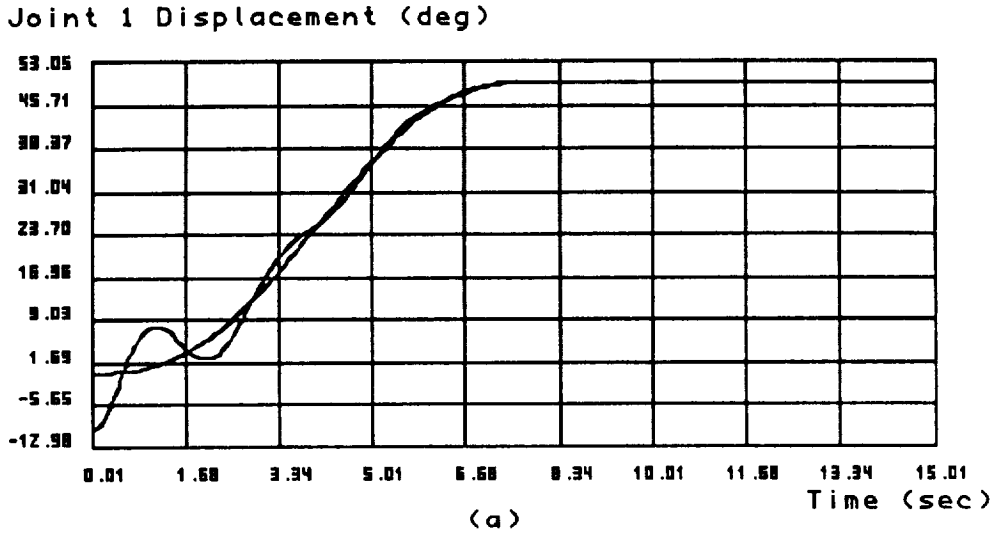
The simulation results are displayed in Figures 5.12(a) to 5.12(s) which show satisfactory performance as expected. Now, for comparison purposes, the last seven diagonal elements of Q_1, Q_2, and Q_3 submatrices are increased 100 times to

$$Q_1 = [(10000)_6\,(100)_7] \quad Q_2 = [(5000)_6\,(100)_7] \quad Q_3 = [(100000)_6(100)_7]$$

according to the linear relationship between P and Q, the P submatrices are then altered to

$$P_1 = [(19380)_6\,(200)_7] \quad P_2 = [(654.8)_6\,(15)_7] \quad P_3 = [(-25774)_6\,(-115)_7]$$
$$P_4 = [(262.9)_6\,(10.83)_7] \quad P_5 = [(-833.3)_6\,(-8.33)_7] \quad P_6 = [(78452)_6\,(181.67)_7]$$

This adjustment increases the last seven entries by a factor of 100 and reduces the difference between the nominal and vibratory mode portions in P. Simulation of the third model is repeated for the new P values and the results are plotted in Figures 5.13(a) to 5.13(s). The system response deteriorates drastically although the stability is maintained for the new P. This comparison shows that a proper selection of P values can significantly enhance controller performance. For example, in this case a large difference between the nominal and vibrational entries in P is preferable according to the previous successful examples. An analytical explanation for such results was presented in the previous section regarding the effect of the P matrix on system response.



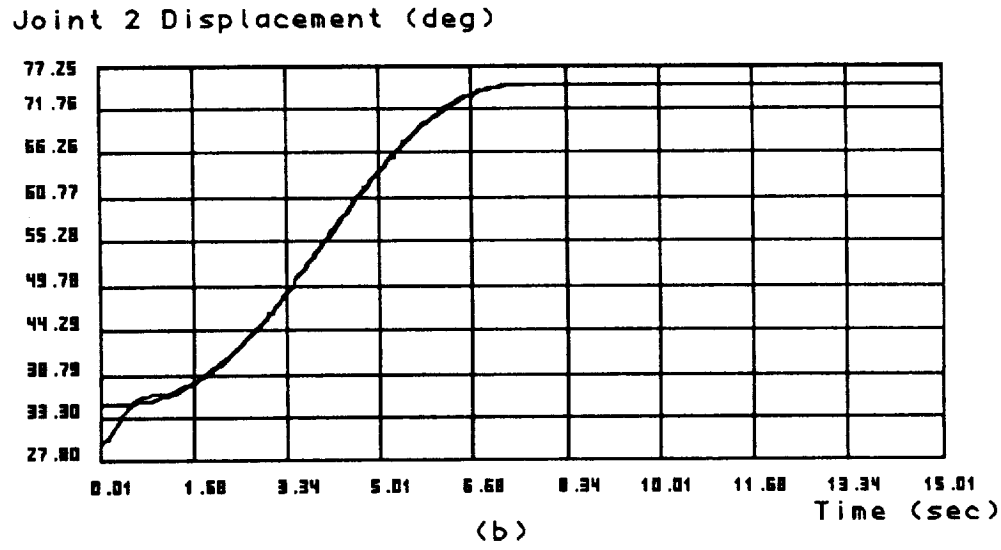
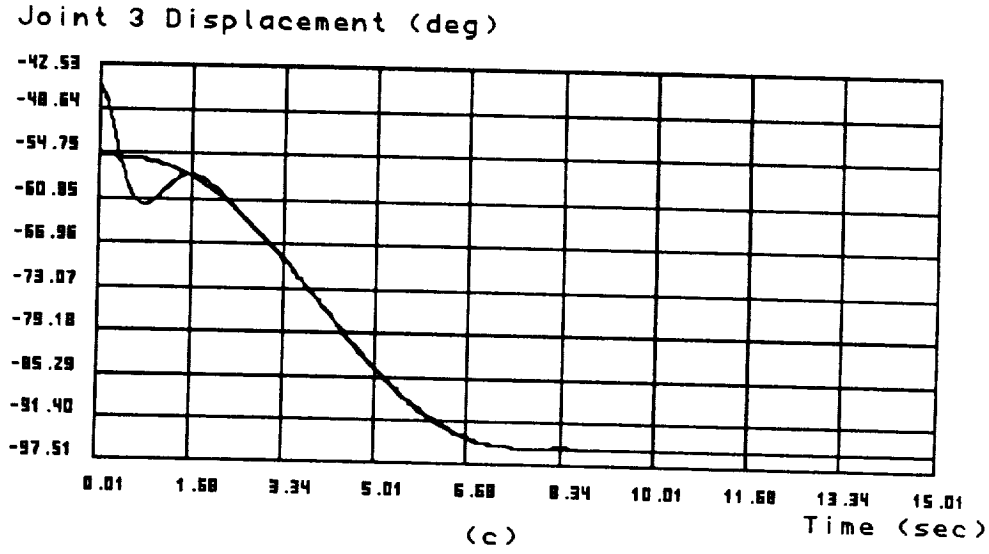


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



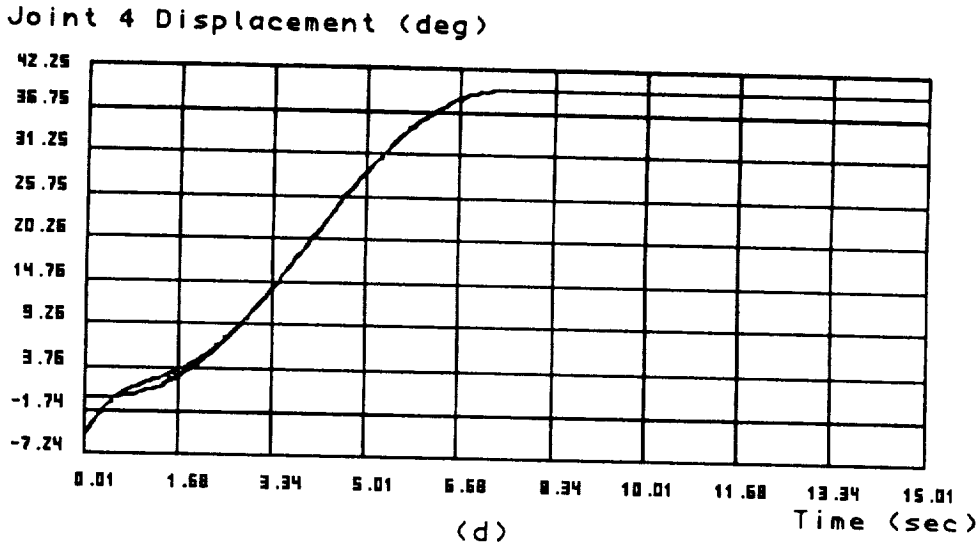
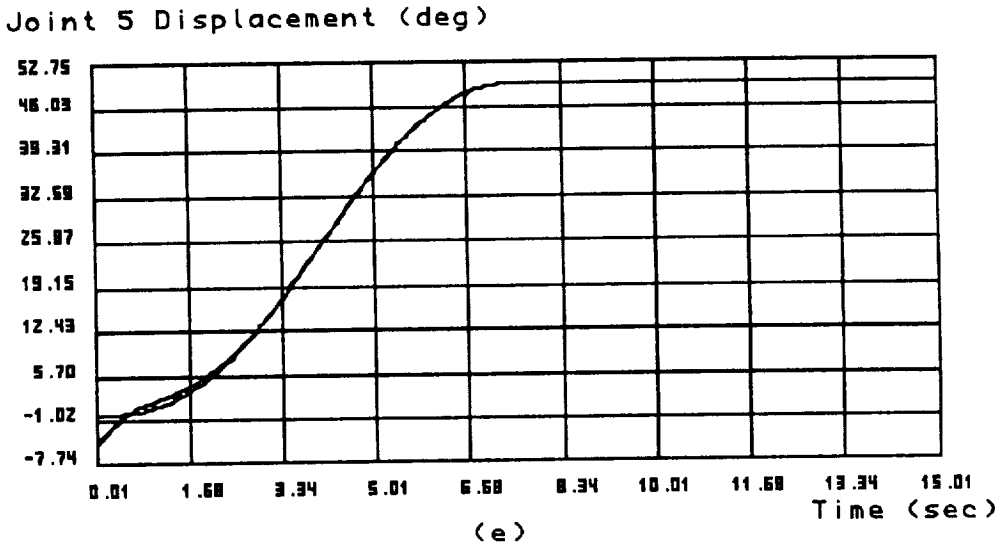


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



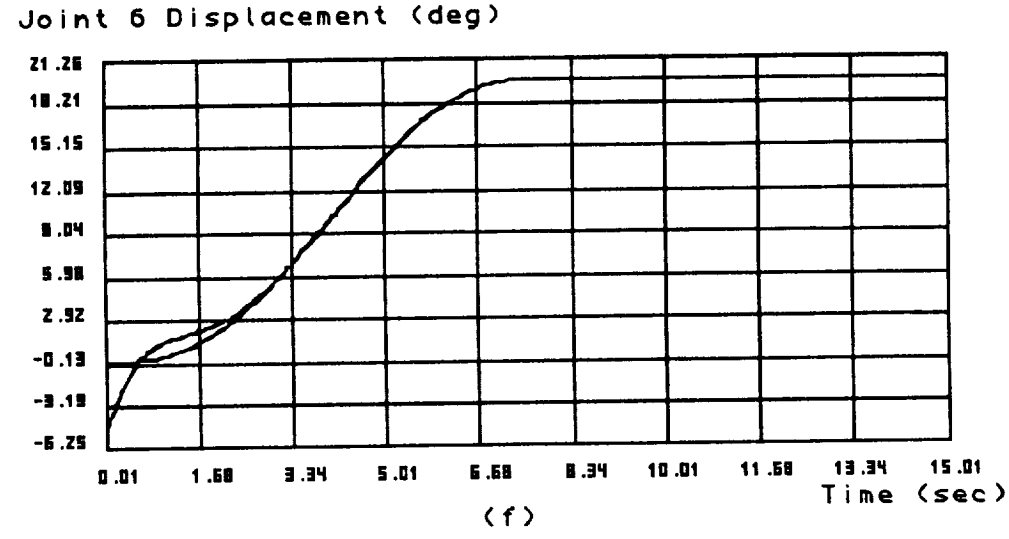
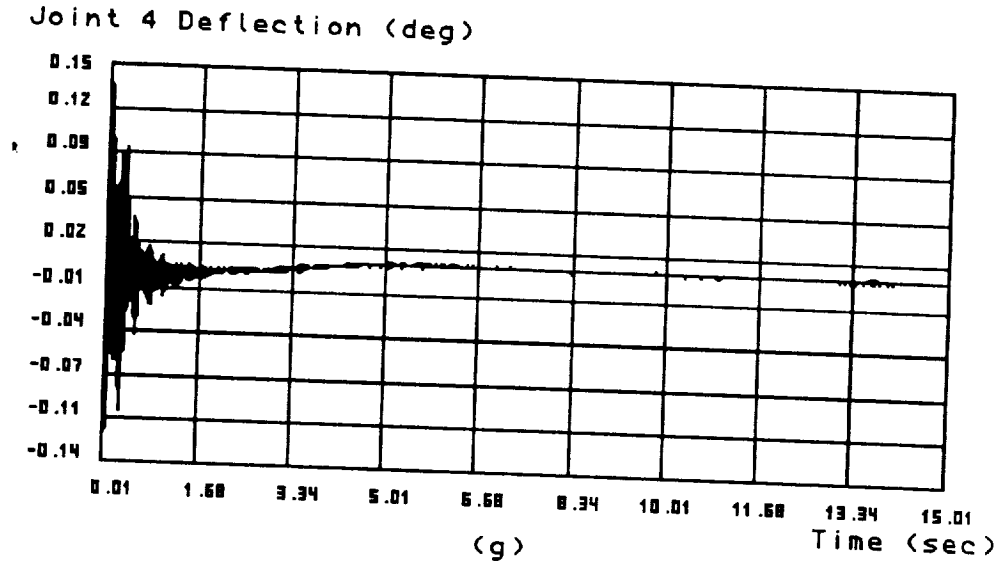


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



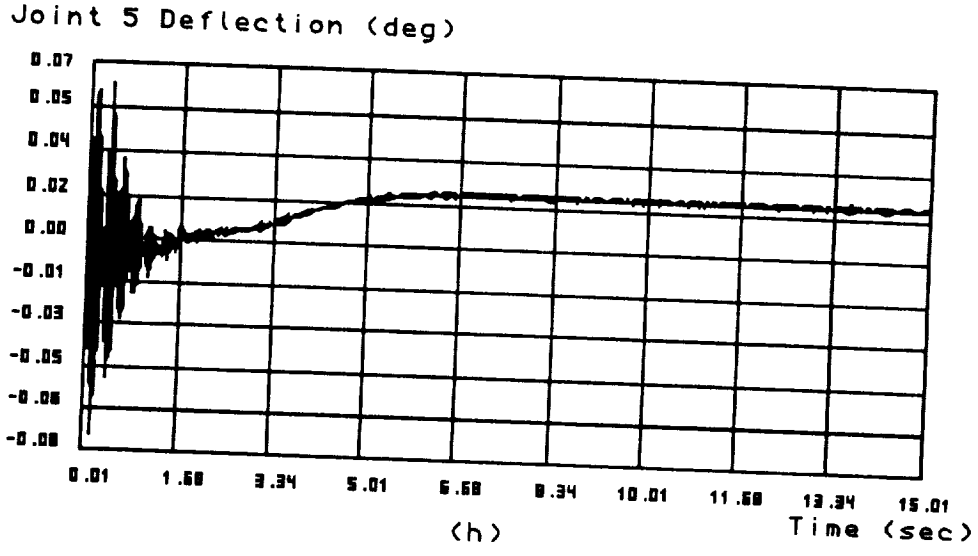
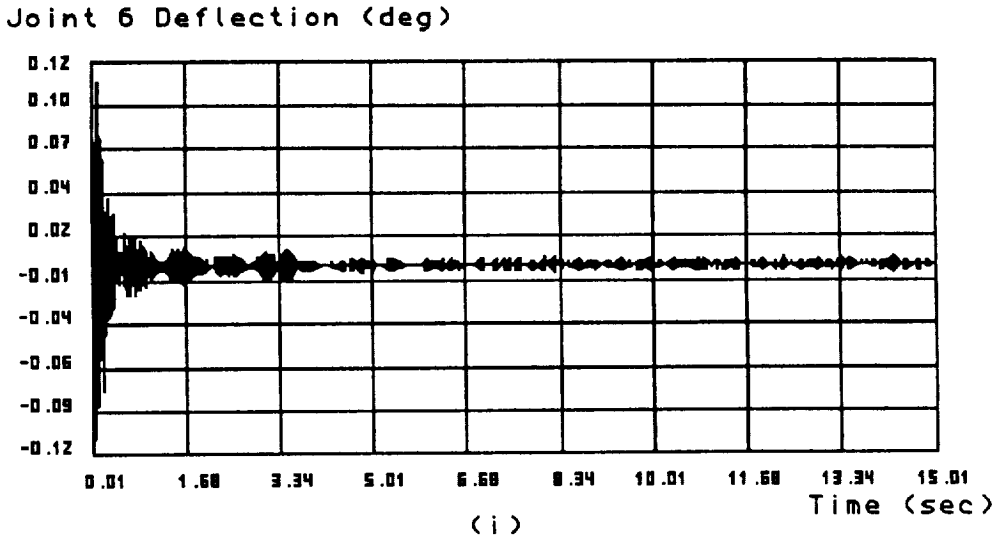


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



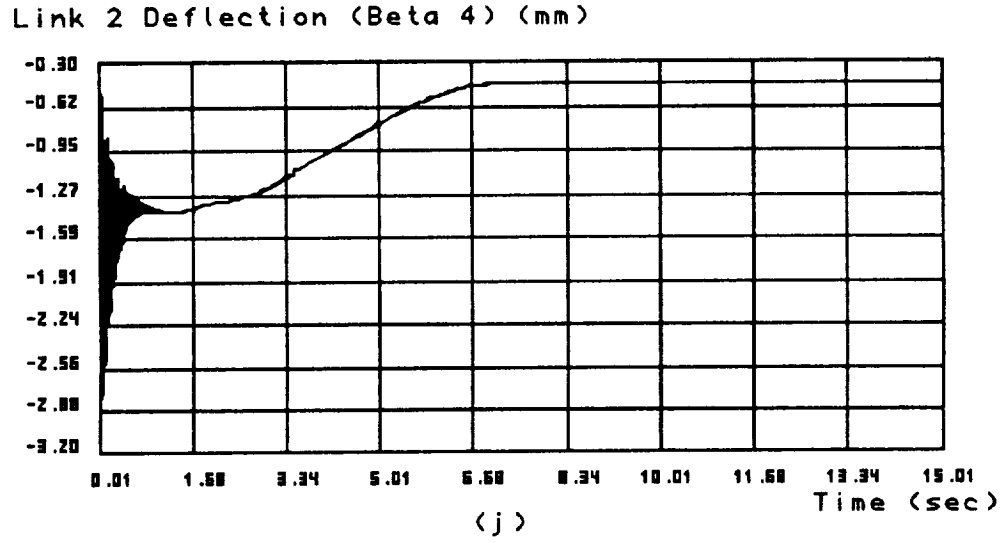
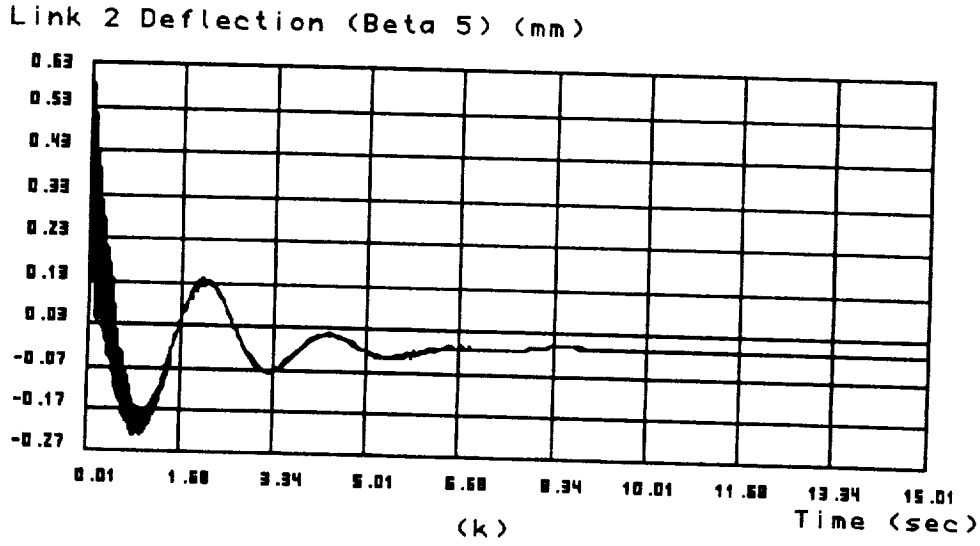


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



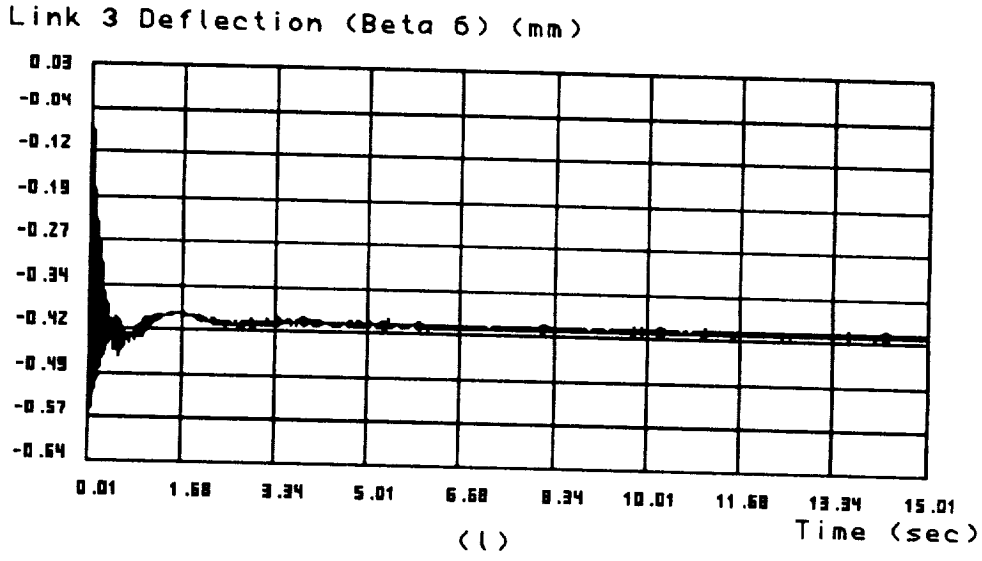
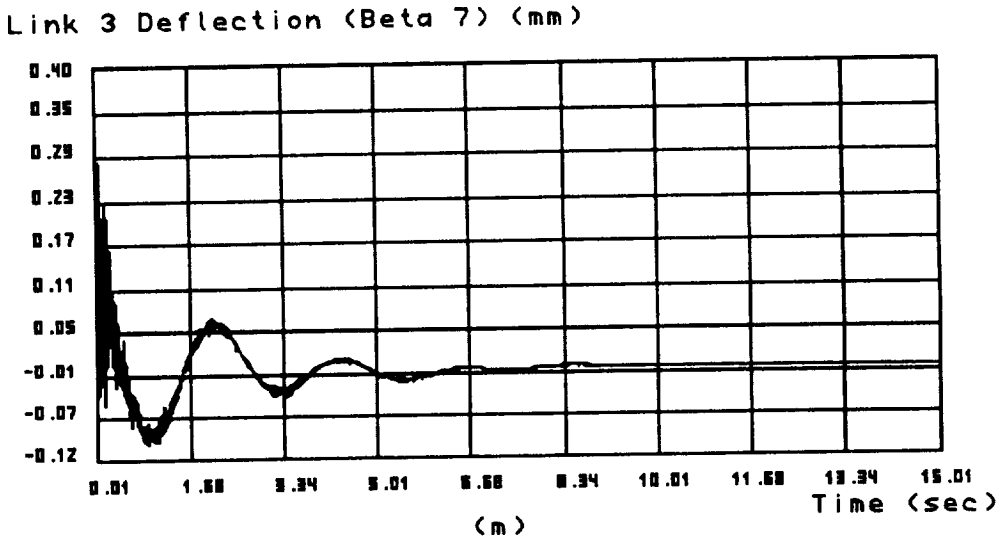


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



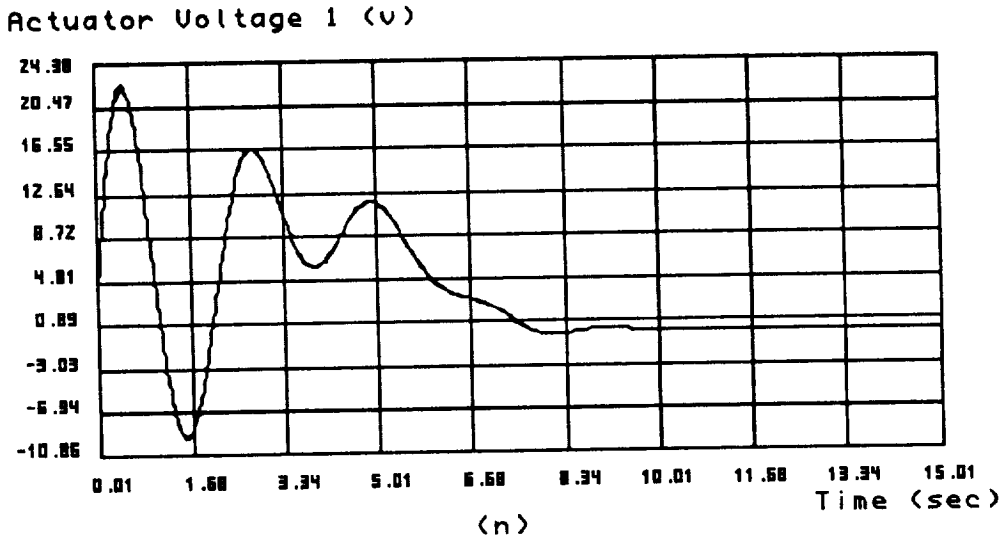
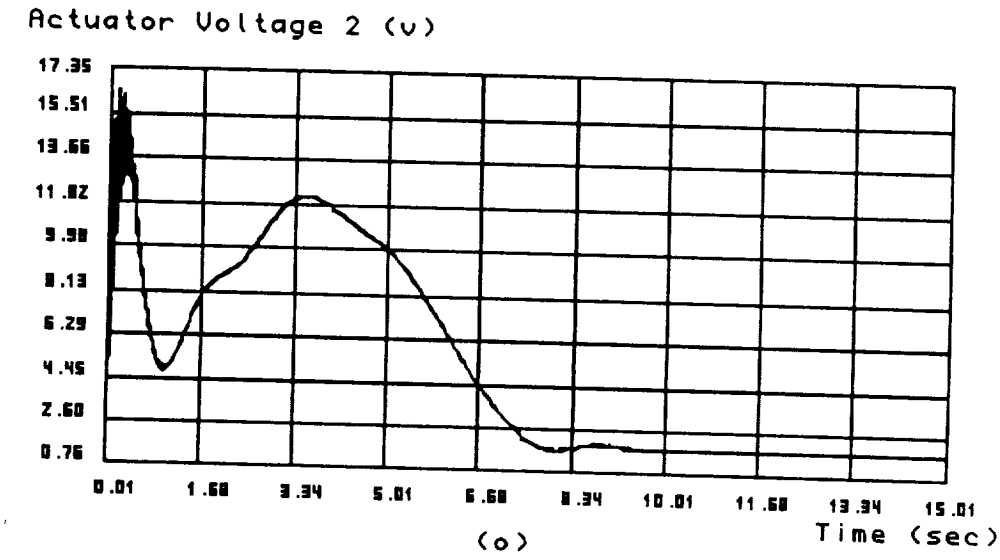


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



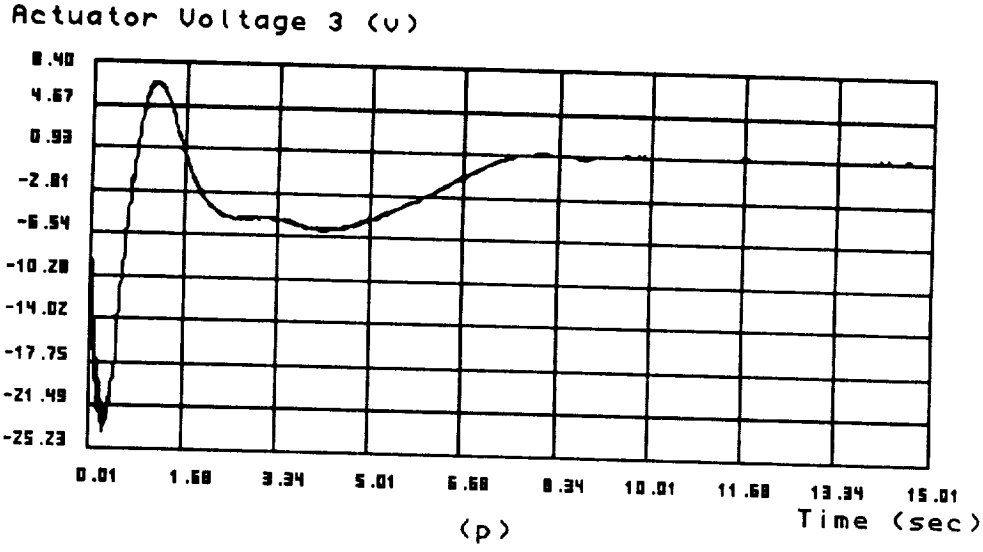
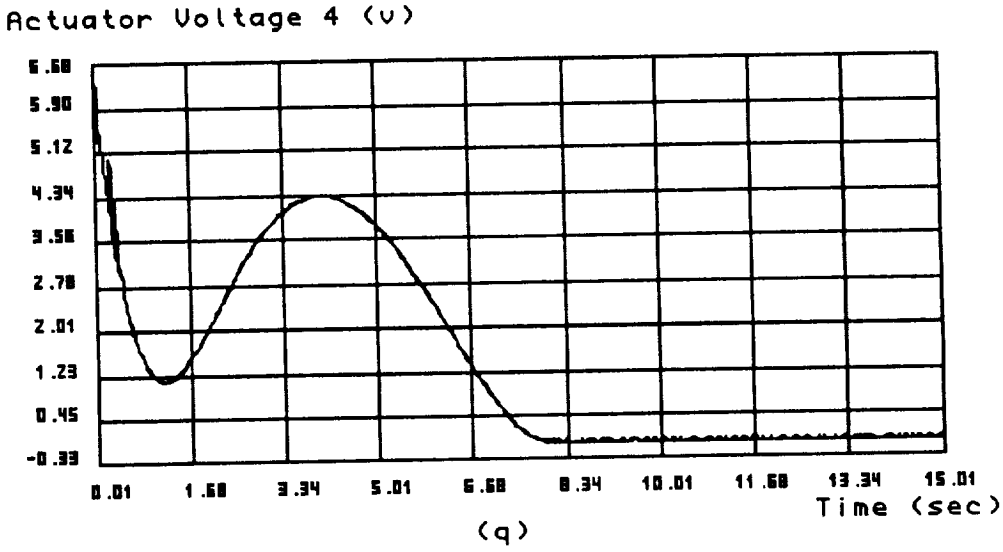


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



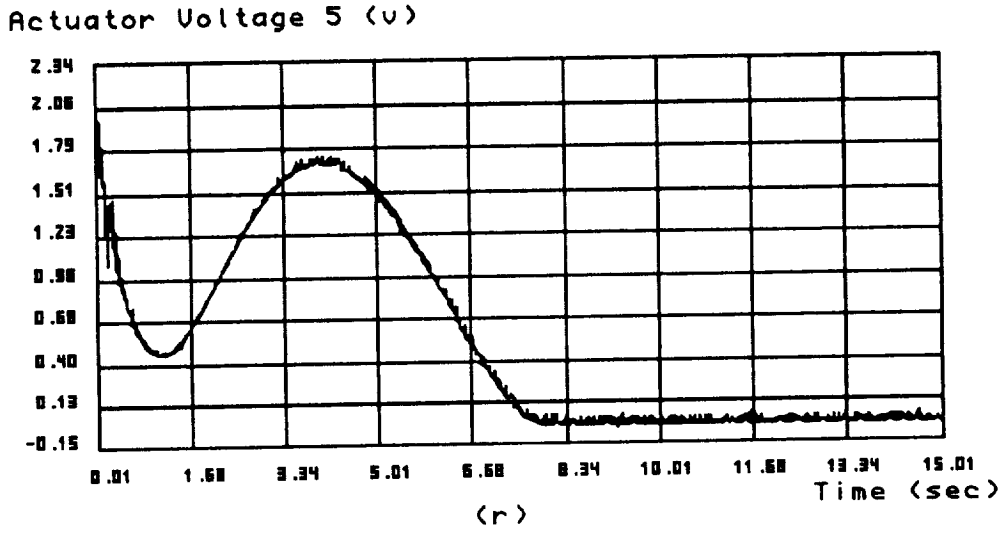


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)

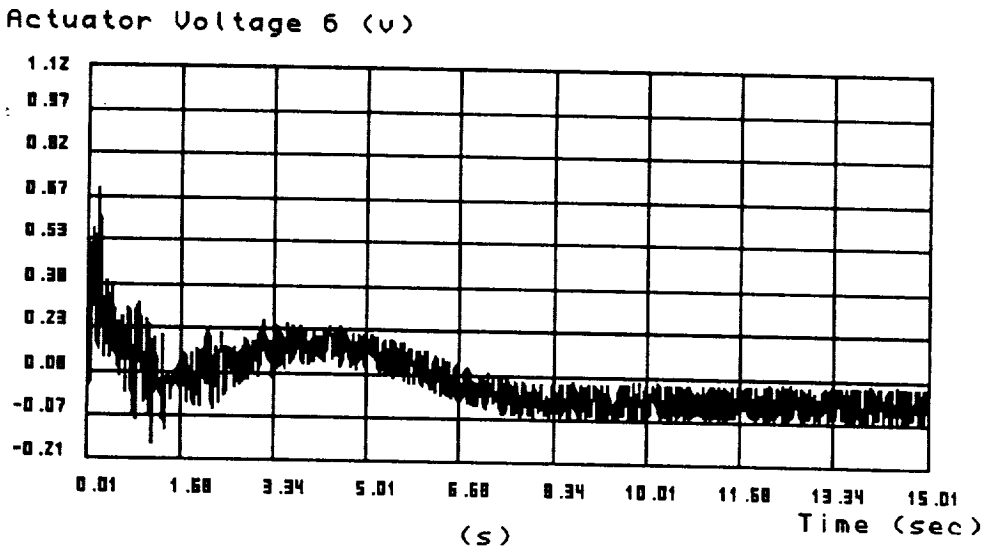
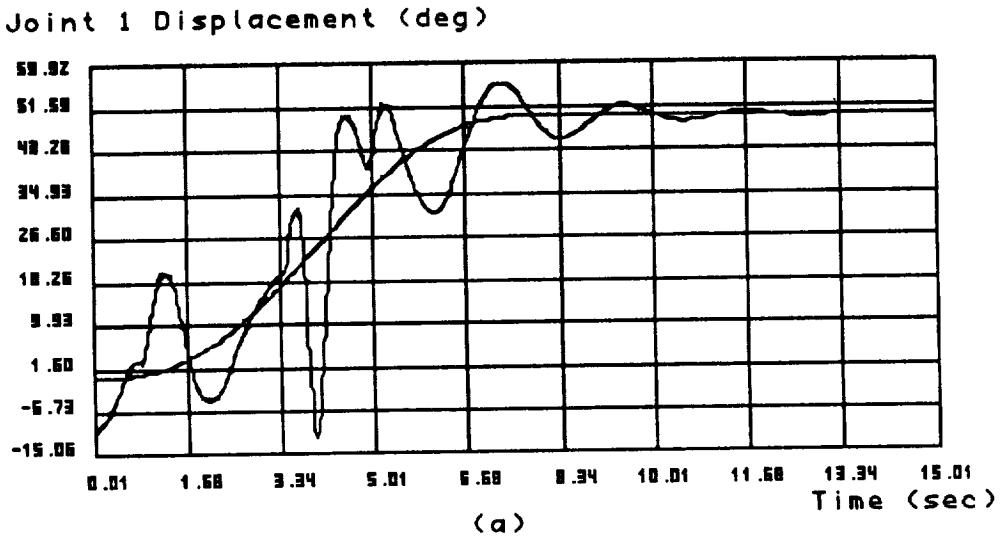


Figure 5.12: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Appropriate P Matrix)



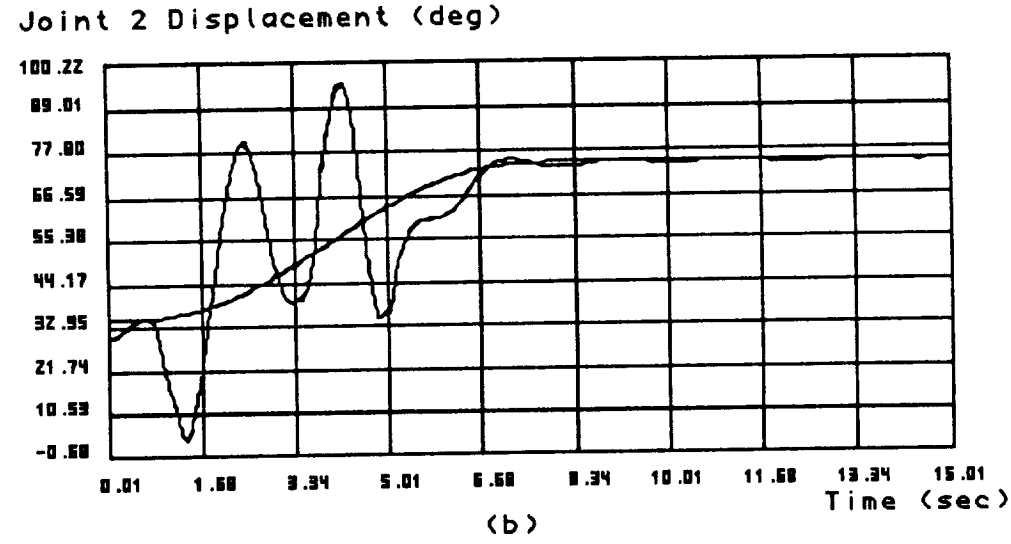
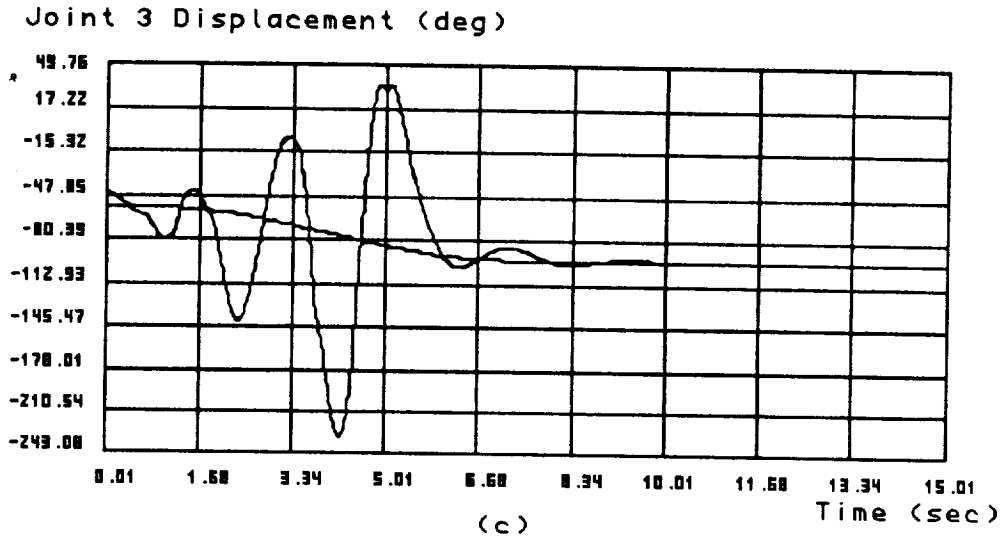


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



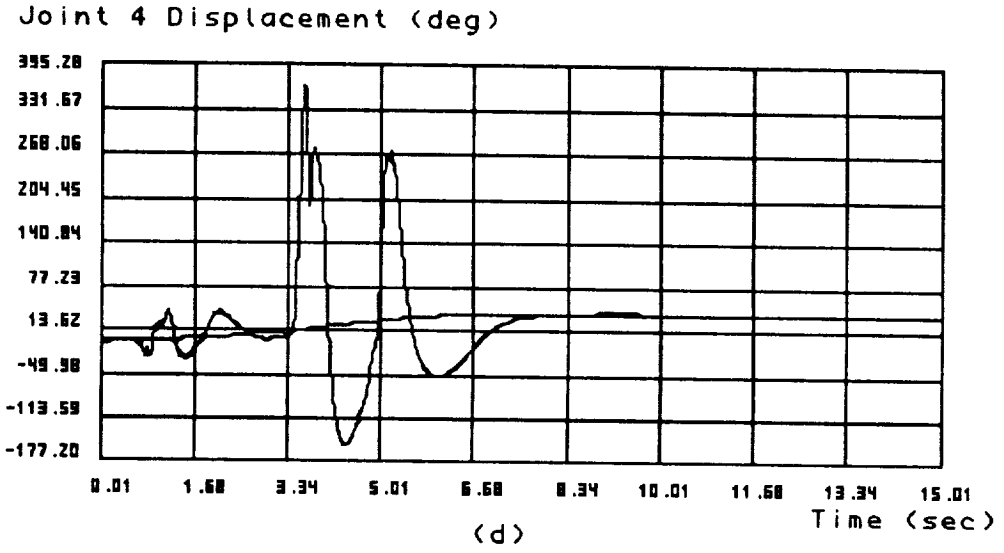
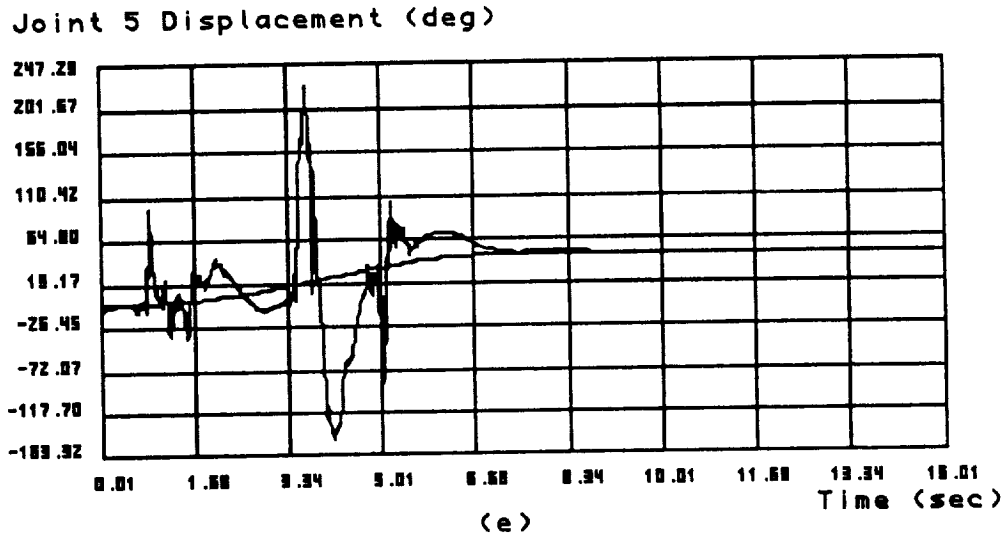


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



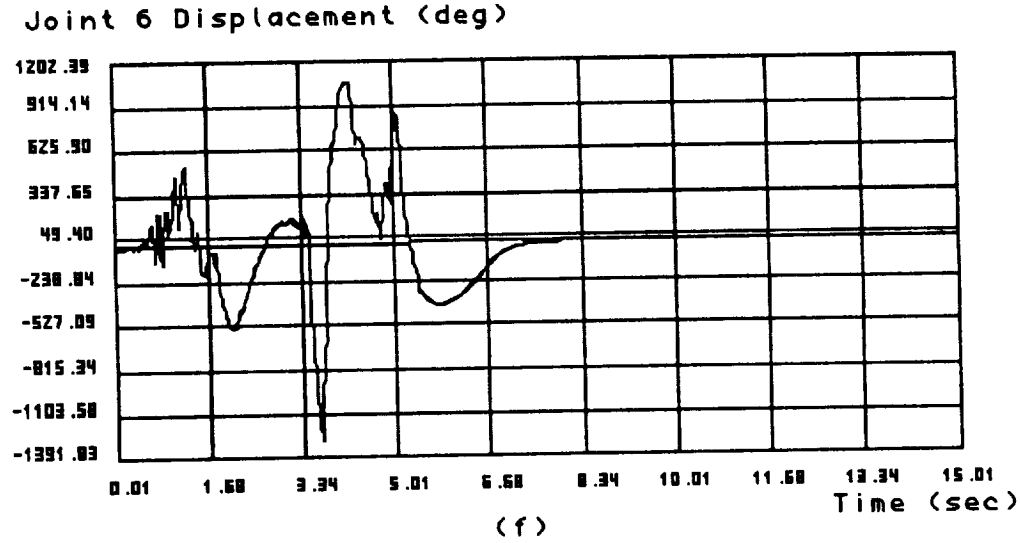
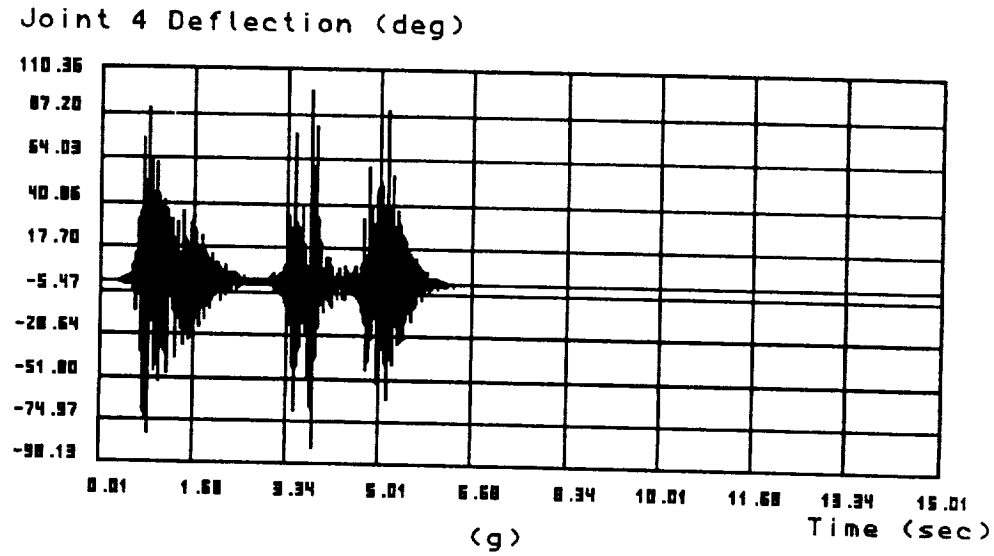


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



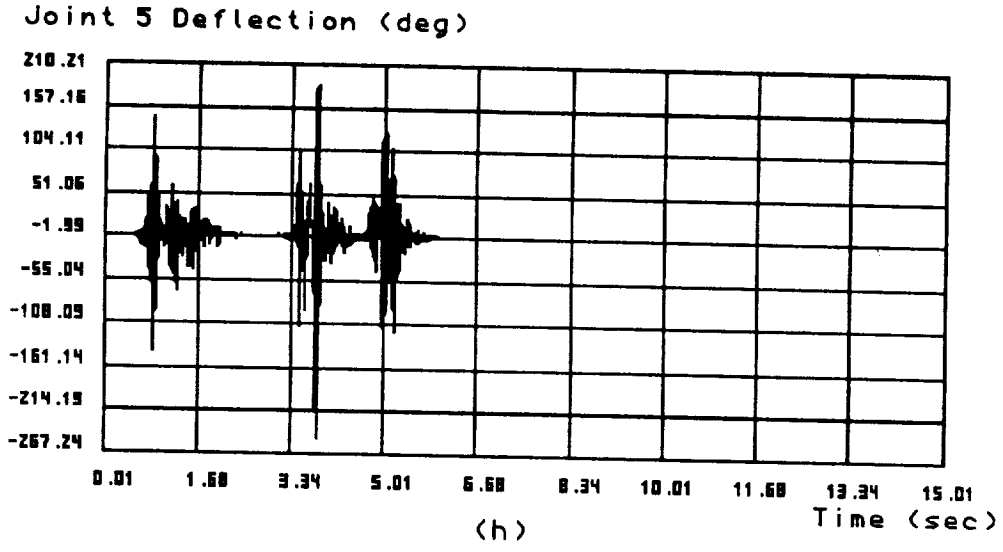
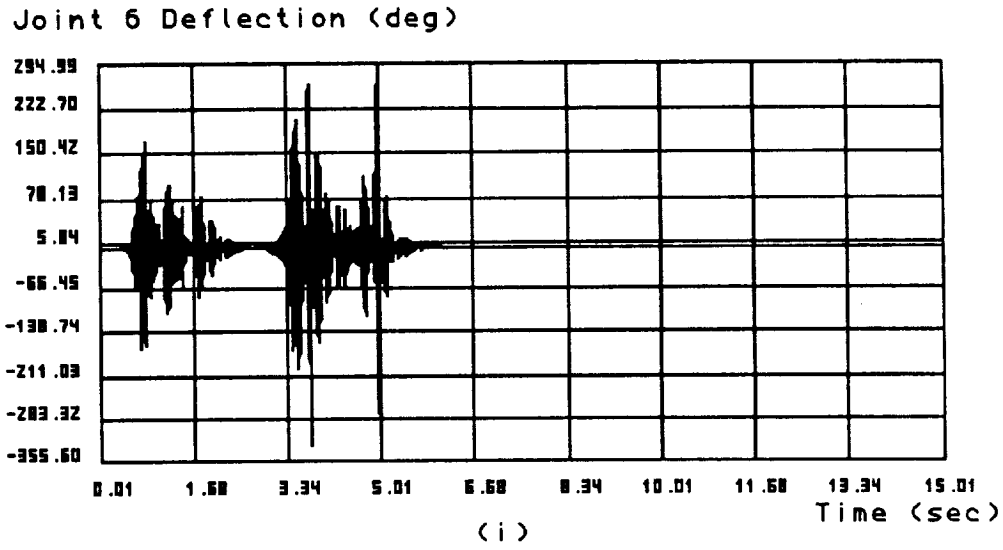


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



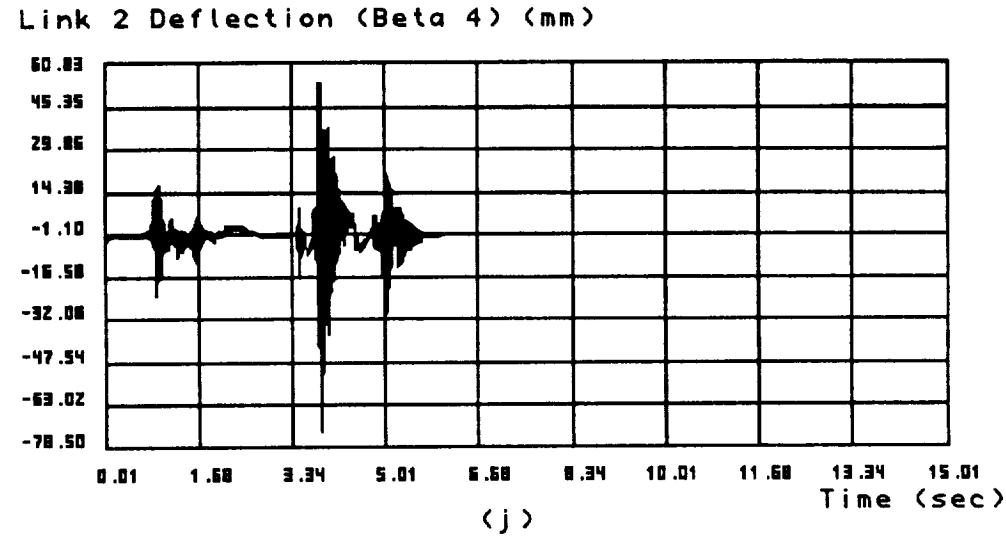
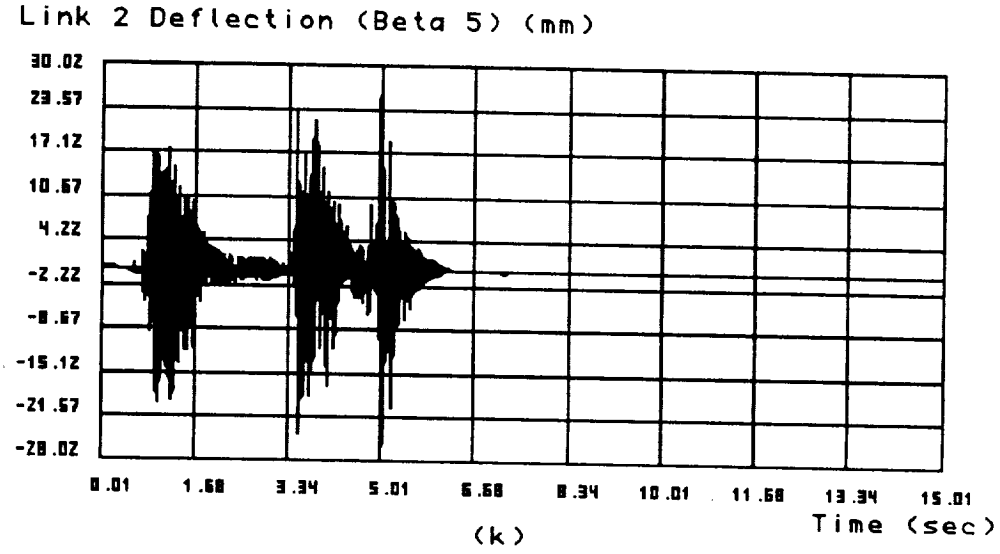


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



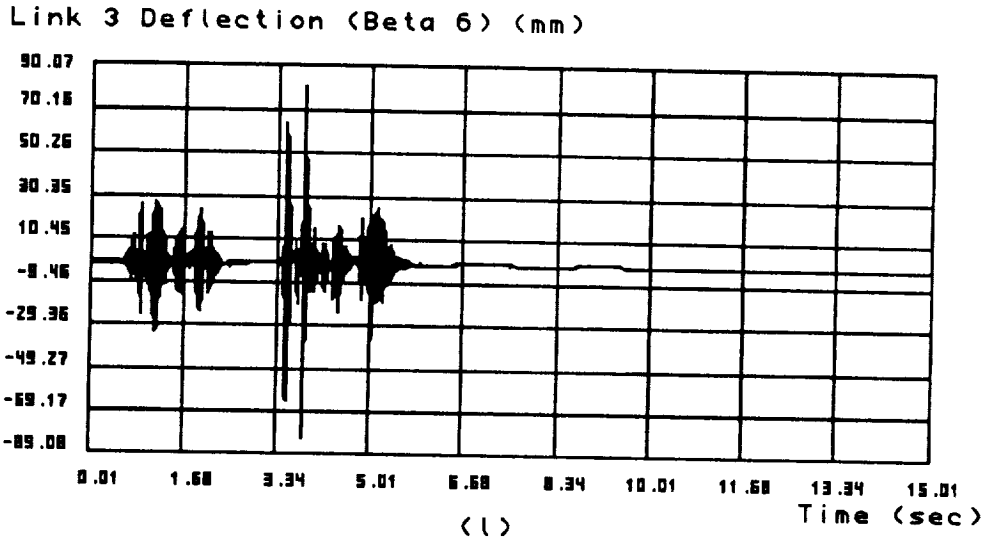
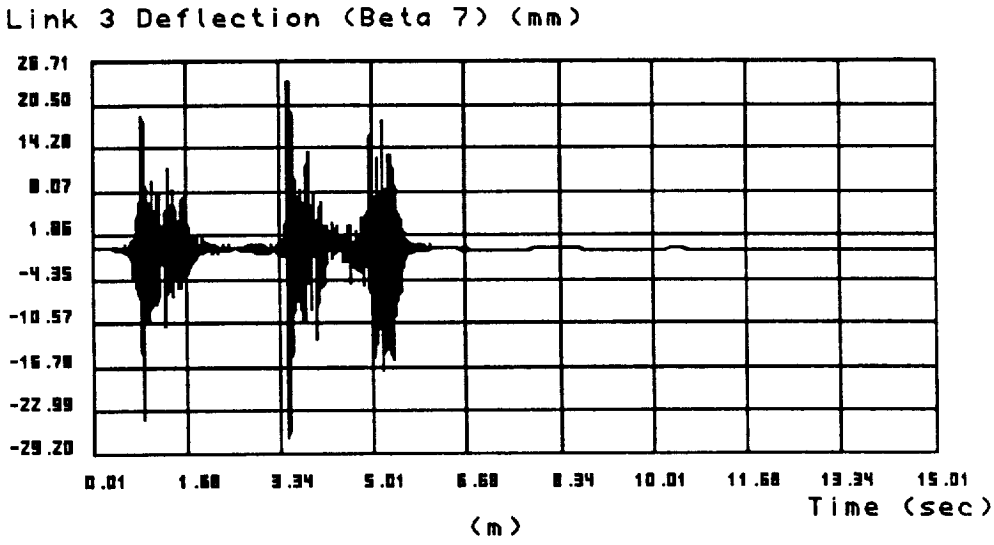


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



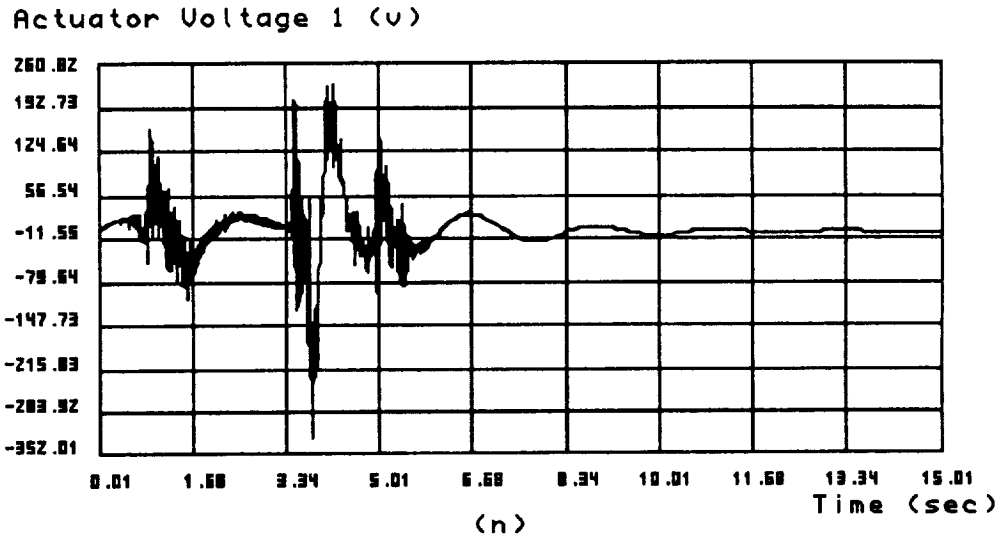
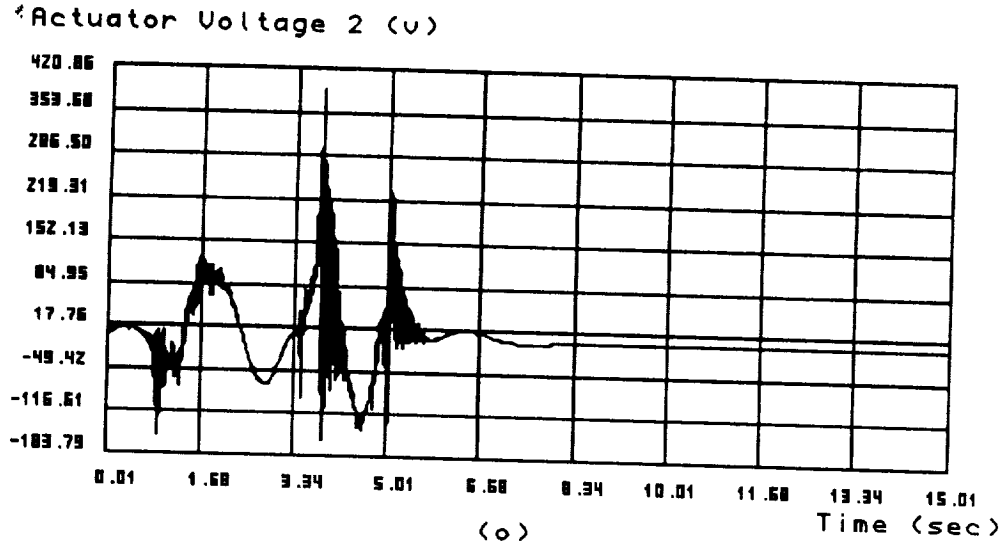


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



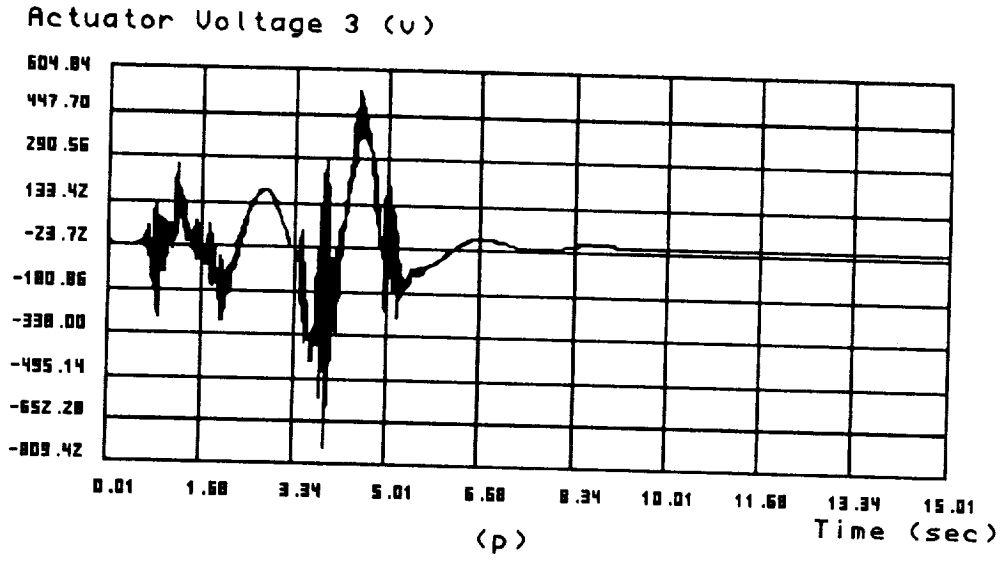
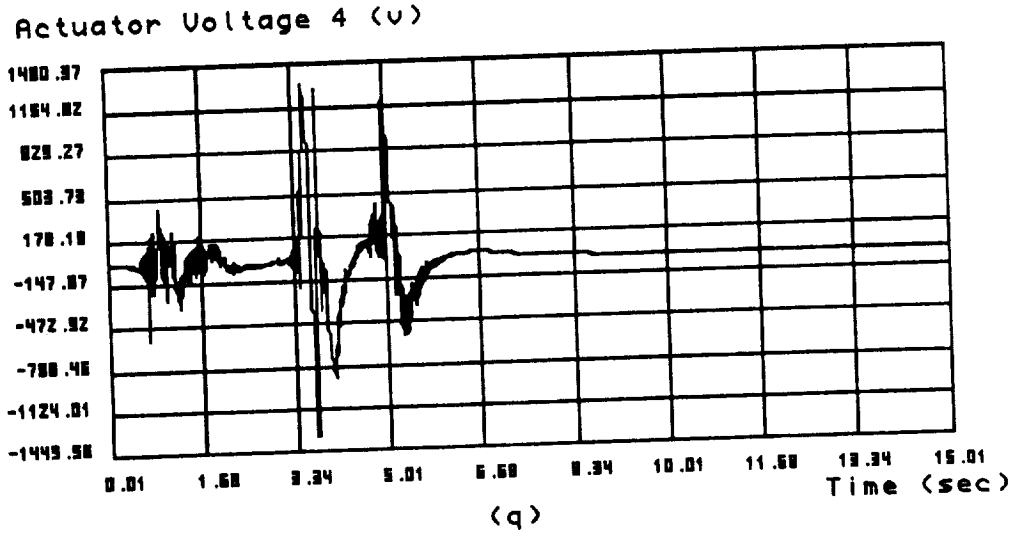


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)



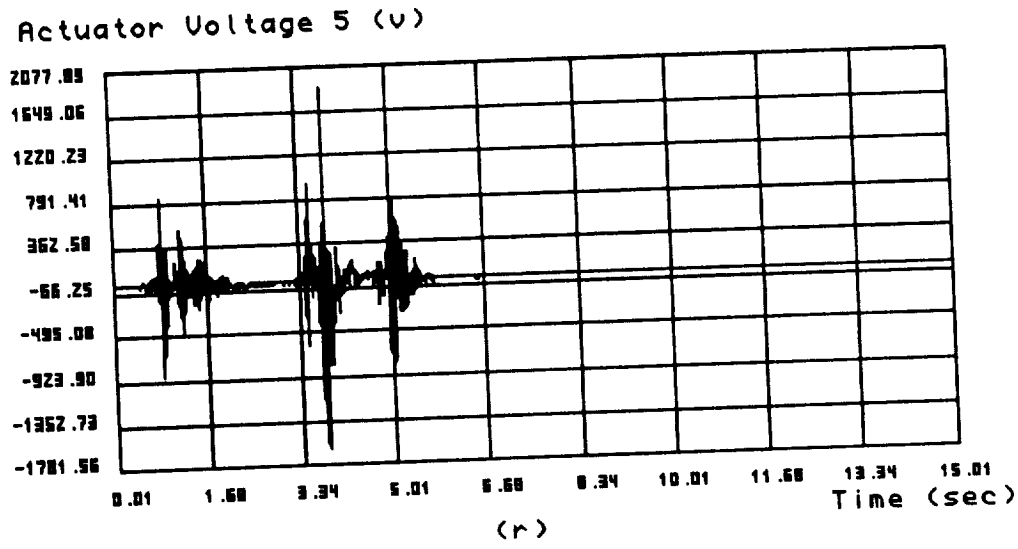


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)

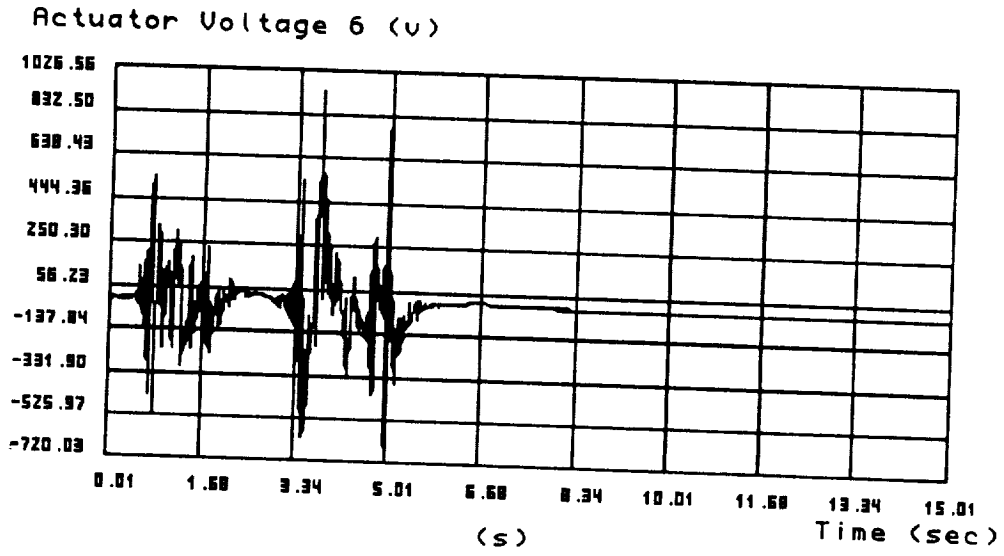


Figure 5.13: Simulation Results of the Six-Link Robot Modeled with Wrist, Upper Arm, and Forearm Compliances (Improper P Matrix)

5.10 Summary

In this chapter, various controllers are presented to control compliant manipulators whose system parameters are assumed to be known a priori. First, the difficulty of ideal acceleration assignment is analyzed, which could be considered as solving an overdetermined problem. Due to the dimensional mismatch between the available number of actuators and the modeled degrees of freedom, solution of a given overdetermined problem is not ensured. Therefore, Lyapunov's stability criterion is employed to assist controller design. Since Lyapunov stability is defined in $\mathcal{R}^1$ space, control design eventually becomes a redundant problem where multiple selections for control structure exist. Then, the orthogonal projection method is used to construct the first proposed control algorithm. However, this method is applicable only to systems with $n_\theta \geq n_\beta$. Therefore, modified algorithms are presented which demand less computational effort than the orthogonal projection method. In the designed controllers, the effects of the P matrix on system stability and system performance are discussed. To obtain a desirable P structure, an explicit Lyapunov matrix solution is presented, which allows direct adjustment of the P matrix structure through regulation of Q matrix values. Three case studies of the modified control algorithms are conducted on a compliant six-link robot. The first case models the robot with six joint compliance. In this case, the controller is further simplified after exploiting the inverse identity of the generalized inertia matrix. Voltage saturation effect is also tested in the first case. The second example uses lumped parameters to model the first three joint compliances and the forearm and upper arm flexibilities. Controller robustness is examined by a large payload uncertainty. Finally, generality of the proposed controller is demonstrated through the third ex-

ample which is a six-link robot with a compliant wrist and two flexible links. Seven lumped springs are used to model structural flexibility. The influence of P matrix values on controller performance is also investigated in the third example.

Chapter 6

Adaptive Control of Compliant Robotic Manipulators

In the last chapter, the controllers for compliant manipulators are constructed based on well-known system and payload parameters. The robustness of these controllers are illustrated in computer simulations through a payload uncertainty which represents a discrepancy between actual system and mathematic model. The simulation results show a reliable stability on motion control under the presence of payload uncertainty. However, controller performance degrades as the uncertainty becomes considerably large, as shown in the second example of the last chapter where payload error is twice of the assumed value. Generally, for routine operations, the payload range could be estimated from a given task. Therefore, a conservative approach could be to choose the average payload value as the working object of controller. Yet, a controller designed in this manner becomes inefficient when payload varies significantly. Therefore, when the information on working objects is not known precisely or if it varies significantly, such as in mining or undersea exploration applications, it will be desirable and very effective to have controller adaptively adjust the commands to meet the uncertainty and additionally provide information about the payload. Such adaptive controllers not only provide robots with a sense of intelligence but also assist human operators to identify the working object. Being motivated by this practical as well as challenging objective, an adaptive control law for compli-

ant manipulators is developed and tested in this chapter.

Presentation of the adaptive controller will proceed in six sections. The first section shows two methods of expressing manipulator dynamics in a linear form in structural and payload inertial parameters which include mass, center of mass location, and moment of inertia of each constituent link and payload. By considering these inertial parameters as to be determined, the second section introduces, for this linear system, a continuous-time standard least-squared estimation technique modified with exponential forgetting factor. From the first two section results, an adaptive controller algorithm achieving both motion control and on-line uncertain parameter estimation is developed in the third section. The fourth section will test the proposed controller through case studies. In the fifth section, update delay effect on the adaptive controller performance is analyzed. Suggestions are given to reduce the impact of update delay on the controlled system response. The final section will make comparisons between adaptive and non-adaptive control algorithms by example studies.

6.1 Dynamic Formulation of Explicit Linear System Parameters

It is important to have a precise dynamic description to build an efficient controller. In robotic manipulators, dynamic equations are formulated in terms of the generalized coordinates and inertial properties. The former could be measured on-line by attached transducers such as tachometers, potentiometers, and strain gauges; the latter (inertial parameters) are composed mainly of the link length, mass, location of center of mass, and moment of inertia of each link including the payload. These inertial parame-

ters are generally constant for a given robotic manipulator, and manipulator inertial magnitudes are obtainable from the robot manufacturer. Yet, it is desirable to have occasional on-site calibration especially when manipulator is reassembled due to maintenance, transportation, or modularized structure. Among these inertial properties, only the link lengths can be measured immediately from the manipulator. The exact link mass, center of mass location, and moment of inertia of a completely assembled manipulator are difficult to calibrate directly, especially after adding the inertia of driving systems. Nevertheless, it will be shown later that manipulator dynamics could be expressed as a linear equation of these not directly measurable inertial quantities so that an on-line calibration is possible experimentally.

In the following sections, robotic dynamic equations will be derived by both the Newton-Euler and Lagrange methods. Unlike the dynamic equations introduced in the second and third chapters, the final dynamic expressions are linear in mass, center of mass location, and moment of inertia of each modeled link. Since manipulator payload is generally constant and could be modeled as a fixed link attached to manipulator gripper, the following derivations are general for manipulators with or without payload. Therefore, the results could be applied to manipulators in off-line inertial calibration or on-line motion control and payload estimation. Structural compliances are included in the following analysis, and lumped parameter approximation is employed to model manipulator flexibility.

6.1.1 The Newton-Euler Method

The Newton-Euler method of deriving manipulator dynamics is to find the force and torques acting on each link first, then to project them to

the upstream, closer to the base, driving joints. The sum of projected forces or torques at each joint give the corresponding actuating force of a prismatic actuator or actuating torque of a revolute joint. Without externally applied force and torques, the force and torque on each link considered here are due to inertial load. It will be shown that these inertial force and torques are linear in mass, center of mass location, and moment of inertia of each link. Before giving the derivation, it should be noted that the center of mass location and moment of inertia of a moving link are constants only with respect to a fixed coordinate frame attached to the link. So, in order to distinguish a local coordinate description from a global one, a superscript, e.g. l, is added to each spatial vector defined in a local coordinate frame fixed in link l.

Figure 6.1 shows a floating link, l, with a mass m, center of mass location r^l, and moment of inertia I_c^l, whose components are

$$r^l = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \in \mathcal{R}^3 ; \; I_c^l = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \in \mathcal{R}^{3\times3} \tag{6.1}$$

and due to the symmetry in I_c^l, $I_{xy} = I_{yx}$, $I_{xz} = I_{zx}$, and $I_{yz} = I_{zy}$. The superscript l indicates that these parameters are defined by the lth frame and hence constants. It will be shown that the final inertial dynamics are a linear function of ten inertial parameters composed by

$$m,\, r_x,\, r_y,\, r_z,\, I_{xx},\, I_{xy},\, I_{xz},\, I_{yy},\, I_{yz},\, I_{zz}$$

If we define $P^l \in \mathcal{R}^3$ as the distance vector between the origins of frame l and global frame, and let $R^l \in \mathcal{R}^3$ be the location of the center of mass measured from the global origin, where both vectors are expressed in the frame l coordinates, then

$$R^l = P^l + r^l$$

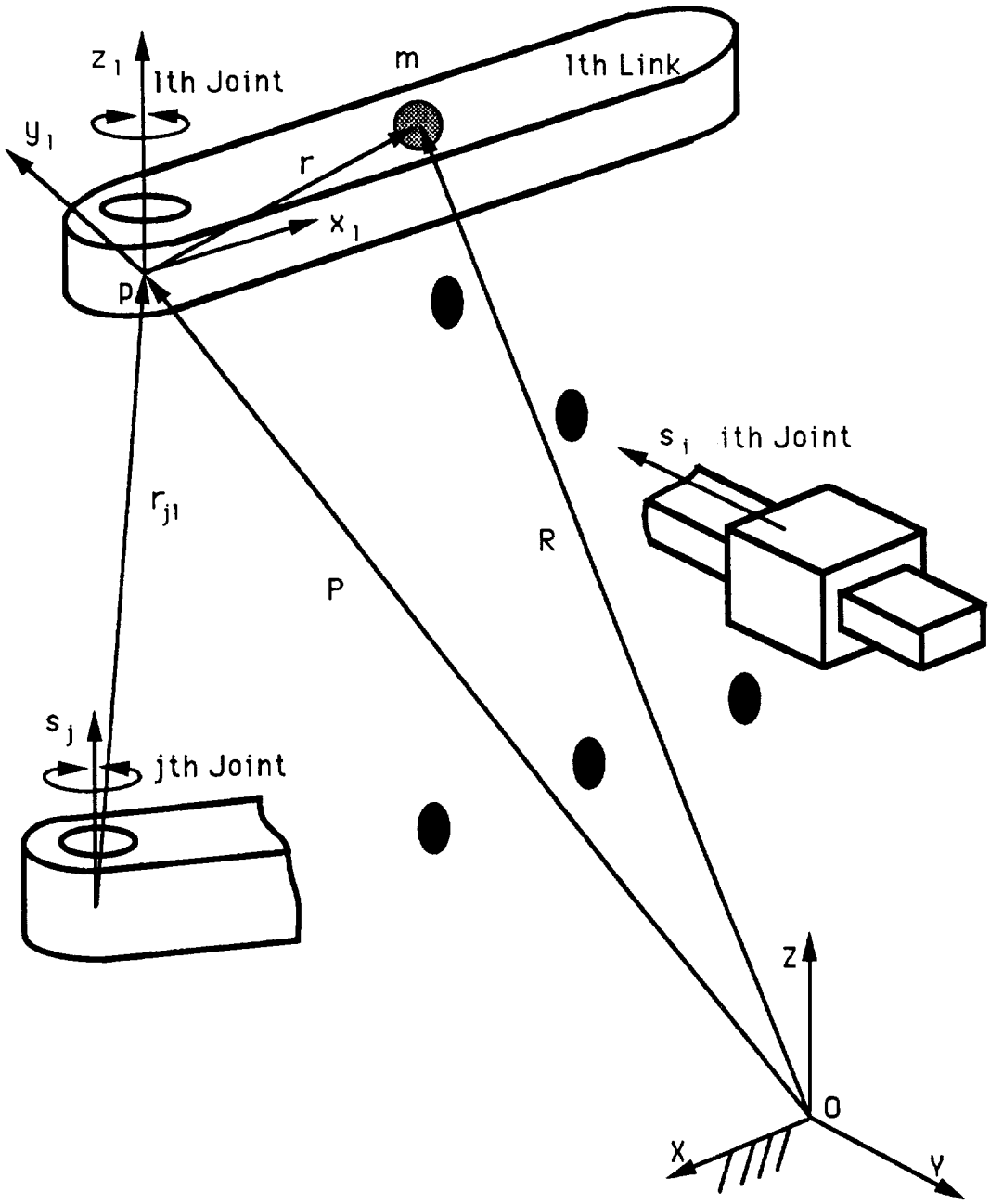


Figure 6.1 A Simple Floating Link Model

$$\begin{aligned} \dot{R}^l &= \dot{P}^l + \omega^l \times r^l \\ \ddot{R}^l &= \ddot{P}^l + \dot{\omega}^l \times r^l + \omega^l \times (\omega^l \times r^l) \end{aligned} \tag{6.2}$$

where $\omega^l = [\omega_x^l\, \omega_y^l\, \omega_z^l]^T \in \mathcal{R}^3$ is the angular velocity of link l expressed in frame l. In Figure 6.1, the origin of l coordinate frame, point p, is generally located along the axis of driving joint of link l or at the center of gripper, if link l is a payload. Therefore, once the inertial force and torque are defined at p, they can be transferred to the upstream links recursively.

The inertial force, $F^l \in \mathcal{R}^3$, at the center of mass of link l is

$$\begin{aligned} F^l &= m\ddot{R}^l + m\hat{g}^l \\ &= m(\ddot{P}^l + \hat{g}^l) + \dot{\omega}^l \times mr^l + \omega^l \times (\omega^l \times mr^l) \end{aligned} \tag{6.3}$$

where $\hat{g}^l$ is the vector of gravitational acceleration defined in l frame, and in the global frame expression, $\hat{g} = [0\ 0\ g]^T \in \mathcal{R}^3$ where g is the gravitational acceleration constant. The inertial torque, $\tau^l \in \mathcal{R}^3$, at the same point is

$$\tau^l = I_c^l \dot{\omega}^l + \omega^l \times (I_c^l \omega^l) \tag{6.4}$$

A shift of both vectors to the origin p of l frame produces a force F_p^l with $F_p^l = F^l$ and a torque τ_p^l defined by

$$\begin{aligned} \tau_p^l &= \tau^l + r^l \times F^l \\ &= I_c^l \dot{\omega}^l + \omega^l \times (I_c^l \omega^l) + r^l \times m(\ddot{P}^l + \hat{g}^l) \\ &\quad + m\{r^l \times [\dot{\omega}^l \times r^l + \omega^l \times (\omega^l \times r^l)]\} \end{aligned} \tag{6.5}$$

The last term in the right-hand side of the above equation is simplified as follows. Given three $\mathcal{R}^3$ vectors a, b, and c, a vector triple product identity is defined as $a \times (b \times c) = (a \cdot c)b - (a \cdot b)c$, hence $r^l \times (\dot{\omega}^l \times r^l)$ becomes

$$\begin{aligned} r^l \times (\dot{\omega}^l \times r^l) &= (r^l \cdot r^l)\dot{\omega}^l - (r^l \cdot \dot{\omega}^l) r^l \\ &= (r^l \cdot r^l \mathcal{I} - r^l r^{l\,T})\dot{\omega}^l \end{aligned} \tag{6.6}$$

where $\mathcal{I} \in \mathcal{R}^{3\times3}$ is a unit matrix, and $r^l r^{l^T}$ is a 3×3 dyad. Also, by using the identity

$$a \times [b \times (b \times a)] = b \times [a \times (b \times a)] = -(a \cdot b)(b \times a)$$

the term $r^l \times [\omega^l \times (\omega^l \times r^l)]$ becomes

$$\begin{aligned} r^l \times [\omega^l \times (\omega^l \times r^l)] &= \omega^l \times [r^l \times (\omega^l \times r^l)] \\ &= \omega^l \times (r^l \cdot r^l \mathcal{I} - r^l r^{l^T})\omega^l \end{aligned} \tag{6.7}$$

After collecting $(\cdot)\dot{\omega}^l$ and $\omega^l \times (\cdot)\omega^l$ terms separately, τ_p^l has a compact form of

$$\tau_p^l = I_p^l \dot{\omega}^l + \omega^l \times I_p^l \omega^l + m r^l \times (\ddot{P}^l + \hat{g}^l) \tag{6.8}$$

with I_p^l defined as

$$\begin{aligned} I_p^l &= I_c^l + m(r^l \cdot r^l \mathcal{I} - r^l r^{l^T}) \\ &= \begin{bmatrix} I_{xx} + m(r_y^2 + r_z^2) & I_{xy} - m r_x r_y & I_{xz} - m r_x r_z \\ I_{yx} - m r_y r_x & I_{yy} + m(r_x^2 + r_z^2) & I_{yz} - m r_y r_z \\ I_{zx} - m r_z r_x & I_{zy} - m r_z r_y & I_{zz} + m(r_x^2 + r_y^2) \end{bmatrix} \\ &= \begin{bmatrix} I_{pxx} & I_{pxy} & I_{pxz} \\ I_{pyx} & I_{pyy} & I_{pyz} \\ I_{pzx} & I_{pzy} & I_{pzz} \end{bmatrix} \end{aligned} \tag{6.9}$$

Eventually, the elements of I_p^l will appear in the final dynamic equations. Since I_p^l is a constant matrix, I_c^l could be obtained once I_p^l, r^l, and m are identified. From both force and torque expressions, Equations 6.3 and 6.8. it will be shown that the final dynamic equations are linear in *ten* inertial parameters

$$m,\ m r_x,\ m r_y,\ m r_z,\ I_{pxx},\ I_{pxy},\ I_{pxz},\ I_{pyy},\ I_{pyz},\ I_{pzz}$$

First, these ten parameters must be organized from the matrix multiplication and vector cross product terms in Equations 6.3 and 6.8. In order to extract

these inertial parameter, two simple notations are introduced here. The first notation is defined as

$$[\omega^l \cdot] \stackrel{\text{def}}{=} \begin{bmatrix} \omega_x^l & \omega_y^l & \omega_z^l & 0 & 0 & 0 \\ 0 & \omega_x^l & 0 & \omega_y^l & \omega_z^l & 0 \\ 0 & 0 & \omega_x^l & 0 & \omega_y^l & \omega_z^l \end{bmatrix} \tag{6.10}$$

and for the cross product of any two $\mathcal{R}^3$ vectors, $A = [a_1\, a_2\, a_3]^T$ and $B = [b_1\, b_2\, b_3]^T$, the second notation is defined as

$$\begin{aligned} A \times B &= \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \times \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix} \\ &= \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \\ &\stackrel{\text{def}}{=} [A\times]B \end{aligned} \tag{6.11}$$

Then, by the first notation,

$$\begin{aligned} I_p^l \omega^l &= \begin{bmatrix} I_{pxx} & I_{pxy} & I_{pxz} \\ I_{pyx} & I_{pyy} & I_{pyz} \\ I_{pzx} & I_{pzy} & I_{pzz} \end{bmatrix} \begin{bmatrix} \omega_x^l \\ \omega_y^l \\ \omega_z^l \end{bmatrix} \\ &= \begin{bmatrix} I_{pxx}\omega_x^l + I_{pxy}\omega_y^l + I_{pxz}\omega_z^l \\ I_{pyx}\omega_x^l + I_{pyy}\omega_y^l + I_{pyz}\omega_z^l \\ I_{pzx}\omega_x^l + I_{pzy}\omega_y^l + I_{pzz}\omega_z^l \end{bmatrix} \\ &= [\omega^l \cdot] \begin{bmatrix} I_{pxx} \\ I_{pxy} \\ I_{pxz} \\ I_{pyy} \\ I_{pyz} \\ I_{pzz} \end{bmatrix} \end{aligned} \tag{6.12}$$

and according to the second notation, the cross product terms in Equations 6.3 and 6.8 could be expressed by

$$\dot{\omega}^l \times mr^l = [\dot{\omega}^l\times]mr^l \quad ; \quad \omega^l \times (\omega^l \times mr^l) = [\omega^l\times][\omega^l\times]mr^l$$

$$mr^l \times (\ddot{P}^l + \hat{g}^l) = [-(\ddot{P}^l + \hat{g}^l)\times]mr^l \quad ; \quad \omega^l \times I_p^l\omega^l = [\omega^l\times]I_p^l\omega^l$$

With the assistance of these notations, the force and torque at point p due to the inertial force of link l could be expressed as

$$\begin{bmatrix} F_p^l \\ \tau_p^l \end{bmatrix} = \begin{bmatrix} (\ddot{P}^l + \hat{g}^l) & [\dot{\omega}^l\times] + [\omega^l\times][\omega^l\times] & 0 \\ 0 & [-(\ddot{P}^l + \hat{g}^l)\times] & [\omega^l\cdot] + [\omega^l\times][\omega^l\cdot] \end{bmatrix} \mathbf{a}_l$$
$$\stackrel{\text{def}}{=} R_l''\mathbf{a}_l\,;\; R_l'' \in \mathcal{R}^{6\times 10} \tag{6.13}$$

where $\mathbf{a}_l$ is the vector which contains ten constant inertial parameters of link l which are

$$\begin{aligned} \mathbf{a}_l &= [a_1\, a_2\, \cdots\, a_{10}]^T \in \mathcal{R}^{10} \\ &= [m\; mr_x\; mr_y\; mr_z\; I_{pxx}\; I_{pxy}\; I_{pxz}\; I_{pyy}\; I_{pyz}\; I_{pzz}]^T \end{aligned} \tag{6.14}$$

So far, the inertial force and torque, F_p^l and τ_p^l, are l-frame vectors, they should be converted into the global frame values to compute the corresponding driving torque or force at each joint. Letting T_l be the 3×3 transformation matrix converting l frame coordinates to global frame coordinates, then the global coordinates of F_p^l and τ_p^l are simply given by

$$\begin{aligned} \begin{bmatrix} F_p \\ \tau_p \end{bmatrix} &= \begin{bmatrix} T_l & 0 \\ 0 & T_l \end{bmatrix} \begin{bmatrix} F_p^l \\ \tau_p^l \end{bmatrix} \\ &= \begin{bmatrix} T_l & 0 \\ 0 & T_l \end{bmatrix} R_l''\mathbf{a}_l \\ &\stackrel{\text{def}}{=} R_l'\mathbf{a}_l\,;\; R_l' \in \mathcal{R}^{6\times 10} \end{aligned} \tag{6.15}$$

Since the conversion process is a matrix premultiplication, the linear relation with the constant inertial vector $\mathbf{a}_l$ is maintained in the above expression. Equation 6.15 gives the inertial force and torque of the lth link in terms of the global coordinates. The corresponding driving input at each upstream joint is computed next. In Figure 6.1 a prismatic joint, i, is located at the upstream of link l, then the driving force of i to counteract the inertial force

and torque at link l is

$$\begin{aligned} F_{il} &= s_i \cdot F_p \\ &= [s_i^T \, 0 \, 0 \, 0] R_l' \mathbf{a}_l \\ &\stackrel{\text{def}}{=} (R_F)_l \mathbf{a}_l \; ; \; (R_F)_l^T \in \mathcal{R}^{10} \end{aligned} \tag{6.16}$$

where s_i is unit directional vector of joint i. Summation of all inertia force on i gives the ith joint actuator force as

$$\begin{aligned} F_i &= \sum_{l=i}^{N} F_{il} \\ &= \sum_{l=i}^{N} (R_F)_l \mathbf{a}_l \\ &= [0 \cdots 0 \, (R_F)_i \, \ldots \, (R_F)_N] \mathbf{a} \\ &= R_F \mathbf{a} \; ; \; R_F^T \in \mathcal{R}^{10} \end{aligned} \tag{6.17}$$

where N is the number of links including payload and

$$\mathbf{a} = [\mathbf{a}_1^T \, \mathbf{a}_2^T \, \cdots \, \mathbf{a}_N^T]^T \in \mathcal{R}^{10N}$$

Notice that since all links located at the upstream of joint i contribute no inertial effect on i, they are excluded from the above force summation. Similarly, in Figure 6.1, joint l is located at r_{jl} above a revolute joint j, the torque on joint j due to the inertia of link l is

$$\begin{aligned} \tau_{jl} &= s_j \cdot (r_{jl} \times F_p) + s_j \cdot \tau_p \\ &= (s_j \times r_{jl}) \cdot F_p + s_j \cdot \tau_p \\ &= [(s_j \times r_{jl})^T \, (s_j^T)] R_l' \mathbf{a}_l \\ &\stackrel{\text{def}}{=} (R_T)_l \mathbf{a}_l \; ; \; (R_T)_l^T \in \mathcal{R}^{10} \end{aligned} \tag{6.18}$$

where $s_j \in \mathcal{R}^3$ is unit directional vector of joint j. Notice that scalar triple product identity $a \cdot (b \times c) = (a \times b) \cdot c$ is used in the above derivation. The

sum of all inertial torque on joint j is

$$\begin{aligned}
\tau_j &= \sum_{l=j}^{N} \tau_{jl} \\
&= \sum_{l=j}^{N} (R_T)_l \mathbf{a}_l \\
&= [0 \cdots 0 \, (R_T)_j \, \ldots \, (R_T)_N]\mathbf{a} \\
&\stackrel{\text{def}}{=} R_T \mathbf{a} \; ; \; R_T^T \in \mathcal{R}^{10}
\end{aligned} \tag{6.19}$$

It should be noticed that the above derivations are applicable to both rigid and lumped-compliant manipulator models. In the lumped complaint manipulator model, the primary link and joint oscillations are depicted by the movements of pseudo prismatic or revolute joints which are driven by lumped springs. By following the lumped model description, joints i and j in Figure 6.1 could represent either actual or pseudo joints. For actual joints, F_i and τ_j are respectively the actuator driving force and torques. For pseudo joints, F_i and τ_i are the stored resilient force and torque in lumped springs located at joint i and j separately, which are $F_i = -K_i\beta_i$ and $\tau_j = -K_j\beta_j$, where K_i and K_j are spring stiffnesses and β_i and β_j are spring deformations. Therefore, as shown in Equations 6.17 and 6.19, the inertial dynamics of a robotic manipulator, rigid or lumped compliant, could be expressed as a linear function of inertial parameters $\mathbf{a}$. Similar dynamic expressions are employed by [Sklar, Hudgens, and Tesar, 1990] to perform calibration of rigid robot inertial parameters.

6.1.2 The Lagrange Method

The first step in the Lagrange method is to find system kinetic and potential energy expressions in terms of structural inertia. Let KE_l denote the

kinetic energy of link l, then the total kinetic energy of an N-link manipulator, including payload, is

$$KE = \sum_{l=1}^{N} KE_l$$

and letting $(PE_g)_l$ denote the gravitational potential of link l and $(PE_s)_l$ be the potential energy due to structural resilience of l, then the total potential energy is

$$PE = \sum_{l=1}^{N} \{(PE_g)_l + (PE_s)_l\}$$

Typically, ten constant inertial parameters of link l, which are similar to that defined in Equation 6.14, will be grouped from the kinetic and potential energy expressions into a linear form. Since the differentiations in the Lagrange equation are linear operators, the linear relationship with inertial parameters remains intact in the final dynamic equations. Before presenting this derivation, a note about the usage of notation is that although kinetic and potential energy are scalar physical quantities and independent of the selection of coordinate frame, the following vectors are defined in global frame except those added with a superscript l to emphasize l-frame vectors.

According to Figure 6.1,

$$\begin{aligned} 2KE_l &= m(\dot{P} + \dot{r})^2 + \omega^T T_l I_c^l T_l^T \omega \\ &= m\dot{P}^T\dot{P} + 2m\dot{P}^T\dot{r} + m\dot{r}^T\dot{r} + \omega^T T_l I_c^l T_l^T \omega \end{aligned} \tag{6.20}$$

in which T_l is the 3×3 transformation matrix converting l frame into global coordinates, and $\omega \in \mathcal{R}^3$ is the angular velocity of link l. For $\dot{r} = \omega \times r$, the $(m\dot{r}^T\dot{r})$ term in the above equation could be written as a dyad matrix form through the following operations

$$m\dot{r}^T\dot{r} = m\dot{r} \cdot \dot{r}$$

$$
\begin{aligned}
&= m(\omega \times r) \cdot (\omega \times r) \\
&= m\{\omega \cdot [r \times (\omega \times r)]\} \\
&= m\omega \cdot [(r \cdot r)\omega - r(r \cdot w)] \\
&= \omega^T [m(r^T r \mathcal{I} - r r^T)]\omega \\
&= \omega^T T_l [m(r^{l^T} r^l \mathcal{I} - r^l r^{l^T})] T_l^T \omega \qquad (6.21)
\end{aligned}
$$

where r is transformed into r^l by the relation $r = T_l r^l$, also the orthogonal property of coordinate transformation matrix is used here, i.e., $T_l^T T_l = \mathcal{I}$, and $\mathcal{I} \in \mathcal{R}^{3\times 3}$ is a unit matrix. Then the moment of inertia term in the kinetic energy expression could be redefined with respect to the point p in Figure 6.1 instead of the center of mass as

$$
\begin{aligned}
m\dot{r}^T\dot{r} + \omega^T T_l I_c^l T_l^T \omega &= \omega^T T_l [I_c^l + m(r^{l^T} r^l \mathcal{I} - r^l r^{l^T})] T_l^T \omega \\
&= \omega^T T_l I_p^l T_l^T \omega \qquad (6.22)
\end{aligned}
$$

where

$$
I_p^l = I_c^l + m(r^{l^T} r^l \mathcal{I} - r^l r^{l^T})
$$

Because the constant matrix I_p^l is symmetric, it could be decomposed into six matrices

$$
\begin{aligned}
I_p^l &= \begin{bmatrix} I_{pxx} & I_{pxy} & I_{pxz} \\ I_{pyx} & I_{pyy} & I_{pyz} \\ I_{pzx} & I_{pzy} & I_{pzz} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} I_{pxx} + \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} I_{pxy} + \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} I_{pxz} \\
&+ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} I_{pyy} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} I_{pyz} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} I_{pzz} \\
&\stackrel{\text{def}}{=} \sum_{i=5}^{10} I_{pi} a_i \; ; \; I_{pi} \in \mathcal{R}^{3\times 3} \qquad (6.23)
\end{aligned}
$$

where

$$[a_5\ a_6\ a_7\ a_8\ a_9\ a_{10}] = [I_{pxx}\ I_{pxy}\ I_{pxz}\ I_{pyy}\ I_{pyz}\ I_{pzz}]$$

and I_{pi}, $i \in \{5, \cdots, 10\}$, is defined accordingly. By the distributive law of matrix multiplication, the kinetic energy of moment of inertia could be expressed as a linear sum of the above constant a_i elements, or

$$\begin{aligned} \omega^T T_l I_p^l T_l^T \omega &= \sum_{i=5}^{10} (\omega^T T_l I_{pi} T_l^T \omega)\, a_i \\ &= \sum_{i=5}^{10} (\dot{q}^T\ {}_R G_l^T T_l I_{pi} T_l^T\ {}_R G_l \dot{q})\, a_i \\ &\stackrel{\text{def}}{=} \sum_{i=5}^{10} (\dot{q}^T I_i^* \dot{q})\, a_i \end{aligned} \tag{6.24}$$

In the above expressions, angular velocity ω is replaced by the joint speed $\dot{q}$ and rotational influence coefficient ${}_R G_l$ through the relation $\omega = {}_R G_l \dot{q}$, where $q = [\theta^T\ \beta^T]^T \in \mathcal{R}^{n_\theta + n_\beta}$, and θ is an n_θ vector of actual joint displacements and β is an n_β vector of modeled pseudo joint variables. Similar replacements will be used in the following derivations so that the first and second order influence coefficients could be employed to formulate the final dynamic equations. Notice that the above I_i^*, with $i = 5, \cdots, 10$, are symmetric matrices.

Now for the $m\dot{P}^T\dot{r}$ term in the $2KE_l$ expression given by Equation 6.20,

$$\begin{aligned} m\dot{P}^T\dot{r} &= m\dot{P} \cdot (\omega \times r) \\ &= m\dot{P} \cdot [\omega \times (r_x\hat{x} + r_y\hat{y} + r_z\hat{z})] \\ &= -\dot{P} \cdot (\hat{x} \times \omega)(mr_x) - \dot{P} \cdot (\hat{y} \times \omega)(mr_y) \\ &\quad - \dot{P} \cdot (\hat{z} \times \omega)(mr_z) \end{aligned} \tag{6.25}$$

where $\hat{x}$, $\hat{y}$, and $\hat{z}$ are respectively x, y, and z unit directional vectors of l frame coordinates. By the influence coefficient definitions $\omega = {}_R G_l \dot{q}$ and

$\dot{P} = {}_T G_p \dot{q}$, the above equation becomes

$$\begin{aligned} 2m\dot{P}^T\dot{r} &= m\dot{P}^T\dot{r} + (m\dot{P}^T\dot{r})^T \\ &= \sum_{s=x,y,z} \dot{q}^T\{-[\,{}_T G_p \cdot (\hat{s} \times {}_R G_l)] - [\,{}_T G_p \cdot (\hat{s} \times {}_R G_l)]^T\}\dot{q}(mr_s) \\ &\stackrel{\text{def}}{=} \sum_{i=2}^{4} (\dot{q}^T I_i^* \dot{q})\, a_i \end{aligned} \tag{6.26}$$

with $[a_2\ a_3\ a_4] = [mr_x\ mr_y\ mr_z]$, and I_2^*, I_3^*, and I_4^* are symmetric matrices. In the above equation, since ${}_R G_l$ is a $3 \times (n_\theta + n_\beta)$ matrix, the cross product $\hat{s} \times {}_R G_l$ means cross product of $\hat{s}$ with respect to each $\mathcal{R}^3$ column of ${}_R G_l$, which is defined in the followings. Set $\dot{q} = [\dot{q}_1\ \dot{q}_2\ \cdots\ \dot{q}_{n_\theta+n_\beta}]^T$ and ${}_R G_l = [c_1\, c_2 \cdots c_{n_\theta+n_\beta}]$, where $c_i \in \mathcal{R}^3$, $i \in \{1, \cdots, n_\theta + n_\beta\}$, is column vector of ${}_R G_l$, then

$$ {}_R G_l \dot{q} = \sum_{i=1}^{n_\theta+n_\beta} c_i \dot{q}_i \tag{6.27}$$

and

$$\begin{aligned} \hat{s} \times (\,{}_R G_l \dot{q}) &= \sum_{i=1}^{n_\theta+n_\beta} (\hat{s} \times c_i)\dot{q}_i \\ &= [(\hat{s} \times c_1)\,(\hat{s} \times c_2) \cdots (\hat{s} \times c_{n_\theta+n_\beta})]\,\dot{q} \\ &\stackrel{\text{def}}{=} \hat{s} \times {}_R G_l \dot{q} \end{aligned} \tag{6.28}$$

Finally, by defining $a_1 = m$, the linear term of a_1 in $2KE_l$, Equation 6.20, is simply

$$\begin{aligned} m\dot{P}^T\dot{P} &= m(\dot{q}^T\, {}_T G_p^T\, {}_T G_p \dot{q}) \\ &\stackrel{\text{def}}{=} (\dot{q} I_1^* \dot{q})\, a_1 \end{aligned} \tag{6.29}$$

and $I_1^* = (I_1^*)^T$. A collection of the above expressions gives

$$\begin{aligned} 2KE_l &= \sum_{i=1}^{10} (\dot{q}^T I_i^* \dot{q})\, a_i \\ &= [(\dot{q}^T I_1^* \dot{q})\,(\dot{q}^T I_2^* \dot{q}) \cdots (\dot{q}^T I_{10}^* \dot{q})]_l\, \mathbf{a}_l \\ &\stackrel{\text{def}}{=} [\dot{q}^T I_i^* \dot{q}]_l\, \mathbf{a}_l \end{aligned} \tag{6.30}$$

where $[\dot{q}^T I_i^* \dot{q}]_l^T \in \mathcal{R}^{10}$ with $(\dot{q}^T I_j^* \dot{q})$ as the jth element. By stacking up the constant inertial parameters of all links into a column vector $\mathbf{a}$, the total sum of N-link kinetic energy could be written as a linear equation of $\mathbf{a}$ as

$$2KE = \left[[\dot{q}^T I_i^* \dot{q}]_1 \; [\dot{q}^T I_i^* \dot{q}]_2 \; \cdots \; [\dot{q}^T I_i^* \dot{q}]_N \right] \mathbf{a} \tag{6.31}$$

with

$$\mathbf{a} = [\mathbf{a}_1^T \; \mathbf{a}_2^T \; \cdots \; \mathbf{a}_N^T]^T \in \mathcal{R}^{10N}$$

A linear relation of $\mathbf{a}$ could also be found for gravitational potential energy expressions. Since among the constant parameters only m and mr show up in gravity force, the derivation of the lth link gravitational potential energy is straightforward as

$$\begin{aligned}
(PE_g)_l &= m\hat{g} \cdot (P + r) \\
&= (\hat{g} \cdot P)m + (\hat{g} \cdot mr) \\
&= [(\hat{g} \cdot P) \; (\hat{g} \cdot \hat{x}) \; (\hat{g} \cdot \hat{y}) \; (\hat{g} \cdot \hat{z})][a_1 \; a_2 \; a_3 \; a_4]^T \\
&= [(\hat{g} \cdot P) \; (\hat{g} \cdot \hat{x}) \; (\hat{g} \cdot \hat{y}) \; (\hat{g} \cdot \hat{z}) \; 0 \; 0 \; \cdots \; 0] \, \mathbf{a}_l \\
&\stackrel{\text{def}}{=} [(\mathcal{PE}_1)(\mathcal{PE}_2) \cdots (\mathcal{PE}_{10})] \, \mathbf{a}_l \\
&\stackrel{\text{def}}{=} [\mathcal{PE}_i]_l \, \mathbf{a}_l
\end{aligned} \tag{6.32}$$

where $\hat{g} \in \mathcal{R}^3$ is vector of gravitational acceleration, and $[\mathcal{PE}_i]_l^T \in \mathcal{R}^{10}$, with $(\mathcal{PE}_j)$ as the jth element, is defined for notational convenience. Therefore,

$$PE = [[\mathcal{PE}_i]_1 \; [\mathcal{PE}_i]_2 \; \cdots \; [\mathcal{PE}_i]_N] \, \mathbf{a} + \sum_{l=1}^{N} (PE_s)_l \tag{6.33}$$

Since the time derivative and partial differential in the Lagrange equation

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{q}}\right) - \frac{\partial KE}{\partial q} + \frac{\partial PE}{\partial q} = \begin{bmatrix} v \\ 0 \end{bmatrix} \tag{6.34}$$

are linear operators and $\mathbf{a}$ is a constant vector, the linear form of $\mathbf{a}$ is maintained in the final dynamic equations. In the above equation, v is an n_θ

vector of generalized actuator force. Before presenting details of the dynamic equations, the resilient potential energy term, $\sum_{l=1}^{N}(PE_s)_l$, in the Lagrange equation is moved to the right-hand side and placed under the vector of generalized force as

$$\begin{bmatrix} v \\ 0 \end{bmatrix} - \frac{\partial}{\partial q}\left(\sum_{l=1}^{N}(PE_s)_l\right) = \begin{bmatrix} v \\ -K\beta \end{bmatrix}$$

where $K \in \mathcal{R}^{n_\beta \times n_\beta}$ is a diagonal stiffness matrix and $\beta \in \mathcal{R}^{n_\beta}$ is vector of compliant deformations. This arrangement makes the dynamic equations exactly linear in **a** in the following Lagrange equation

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{q}}\right) - \frac{\partial KE}{\partial q} + \frac{\partial}{\partial q}\left(\sum_{l=1}^{N}(PE_g)_l\right) = \begin{bmatrix} v \\ -K\beta \end{bmatrix} \tag{6.35}$$

Recalling that in the previous chapters the inertial dynamics with well-known system inertia are expressed as

$$I^*\ddot{q} + \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} v \\ -K\beta \end{bmatrix}$$

similar notation will be used here. In the above derivations, $\frac{1}{2}\dot{q}^T I_i^* \dot{q}$ and $\mathcal{PE}_i$ are respectively the kinetic and gravitational potential energy associated with a unit inertial parameter a_i of a given link, they are deliberately formulated so that every I_i^* is symmetric, hence the dynamic equation associated with a_i could have a familiar form of

$$I_i^*\ddot{q} + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix} = \frac{d}{dt}\left(\frac{\partial(\frac{1}{2}\dot{q}^T I_i^* \dot{q})}{\partial \dot{q}}\right) - \frac{\partial(\frac{1}{2}\dot{q}^T I_i^* \dot{q})}{\partial q} + \frac{\partial \mathcal{PE}_i}{\partial q} \tag{6.36}$$

with

$$\begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix} = \dot{I}_i^*\dot{q} - \frac{\partial(\frac{1}{2}\dot{q}^T I_i^* \dot{q})}{\partial q} + \frac{\partial \mathcal{PE}_i}{\partial q} \tag{6.37}$$

And for the whole manipulator

$$\left[\cdots\left(I_i^*\ddot{q} + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix}\right)_l \cdots\right]\mathbf{a} = \begin{bmatrix} v \\ -K\beta \end{bmatrix} ; \begin{cases} l \in \{1,2,\cdots,N\} \\ i \in \{1,2,\cdots,10\} \end{cases} \tag{6.38}$$

which is further defined as

$$Z_0'\mathfrak{a} = U' \tag{6.39}$$

where $U' = [v^T\ (-K\beta)^T]^T$, and

$$Z_0' = Z_0'(q, \dot{q}, \ddot{q}) = \left[\cdots \left(I_i^* \ddot{q} + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix} \right)_l \cdots \right] \in \mathcal{R}^{(n_\theta + n_\beta) \times 10N}$$

where the subscript l denotes the inertial dynamics of link l, and i means the dynamics of a unit parameter a_i of link l, in which $i = 1, \cdots, 10$. Since the final inertial dynamics are composed by elements I_i^*, $\dot{I}_i^*$, and $\frac{\partial}{\partial q}(\frac{1}{2}\dot{q}^T I_i^* \dot{q})$, they are tabulated in Table 6.1 for $i = 1, 2, \cdots, 10$. Apparently, for given Z_0' and U', $\mathbf{a}$ could be estimated from the linear equation $Z_0'\mathbf{a} = U'$. But Z_0' contains joint acceleration $\ddot{q}$, and general manipulator sensor only provides position q and velocity $\dot{q}$ readings, therefore, the inertial dynamics are integrated with respect to time to remove dependence on acceleration signals. Because integral is a linear operator, the linear relation of $\mathbf{a}$ is intact after integration. By defining $Z_0 = \int Z_0' dt$ and $U = \int U' dt$, the final linear equation is

$$Z_0\mathbf{a} = U \tag{6.40}$$

and $Z_0 = Z_0(q, \dot{q})$. This equation will be used to derive an estimation algorithm for uncertain $\mathbf{a}$ in the next section. The above Z_0 could be derived term by term from the following integration equation

$$\int \left(I_i^* \ddot{q} + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix} \right)_l dt = (I_i^* \dot{q})_l - \int (\dot{I}_i^* \dot{q})_l dt + \int \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix}_l dt \tag{6.41}$$

with $i \in \{1, 2, \cdots, 10\}$ and $l \in \{1, 2, \cdots, N\}$. Notice that $\dot{I}_i^* = \dot{I}_i^*(q, \dot{q})$ is independent of acceleration. As an example, details of the the above dynamic elements are listed in Table 6.1 for link l. A part of Table 6.1 results are used to construct $(f_1)_i$ and $(f_2)_i$ defined by Equation 6.37.

Table 6.1

$$
\begin{aligned}
I_1^* &= m_l\ {}_T G_p^T\ {}_T G_p^T \\
\dot{I}_1^* &= m_l\ {}_T \dot{G}_p^T\ {}_T G_p^T + m_l\ {}_T G_p^T\ {}_T \dot{G}_p^T \\
\frac{\partial(\dot{q}^T I_1^* \dot{q})}{\partial q} &= \dot{q}^T\ {}_T H_p^T\ {}_T G_p \dot{q} + \dot{q}^T\ {}_T G_p^T\ {}_T H_p \dot{q} \\
I_{2\to 4}^* &= -\ {}_T G_p^T \cdot (s \times\ {}_R G_p) - \{\ {}_T G_p^T \cdot (s \times\ {}_R G_p)\}^T ;\ s = \hat{x}, \hat{y}, \hat{z} \\
\dot{I}_{2\to 4}^* &= -\ {}_T \dot{G}_p^T \cdot (s \times\ {}_R G_p) -\ {}_T G_p^T \cdot (\dot{s} \times\ {}_R G_p) \\
&\quad -\ {}_T G_p^T \cdot (s \times\ {}_R \dot{G}_p) \\
&\quad + \{-\ {}_T \dot{G}_p^T \cdot (s \times\ {}_R G_p) -\ {}_T G_p^T \cdot (\dot{s} \times\ {}_R G_p) \\
&\quad -\ {}_T G_p^T \cdot (s \times\ {}_R \dot{G}_p)\}^T \\
\frac{\partial(\dot{q}^T I_{2\to 4}^* \dot{q})}{\partial q} &= -\dot{q}^T\{\ {}_T H_p^T \cdot (s \times\ {}_R G_l)\}\dot{q} \\
&\quad - \dot{q}^T\{\ {}_T G_p^T \cdot (-(s \times\ {}_R G_l) \times\ {}_R G_l)\}\dot{q} \\
&\quad - \dot{q}^T\{\ {}_T G_p^T \cdot (s \times\ {}_R H_l)\}\dot{q} \\
&\quad + \{-\dot{q}^T\{\ {}_T H_p^T \cdot (s \times\ {}_R G_l)\}\dot{q} \\
&\quad - \dot{q}^T\{\ {}_T G_p^T \cdot (-(s \times\ {}_R G_l) \times\ {}_R G_l)\}\dot{q} \\
&\quad - \dot{q}^T\{\ {}_T G_p^T \cdot (s \times\ {}_R H_l)\}\dot{q}\}^T \\
I_{5\to 10}^* &= {}_R G_l^T T_l I_{pi} T_l^T\ {}_R G_l \\
\dot{I}_{5\to 10}^* &= {}_R \dot{G}_l^T T_l I_{pi} T_l^T\ {}_R G_l +\ {}_R G_l^T \dot{T}_l I_{pi} T_l^T\ {}_R G_l \\
&\quad +\ {}_R G_l^T T_l I_{pi} \dot{T}_l^T\ {}_R G_l +\ {}_R G_l^T T_l I_{pi} T_l^T\ {}_R \dot{G}_l \\
\frac{\partial(\dot{q}^T I_{5\to 10}^* \dot{q})}{\partial q} &= \dot{q}^T(\ {}_R H_l^T) T_l I_{pi} T_l^T\ {}_R G_l \dot{q} + \dot{q}^T\ {}_R G_l^T (\nabla T_l) I_{pi} T_l^T\ {}_R G_l \dot{q} \\
&\quad + \dot{q}^T\ {}_R G_l^T T_l I_{pi} (\nabla T_l^T)\ {}_R G_l \dot{q} + \dot{q}^T\ {}_R G_l^T T_l I_{pi} T_l^T (\ {}_R H_l) \dot{q} \\
{}_T \dot{G}_p &= {}_T H_p \dot{q} \\
{}_R \dot{G}_l &= {}_R H_l \dot{q}
\end{aligned}
$$

$$
\begin{aligned}
T_l &= [\hat{x}\ \hat{y}\ \hat{z}] \\
\dot{T}_l &= [(\omega \times \hat{x})\ (\omega \times \hat{y})\ (\omega \times \hat{z})] \\
\nabla T_l &= [(-\hat{x} \times {}_R G_l)\ (-\hat{y} \times {}_R G_l)\ (-\hat{z} \times {}_R G_l)]
\end{aligned}
$$

(end of Table 6.1)

6.2 The Least Square Estimation Method for Constant Linear Parameters

In linear equation $Z_0\mathbf{a} = U$, with $Z_0 \in \mathcal{R}^{(n_\theta+n_\beta)\times 10N}$, $\mathbf{a} \in \mathcal{R}^{10N}$, and $U \in \mathcal{R}^{n_\theta+n_\beta}$, for given Z_0 and U, existence of a unique solution $\mathbf{a}$ relies on the relation between U and the column space of matrix Z_0. If U is in the space spanned by the column vectors of Z_0, then at least one $\mathbf{a}$ satisfies the equation. In contrast, if U is not entirely inside the column space of Z_0 then there is no solution to the equation. However, $Z_0 = Z_0(q, \dot{q})$ which is a nonlinear function of position and velocities, hence its column space changing not only with position but also with speed of motion. Therefore identifying the relation between U and Z_0 directly is not a proper way of solving $\mathbf{a}$. Despite the difficulty of obtaining an exact $\mathbf{a}$, an alternative way is to find an estimate $\hat{\mathbf{a}}$ that has minimal estimation error defined as $||Z_0\hat{\mathbf{a}} - U||$, which is generally known as the least square method. The algorithms of the least square estimation method could be found in [Astrom and Wittenmark, 1989] and [Li and Slotine, 1987]. Since Z_0 is a nonlinear function varying with position and velocity and $\mathbf{a}$ is a constant vector, the time history of Z_0 is included in the optimization process by defining a cost function as

$$J(\hat{\mathbf{a}}(t)) = \int_0^t e^{-\sigma(t-s)}||Z_0(s)\hat{\mathbf{a}}(t) - U(s)||^2 ds \qquad (6.42)$$

in which $e^{-\sigma(t-s)}$ is an exponential forgetting factor with $\sigma > 0$ to weight higher on current error than far past one. To find an optimal $\hat{\mathbf{a}}$, the gradient

of cost function $J(\hat{\mathbf{a}})$ with respect to $\hat{\mathbf{a}}$ is set to zero, which produces

$$\int_0^t e^{-\sigma(t-s)} Z_0^T(s) Z_0(s) ds \hat{\mathbf{a}}(t) = \int_0^t e^{-\sigma(t-s)} Z_0^T(s) U(s) ds \tag{6.43}$$

or

$$R^{-1}\hat{\mathbf{a}} = O$$

with $R^{-1} \in \mathcal{R}^{10N \times 10N}$ and $O \in \mathcal{R}^{10N}$ defined respectively as

$$R^{-1} \stackrel{\text{def}}{=} \int_0^t e^{-\sigma(t-s)} Z_0^T(s) Z_0(s) ds \tag{6.44}$$

$$O \stackrel{\text{def}}{=} \int_0^t e^{-\sigma(t-s)} Z_0^T(s) U(s) ds \tag{6.45}$$

since $Z_0^T Z_0$ is a positive semidefinite matrix, R^{-1} could become positive definite and invertible after integration. This condition is defined as *persistent excitation*. If R exists, then

$$\hat{\mathbf{a}} = RO \tag{6.46}$$

However, the convolution integrals in R^{-1} and O require storage of all past data, and in addition, on-line inversion is required for R, therefore the above approach is ineffective in computing $\hat{\mathbf{a}}$. An alternative and efficient way of computing $\hat{\mathbf{a}}$ is to integrate $\dot{\hat{\mathbf{a}}}$ and $\dot{R}$ recursively by

$$\begin{aligned} \hat{\mathbf{a}}(t+\Delta t) &= \hat{\mathbf{a}}(t) + \int_t^{t+\Delta t} \dot{\hat{\mathbf{a}}} dt \\ R(t+\Delta t) &= R(t) + \int_t^{t+\Delta t} \dot{R} dt \end{aligned}$$

where $\dot{R}$ could be obtained from R^{-1} and its time derivative $(\dot{R^{-1}})$. For persistently exciting system, $R^{-1} > 0$ so $RR^{-1} = I$ whose time differentiation gives $\dot{R}R^{-1} + R(\dot{R^{-1}}) = 0$, hence $\dot{R} = -R(\dot{R^{-1}})R$.

Since in Equation 6.44, both integrand and integral limits are functions of present time t, Leibnitz' rule is applied to find $(\dot{R^{-1}})$, which states

that for a given integration defined as

$$R^{-1} = \int_{a(t)}^{b(t)} \phi(s,t)ds$$

its time derivative is

$$(\dot{R^{-1}}) = \int_{a(t)}^{b(t)} \frac{\partial \phi(s,t)}{\partial t} ds + \phi(b(t),t)\frac{db(t)}{dt} - \phi(a(t),t)\frac{da(t)}{dt}$$

Since in our case $b(t) = t$, $a(t) = 0$ and $\phi(s,t) = e^{-\sigma(t-s)} Z_0^T(s) Z_0(s)$,

$$\begin{aligned} (\dot{R^{-1}}) &= \phi(t,t) + \int_0^t \frac{\partial \phi(s,t)}{\partial t} ds \\ &= Z_0^T(t) Z_0(t) - \sigma R^{-1} \end{aligned} \tag{6.47}$$

which gives

$$\dot{R}(t) = \sigma R(t) - R(t) Z_0^T(t) Z_0(t) R(t) \tag{6.48}$$

and similarly

$$\dot{O} = Z_0^T(t) U(t) - \sigma O \tag{6.49}$$

From Equation 6.46, $\hat{\mathbf{a}} = RO$ which has a time derivative

$$\begin{aligned} \dot{\hat{\mathbf{a}}} &= \dot{R}O + R\dot{O} \\ &= (\sigma R - R Z_0^T Z_0 R)O + R(Z_0^T U - \sigma O) \\ &= R Z_0^T (U - Z_0 R O) \\ &= R Z_0^T (U - Z_0 \hat{\mathbf{a}}) \\ &= - R Z_0^T \varepsilon \end{aligned} \tag{6.50}$$

where ε is the estimation error defined as

$$\begin{aligned} \varepsilon &\stackrel{\text{def}}{=} Z_0 \hat{\mathbf{a}} - U \\ &= Z_0(\hat{\mathbf{a}} - \mathbf{a}) \end{aligned} \tag{6.51}$$

Once Z_0 and U are available in estimation process, $\dot{R}$ and $\dot{\hat{\mathbf{a}}}$ are computed from Equations 6.48 and 6.50 and then integrated over a time step to update R and $\hat{\mathbf{a}}$. One comment on the above update algorithm is about the initial value $R(0)$. At $t = 0$, the upper integral limit of R^{-1} in Equation 6.44 is zero, which means that $R^{-1}(0) = 0$ and $R(0) = \infty$. It is improper to use this initial value to estimate $\mathbf{a}$, so a nonzero but small value is practically chosen to be the initial value of R^{-1}, therefore, $R(0)$ is large but finite. Generally, a diagonal matrix is chosen to be $R^{-1}(0)$. This replacement of initial value has minor effect on the precision of $R^{-1}(t)$ value in the long run, which can be shown by an analytic integration of $(\dot{R^{-1}})$ in Equation 6.47 which produces

$$R^{-1}(t) = e^{-\sigma t}R^{-1}(0) + \int_0^t e^{-\sigma(t-s)}Z_0^T(s)Z_0(s)ds \tag{6.52}$$

the above equation shows that $R^{-1}(0)$ decays exponentially with time and has almost no effect on R^{-1} after a large t.

Since the least square method finds an estimate minimizing the cost function J, it is important to further examine the convergence of the estimate to an exact value. First, let $\tilde{\mathbf{a}} = (\hat{\mathbf{a}} - \mathbf{a})$ be the estimate error, so for a constant $\mathbf{a}$, $\dot{\tilde{\mathbf{a}}} = \dot{\hat{\mathbf{a}}} = RZ_0^T\varepsilon$ by Equation 6.50. Then, a Lyapunov type function is defined as

$$V = \tilde{\mathbf{a}}^T R^{-1}\tilde{\mathbf{a}} \tag{6.53}$$

where R^{-1} is positive definite for a persistently exciting system, hence $V > 0$, $\forall\ \tilde{\mathbf{a}} \neq 0$. It can be shown from Equations 6.47 and 6.51 that

$$\begin{aligned}
\dot{V} &= 2\tilde{\mathbf{a}}^T R^{-1}\dot{\tilde{\mathbf{a}}} + \tilde{\mathbf{a}}^T(\dot{R^{-1}})\tilde{\mathbf{a}} \\
&= -2\tilde{\mathbf{a}}^T R^{-1}(RZ_0^T\varepsilon) + \tilde{\mathbf{a}}^T(-\sigma R^{-1} + Z_0^T Z_0)\tilde{\mathbf{a}} \\
&= -\varepsilon^T\varepsilon - \sigma V
\end{aligned} \tag{6.54}$$

since $\sigma > 0$ and $V > 0$ for all $\tilde{\mathbf{a}} \neq 0$, then $\dot{V} < 0$, $\forall\ \tilde{\mathbf{a}} \neq 0$, which means that the estimation is stable and convergent. Another way of investigating convergence is by direct integration of $\dot{\tilde{\mathbf{a}}}$ defined as

$$\begin{aligned} \dot{\tilde{\mathbf{a}}} = \dot{\hat{\mathbf{a}}} &= -RZ_0^T\varepsilon \\ &= -RZ_0^TZ_0\tilde{\mathbf{a}} \end{aligned} \tag{6.55}$$

From Equation 6.47,

$$Z_0^T(t)Z_0(t) = (\dot{R^{-1}}) + \sigma R^{-1}$$

a substitution of the above expression into Equation 6.55 produces

$$\dot{\tilde{\mathbf{a}}} = -R((\dot{R^{-1}}) + \sigma R^{-1})\tilde{\mathbf{a}}$$

which has an equivalent form of

$$R^{-1}\dot{\tilde{\mathbf{a}}} + (\dot{R^{-1}})\tilde{\mathbf{a}} + \sigma R^{-1}\tilde{\mathbf{a}} = 0$$

After multiplying both sides by $e^{\sigma t}$ and using time differentiation, the above equation becomes

$$\frac{d}{dt}(e^{\sigma t}R^{-1}\tilde{\mathbf{a}}) = 0$$

Let $\tilde{\mathbf{a}}(0)$ denote the initial estimate error, then

$$e^{\sigma t}R^{-1}(t)\tilde{\mathbf{a}}(t) = R^{-1}(0)\tilde{\mathbf{a}}(0)$$

and

$$\tilde{\mathbf{a}}(t) = e^{-\sigma t}R(t)R^{-1}(0)\tilde{\mathbf{a}}(0)$$

Recalling that $\tilde{\mathbf{a}} = (\hat{\mathbf{a}} - \mathbf{a})$,

$$\hat{\mathbf{a}} = \tilde{\mathbf{a}} + \mathbf{a} = \mathbf{a} + e^{-\sigma t}R(t)R^{-1}(0)\tilde{\mathbf{a}}(0) \tag{6.56}$$

for finite $R^{-1}(0)$ and $\tilde{\mathbf{a}}(0)$, $\hat{\mathbf{a}} \rightarrow \mathbf{a}$ as $t \rightarrow \infty$, therefore, for persistently exciting R^{-1}, $\hat{\mathbf{a}}$ approaches to $\mathbf{a}$ as t increases. Hence, convergence of estimate to exact solution is ensured provided that the system is persistently exciting.

Obviously, an essential requirement of the least square method is the condition of persistent excitation, or $R^{-1} > 0$. From Equation 6.44, far past $Z_0^T Z_0$ terms decay exponentially with elapsed time, so near time $Z_0^T Z_0$ terms must have full rank to maintain a positive definite R^{-1}. This means that the column vectors of near time $Z_0^T Z_0$ must span the space that $\mathbf{a}$ resides. This could be verified by a simple examination of Equation 6.44 which is rewritten as

$$R^{-1} = \int_0^{t_1} e^{-\sigma(t-s)} Z_0^T(s) Z_0(s) ds + \int_{t_1}^{t} e^{-\sigma(t-s)} Z_0^T(s) Z_0(s) ds \qquad (6.57)$$

with $0 < t_1 < t$. If R^{-1} stops excitation after t_1, the first integral becomes zero quickly as t increases, but the second term is positive semidefinite, so the inverse of R^{-1} is not ensured. In other words, in order to have an invertible R^{-1}, the second integral must maintain positive definiteness and hence be persistently exciting.

Notice that with persistent excitation, convergence of estimate also can be shown for the case of $\sigma = 0$, i.e., equal weight on all data in Equations 6.44 and 6.45. For a persistently exciting system, R^{-1} grows continuously with time while its inversion R reduces in magnitude and eventually becomes zero. From Equation 6.56, with $\sigma = 0$,

$$\hat{\mathbf{a}} = \mathbf{a} + R(t) R^{-1}(0) \tilde{\mathbf{a}}(0) \qquad (6.58)$$

for finite $R^{-1}(0)$ and $\tilde{\mathbf{a}}(0)$, $\hat{\mathbf{a}} \rightarrow \mathbf{a}$ as $R \rightarrow 0$. But according to Equation 6.56, due to exponential decay, an estimate with exponential forgetting converges

faster than one without exponential forgetting. For estimation without exponential forgetting, the above recursive formula, Equations 6.48 and 6.50, are legitimate after replacing σ with zero.

Application of the least square method requires well-known Z_0 and U. Generally, in an off-line calibration of manipulator inertia, simple commands like sinusoidal or trapezoidal functions could be used to create manipulator motion, from which position and velocity readouts are collected to construct Z_0 and U. However, some inertia might not be activated during a given motion. For example, if one particular link has no rotational movement, its moment of inertia is not excited. If this happens, the system is not persistently excited and estimate may not converge to the exact value. Therefore, more tests involving different manipulator configurations are useful to assure the accuracy of estimation. Once structural inertia are calibrated, they are treated as known parameters, which makes payload inertia the major uncertainty in robot controller. In the next section, an adaptive controller for compliant manipulators is introduced, which conducts on-line estimation of uncertain payload and uses estimates to form control commands for trajectory tracking.

6.3 Adaptive Control Algorithm for Compliant Manipulators with Payload Uncertainty

Before presenting the adaptive control algorithm, compliant manipulator dynamics are reviewed briefly. In the followings, manipulator inertial parameters are assumed well-known and payload is the only uncertainty to the controller. For a compliant manipulator, the dynamic equations, includ-

ing payload, are derived in Chapter 3 as

$$\begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1' \\ f_2 \end{bmatrix} = \begin{bmatrix} v \\ -K\beta \end{bmatrix} \tag{6.59}$$

where $\Lambda_1' \in \mathcal{R}^{n_\theta \times n_\theta}$, $\Sigma \in \mathcal{R}^{n_\beta \times n_\theta}$, and $\Lambda_2 \in \mathcal{R}^{n_\beta \times n_\beta}$ are elements of the generalized inertial matrix composed of nonlinear functions of actual joint displacements, $\theta \in \mathcal{R}^{n_\theta}$, and vibratory modal amplitudes, $\beta \in \mathcal{R}^{n_\beta}$. Also $f_1' \in \mathcal{R}^{n_\theta}$ and $f_2 \in \mathcal{R}^{n_\beta}$ are coupling terms containing Coriolis, centrifugal, and gravitational forces, which are functions of θ, $\dot{\theta}$, β, and $\dot{\beta}$. Note that actuator dynamics are excluded from the above expression. Let subscript k denote the known system parameter and subscript u represent the uncertain payload dynamic parameter, then the generalized inertial matrix in Equation 6.59 could be divided into

$$\begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} = \begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_k + \begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_u \tag{6.60}$$

and $[f_1'^T \; f_2^T]^T$ is composed of two parts as

$$\begin{bmatrix} f_1' \\ f_2 \end{bmatrix} = \begin{bmatrix} f_1' \\ f_2 \end{bmatrix}_k + \begin{bmatrix} f_1' \\ f_2 \end{bmatrix}_u \tag{6.61}$$

For uncertain payload, section 6.1 shows that a linear expression of payload inertial dynamics are given by

$$Z_0'\mathbf{a} = \left[\cdots \left(I_i^* \ddot{q} + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix} \right) \cdots \right] \mathbf{a} \; ; \; i \in \{1, \cdots, 10\} \tag{6.62}$$

where $\mathbf{a} \in \mathcal{R}^{10}$ contains payload constant inertial parameters

$$[a_1 \; a_2 \; a_3 \; a_4 \; a_5 \; a_6 \; a_7 \; a_8 \; a_9 \; a_{10}] = [m \; mr_x \; mr_y \; mr_z \; I_{pxx} \; I_{pxy} \; I_{pxz} \; I_{pyy} \; I_{pyz} \; I_{pzz}]$$

and $q = [\theta^T \; \beta^T]^T$. Since q is independent of $\mathbf{a}$, $Z_0'\mathbf{a}$ has an equivalent form of

$$Z_0'\mathbf{a} = \begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_u \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1' \\ f_2 \end{bmatrix}_u \tag{6.63}$$

in which

$$\begin{bmatrix} \Lambda_1' & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_u = \sum_{i=1}^{10} I_i^* a_i \tag{6.64}$$

and

$$\begin{bmatrix} f_1' \\ f_2 \end{bmatrix}_u = \sum_{i=1}^{10} \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix} a_i \tag{6.65}$$

Since it is preferable to have a complete system dynamic description in controller design, actuator dynamics are added to the above equations, which are defined in Chapter 3 as

$$J\ddot{\theta} + D\dot{\theta} + K_c\theta + v = u$$

in which the generalized actuator force v is defined in Equation 6.59 as

$$v = [\Lambda_1' \ \Sigma^T]_k \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + (f_1')_k + [\Lambda_1' \ \Sigma^T]_u \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + (f_1')_u \tag{6.66}$$

Substituting the above v into actuator dynamics and defining

$$(\Lambda_1)_k = (\Lambda_1')_k + J$$

$$(f_1)_k = (f_1')_k + D\dot{\theta} + K_c\theta$$

$$(\Lambda_1)_u = (\Lambda_1')_u$$

$$(f_1)_u = (f_1')_u$$

the total system dynamics become

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} u \\ -K\beta \end{bmatrix} \tag{6.67}$$

with

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} = \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_k + \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_u \tag{6.68}$$

$$\begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_k + \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_u \tag{6.69}$$

In the following control law development, uncertain payload inertial parameters will be estimated on-line. For the purpose of distinction, a hat, $(\hat{\cdot})$, will be used to denote estimate of an uncertain element, and tilde, $(\tilde{\cdot})$, will represent the error of estimate. For example, $\hat{\mathbf{a}}$ is the estimate of $\mathbf{a}$ and $\tilde{\mathbf{a}} = (\hat{\mathbf{a}} - \mathbf{a})$. Similarly, for the uncertain term in Equation 6.64 the estimate and estimation error are expressed respectively as

$$\begin{bmatrix} \hat{\Lambda}_1 & \hat{\Sigma}^T \\ \hat{\Sigma} & \hat{\Lambda}_2 \end{bmatrix}_u = \sum_{i=1}^{10} I_i^* \, \hat{a}_i \tag{6.70}$$

and

$$\begin{aligned} \begin{bmatrix} \tilde{\Lambda}_1 & \tilde{\Sigma}^T \\ \tilde{\Sigma} & \tilde{\Lambda}_2 \end{bmatrix}_u &= \sum_{i=1}^{10} I_i^* \, (\hat{a}_i - a_i) \\ &= \sum_{i=1}^{10} I_i^* \, \tilde{a}_i \end{aligned}$$

and so forth.

The adaptive controller for dynamic Equation 6.67 will be constructed from two considerations. First, it is desirable to assign each nominal and vibratory mode a specially designed acceleration composed by feedback of position and velocity values and feedforward of desired position, velocity, and acceleration. Ideally, this assignment will produce an error-driven dynamic system which has an exponential decay response after proper selection of feedforward and feedback gains. If accomplished, nominal displacement would converge to the desired trajectory and structural oscillation would be eliminated simultaneously. However, the system has $n_\theta + n_\beta$ degrees of freedom but only has n_θ actuator inputs. The acceleration assignment from actuator inputs is a mapping from $\mathcal{R}^{n_\theta}$ space to $\mathcal{R}^{n_\theta + n_\beta}$ space, and finding exact inputs for the designed accelerations is equivalent to solving an overdetermined problem, therefore, unless the assigned acceleration is in the range of mapping, no solution or actuator inputs could generate the ideal result.

Since a direct assignment of designed acceleration is difficult due to dimensional mismatch, the Lyapunov's second method is applied to assist controller design. The merit of the method, besides simplicity, is that it defines stability criteria in R^1 space, which will eventually transfer the control design from an overdetermined to a redundant problem. In applying the method, a quadratic form of error state is chosen as the Lyapunov function which has a positive scalar value for all nonzero error states. The actuator inputs are structured so that the Lyapunov function has negative time derivative as long as error state remains nonzero. Consequently, error state approaches to zero asymptotically, and convergence of nominal and vibratory displacements to the desired states and estimates to actual values are ensured.

According to the first consideration, for desired trajectories θ_r and β_r, the designed accelerations are defined as

$$\begin{aligned} \ddot{\theta}_d &= \ddot{\theta}_r - K_{p\theta}(\dot{\theta} - \dot{\theta}_r) - K_{I\theta}(\theta - \theta_r) \\ \ddot{\beta}_d &= \ddot{\beta}_r - K_{p\beta}(\dot{\beta} - \dot{\beta}_r) - K_{I\beta}(\beta - \beta_r) \end{aligned} \tag{6.71}$$

where $K_{p\theta}$, $K_{I\theta} \in \mathcal{R}^{n_\theta \times n_\theta}$ and $K_{p\beta}$, $K_{I\beta} \in \mathcal{R}^{n_\beta \times n_\beta}$ are diagonal constant gain matrices. Let $e_\theta \in \mathcal{R}^{n_\theta}$ and $e_\beta \in \mathcal{R}^{n_\beta}$ be the error states defined as

$$e_\theta = (\dot{\theta} - \dot{\theta}_r) + K_{p\theta}(\theta - \theta_r) + K_{I\theta}\int(\theta - \theta_r)dt \tag{6.72}$$

$$e_\beta = (\dot{\beta} - \dot{\beta}_r) + K_{p\beta}(\beta - \beta_r) + K_{I\beta}\int(\beta - \beta_r)dt \tag{6.73}$$

then $\dot{e}_\theta = \ddot{\theta} - \ddot{\theta}_d$ and $\dot{e}_\beta = \ddot{\beta} - \ddot{\beta}_d$. Equations 6.72 and 6.73 are two simple ordinary differential equations with stable eigenvalues after proper selection of gain matrices. For zero e_θ and e_β, the differential equations are homogeneous, and transient response of $(\theta - \theta_r)$ and $(\beta - \beta_r)$ decay exponentially. Furthermore, by taking the Laplace transform of the above equations and

using the final-value theorem, it can be shown that $e_\theta \rightarrow 0 \Rightarrow \theta \rightarrow \theta_r$ and $e_\beta \rightarrow 0 \Rightarrow \beta \rightarrow \beta_r$. This implies that if nominal and vibratory accelerations obtain the designed values, θ will track the desired trajectory θ_r and vibrations will be suppressed by setting $\ddot{\beta}_r = \dot{\beta}_r = \beta_r = 0$. Since the designed acceleration is a vector in $\mathcal{R}^{n_\theta + n_\beta}$ and the control vector is in $\mathcal{R}^{n_\theta}$ space, it is generally difficult to create the desired assignment, therefore, the Lyapunov's second method is employed to assist controller design. Since exact payload is uncertain, estimation of payload parameters will be included as part of the design criteria.

First, a dynamic term defined as

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \tag{6.74}$$

is subtracted from both sides of Equation 6.67, which results in

$$\begin{aligned} \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \dot{e}_\theta \\ \dot{e}_\beta \end{bmatrix} &= \begin{bmatrix} u \\ -K\beta \end{bmatrix} - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \\ &= \begin{bmatrix} u \\ -K\beta \end{bmatrix} - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_k \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_k \\ &\quad - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_u \end{aligned} \tag{6.75}$$

then by adding and subtracting an estimated term

$$\begin{bmatrix} \hat{\Lambda}_1 & \hat{\Sigma}^T \\ \hat{\Sigma} & \hat{\Lambda}_2 \end{bmatrix}_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + \begin{bmatrix} \hat{f}_1 \\ \hat{f}_2 \end{bmatrix}_u \tag{6.76}$$

to the right-hand side of Equation 6.75, system dynamic equations become

$$\begin{aligned} \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \dot{e}_\theta \\ \dot{e}_\beta \end{bmatrix} &= \begin{bmatrix} u \\ -K\beta \end{bmatrix} - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_k \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_k \\ &\quad - \begin{bmatrix} \hat{\Lambda}_1 & \hat{\Sigma}^T \\ \hat{\Sigma} & \hat{\Lambda}_2 \end{bmatrix}_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} - \begin{bmatrix} \hat{f}_1 \\ \hat{f}_2 \end{bmatrix}_u \\ &\quad + \begin{bmatrix} \tilde{\Lambda}_1 & \tilde{\Sigma}^T \\ \tilde{\Sigma} & \tilde{\Lambda}_2 \end{bmatrix}_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + \begin{bmatrix} \tilde{f}_1 \\ \tilde{f}_2 \end{bmatrix}_u \end{aligned} \tag{6.77}$$

The input u is chosen in a composite structure as

$$u = u_1 + u_2 \tag{6.78}$$

where u_1 is defined as

$$u_1 = [\Lambda_1 \ \Sigma^T]_k \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (f_1)_k + [\hat{\Lambda}_1 \ \hat{\Sigma}^T]_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (\hat{f}_1)_u \tag{6.79}$$

By introducing two notations Z_1 and S_1 as

$$\begin{aligned} Z_1\tilde{\mathbf{a}} &\stackrel{\text{def}}{=} \begin{bmatrix} \tilde{\Lambda}_1 & \tilde{\Sigma}^T \\ \tilde{\Sigma} & \tilde{\Lambda}_2 \end{bmatrix}_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + \begin{bmatrix} \tilde{f}_1 \\ \tilde{f}_2 \end{bmatrix}_u \\ &= \sum_{i=1}^{10} \left(I_i^* \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix} \right) \tilde{a}_i \end{aligned} \tag{6.80}$$

and

$$S_1 \stackrel{\text{def}}{=} [\Sigma \ \Lambda_2]_k \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (f_2)_k + [\hat{\Sigma} \ \hat{\Lambda}_2]_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (\hat{f}_2)_u \tag{6.81}$$

with $Z_1 \in \mathcal{R}^{(n_\theta + n_\beta) \times 10}$ and $S_1 \in \mathcal{R}^{n_\beta}$, then Equation 6.77 is reduced to

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \dot{e}_\theta \\ \dot{e}_\beta \end{bmatrix} = \begin{bmatrix} u_2 \\ -K\beta - S_1 \end{bmatrix} + Z_1\tilde{\mathbf{a}} \tag{6.82}$$

To find a u_2 stabilizing the system in Equation 6.82 and eliminating payload estimate error $\tilde{\mathbf{a}}$, a Lyapunov function is selected below. Since the inertial matrix in Equation 6.68 is positive definite, the selected Lyapunov function is

$$V = \frac{1}{2}\rho \mathbf{e}^T \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \mathbf{e} + \frac{1}{2}\tilde{\mathbf{a}}^T R^{-1} \tilde{\mathbf{a}} \tag{6.83}$$

where $\mathbf{e} = [e_\theta^T \ e_\beta^T]^T \in \mathcal{R}^{n_\theta + n_\beta}$, and ρ is a positive scalar constant whose function is to assist payload estimation as shown later. The purpose of adding the quadratic term of $\tilde{\mathbf{a}}$ in V is to include payload estimation in controller design. A similar Lyapunov function is proposed by [Slotine and Li, 1988] to control rigid robotic manipulators. In the above equation, R^{-1} is defined in the last section as

$$R^{-1}(t) = \int_0^t e^{-\sigma(t-s)} Z_0^T(s) Z_0(s) ds$$

which is a positive definite matrix for a persistently exciting system. However, unlike the last section derivations, now the only uncertainty is payload instead of the whole structural inertia, therefore, Z_0 is redefined. From Equations 6.63, 6.67, 6.68, and 6.69

$$\begin{aligned} Z_0'\mathbf{a} &= \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_u \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} + \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_u \\ &= \begin{bmatrix} u \\ -K\beta \end{bmatrix} - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_k \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_k \\ &\stackrel{\text{def}}{=} U' \end{aligned} \tag{6.84}$$

then $Z_0 = \int Z_0' dt$ and $U = \int U' dt$. The integration is to remove the requirement of acceleration signal in Z_0'. From the above relations, an estimate error ε is defined as

$$\begin{aligned} \varepsilon &= Z_0\hat{\mathbf{a}} - U \\ &= Z_0(\hat{\mathbf{a}} - \mathbf{a}) \\ &= Z_0\tilde{\mathbf{a}} \end{aligned} \tag{6.85}$$

From Equation 6.83, the time derivative of V is

$$\dot{V} = \rho\mathbf{e}^T \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \dot{\mathbf{e}} + \frac{1}{2}\rho\mathbf{e}^T \begin{bmatrix} \dot{\Lambda}_1 & \dot{\Sigma}^T \\ \dot{\Sigma} & \dot{\Lambda}_2 \end{bmatrix} \mathbf{e} + \tilde{\mathbf{a}}^T R^{-1}\dot{\tilde{\mathbf{a}}} + \frac{1}{2}\tilde{\mathbf{a}}^T(\dot{R^{-1}})\tilde{\mathbf{a}} \tag{6.86}$$

where the derivative of the generalized inertial matrix could be decomposed into known and uncertain parts as

$$\begin{bmatrix} \dot{\Lambda}_1 & \dot{\Sigma}^T \\ \dot{\Sigma} & \dot{\Lambda}_2 \end{bmatrix} = \begin{bmatrix} \dot{\Lambda}_1 & \dot{\Sigma}^T \\ \dot{\Sigma} & \dot{\Lambda}_2 \end{bmatrix}_k + \begin{bmatrix} \dot{\Lambda}_1 & \dot{\Sigma}^T \\ \dot{\Sigma} & \dot{\Lambda}_2 \end{bmatrix}_u \tag{6.87}$$

and recalling that

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_u = \sum_{i=1}^{10} I_i^* a_i$$

then for constant $\mathbf{a}$

$$\begin{bmatrix} \dot{\Lambda}_1 & \dot{\Sigma}^T \\ \dot{\Sigma} & \dot{\Lambda}_2 \end{bmatrix}_u = \sum_{i=1}^{10} \dot{I}_i^* a_i \tag{6.88}$$

According to the above result, the associated quadratic term in $\dot{V}$ could be replaced by a compact notation $Z_2\mathbf{a}$ defined as

$$\begin{aligned} Z_2\mathbf{a} &\stackrel{\text{def}}{=} \mathbf{e}^T \begin{bmatrix} \dot{\Lambda}_1 & \dot{\Sigma}^T \\ \dot{\Sigma} & \dot{\Lambda}_2 \end{bmatrix}_u \mathbf{e} \\ &= \sum_{i=1}^{10} (\mathbf{e}^T \dot{I}_i^* \mathbf{e})\, a_i \end{aligned} \tag{6.89}$$

$$\begin{aligned} Z_2\hat{\mathbf{a}} &\stackrel{\text{def}}{=} \mathbf{e}^T \begin{bmatrix} \hat{\dot{\Lambda}}_1 & \hat{\dot{\Sigma}}^T \\ \hat{\dot{\Sigma}} & \hat{\dot{\Lambda}}_2 \end{bmatrix}_u \mathbf{e} \\ &= \sum_{i=1}^{10} (\mathbf{e}^T \dot{I}_i^* \mathbf{e})\, \hat{a}_i \end{aligned} \tag{6.90}$$

$$\begin{aligned} Z_2\tilde{\mathbf{a}} &\stackrel{\text{def}}{=} \mathbf{e}^T \begin{bmatrix} \tilde{\dot{\Lambda}}_1 & \tilde{\dot{\Sigma}}^T \\ \tilde{\dot{\Sigma}} & \tilde{\dot{\Lambda}}_2 \end{bmatrix}_u \mathbf{e} \\ &= \sum_{i=1}^{10} (\mathbf{e}^T \dot{I}_i^* \mathbf{e})\, \tilde{a}_i \end{aligned} \tag{6.91}$$

and

$$Z_2\mathbf{a} = Z_2\hat{\mathbf{a}} - Z_2\tilde{\mathbf{a}} \tag{6.92}$$

where $Z_2 \in \mathcal{R}^{10}$. Notice that in the above definitions $\mathbf{a}$ is replaced by $\hat{\mathbf{a}}$ and $\tilde{\mathbf{a}}$ directly after time derivative, which means that no $\dot{\hat{\mathbf{a}}}$ or $\dot{\tilde{\mathbf{a}}}$ term appears in the above operations, so hat and tilde are placed on the top of $\dot{\Lambda}_1$, $\dot{\Sigma}$ and $\dot{\Lambda}_2$ to denote this relation. After substituting the above expressions into Equation 6.86, $\dot{V}$ becomes

$$\begin{aligned} \dot{V} &= \rho \mathbf{e}^T \left\{ \begin{bmatrix} u_2 \\ -K\beta - S_1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \dot{\Lambda}_1 & \dot{\Sigma}^T \\ \dot{\Sigma} & \dot{\Lambda}_2 \end{bmatrix}_k \mathbf{e} + \frac{1}{2} \begin{bmatrix} \hat{\dot{\Lambda}}_1 & \hat{\dot{\Sigma}}^T \\ \hat{\dot{\Sigma}} & \hat{\dot{\Lambda}}_2 \end{bmatrix}_u \mathbf{e} \right\} \\ &\quad + \rho \mathbf{e}^T Z_1 \tilde{\mathbf{a}} + \tilde{\mathbf{a}}^T R^{-1} \dot{\tilde{\mathbf{a}}} + \frac{1}{2} \tilde{\mathbf{a}}^T (\dot{R^{-1}}) \tilde{\mathbf{a}} - \frac{1}{2} \rho Z_2 \tilde{\mathbf{a}} \end{aligned} \tag{6.93}$$

Further simplification of $\dot{V}$ expression is possible by defining u_2 as

$$u_2 = u_3 - \frac{1}{2}[\dot{\Lambda}_1 \ \dot{\Sigma}^T]_k \mathbf{e} - \frac{1}{2}[\hat{\dot{\Lambda}}_1 \ \hat{\dot{\Sigma}}^T]_u \mathbf{e} \tag{6.94}$$

and introducing a new notation S_2 defined as

$$S_2 = \frac{1}{2}[\dot{\Sigma}\ \dot{\Lambda}_2]_k \mathbf{e} + \frac{1}{2}[\hat{\dot{\Sigma}}\ \hat{\dot{\Lambda}}_2]_u \mathbf{e} \tag{6.95}$$

also using the last section derivations that $(\dot{R^{-1}}) = -\sigma R^{-1} + Z_0^T Z_0$, and $Z_0\tilde{\mathbf{a}} = \varepsilon$. By combining these results, $\dot{V}$ becomes

$$\begin{aligned} \dot{V} &= \rho \mathbf{e}^T \begin{bmatrix} u_3 \\ -K\beta - S_1 + S_2 \end{bmatrix} + \tilde{\mathbf{a}}^T (R^{-1}\dot{\tilde{\mathbf{a}}} - \frac{1}{2}\rho Z_2^T + \rho Z_1^T \mathbf{e}) \\ &\quad - \frac{1}{2}\sigma \tilde{\mathbf{a}}^T R^{-1} \tilde{\mathbf{a}} + \frac{1}{2}\varepsilon^T \varepsilon \end{aligned} \tag{6.96}$$

Since the goal is to have a negative $\dot{V}$ for all nonzero $\mathbf{e}$ and $\tilde{\mathbf{a}}$, $\dot{\tilde{\mathbf{a}}}$ is selected to create a relation that

$$R^{-1}\dot{\tilde{\mathbf{a}}} - \frac{1}{2}\rho Z_2^T + \rho Z_1^T \mathbf{e} = -Z_0^T \varepsilon \tag{6.97}$$

which means that parameter updating formula is

$$\dot{\tilde{\mathbf{a}}} = R(\frac{1}{2}\rho Z_2^T - \rho Z_1^T \mathbf{e} - Z_0^T \varepsilon) \tag{6.98}$$

Additionally, by an earlier definition $\varepsilon^T = \tilde{\mathbf{a}}^T Z_0^T$, $\dot{V}$ results in

$$\dot{V} = \rho \mathbf{e}^T \begin{bmatrix} u_3 \\ -K\beta - S_1 + S_2 \end{bmatrix} - \frac{1}{2}\varepsilon^T \varepsilon - \frac{1}{2}\sigma \tilde{\mathbf{a}}^T R^{-1} \tilde{\mathbf{a}} \tag{6.99}$$

where the last two terms in the right-hand side are in negative quadratic form, therefore, u_3 is chosen to produce a negative first term. Since the first term is a scalar value and could be expanded as

$$\rho \mathbf{e}^T \begin{bmatrix} u_3 \\ -K\beta - S_1 + S_2 \end{bmatrix} = \rho\{e_\theta^T u_3 + e_\beta^T(-K\beta - S_1 + S_2)\} \tag{6.100}$$

then u_3 could be chosen so that

$$e_\theta^T u_3 + e_\beta^T(-K\beta - S_1 + S_2) = -\frac{\alpha_1}{\rho}\mathbf{e}^T \mathbf{e} \tag{6.101}$$

with α_1 a positive constant, therefore,

$$\dot{V} = -\alpha_1 \mathbf{e}^T \mathbf{e} - \frac{1}{2}\varepsilon^T \varepsilon - \frac{1}{2}\sigma \tilde{\mathbf{a}}^T R^{-1} \tilde{\mathbf{a}} \tag{6.102}$$

In the above expression, $\dot{V} < 0$, $\forall \mathbf{e}$ and $\tilde{\mathbf{a}} \neq 0$, which means that the designed controller will make $[e_\theta^T \; e_\beta^T]^T \rightarrow 0$, or implicitly $\theta \rightarrow \theta_r$ and $\beta \rightarrow \beta_r$, and also $\hat{\mathbf{a}} \rightarrow \mathbf{a}$.

Since Equation 6.101 is a scalar function of n_θ-vector u_3, finding a u_3 satisfying the equation is solving a redundancy problem, hence more than one solution exists for u_3. Therefore, additional criteria could be enforced in the selection process. Here, u_3 is chosen to have a minimum Euclidean norm value $\|u_3\|$. By defining

$$\gamma_1 = e_\beta^T(-K\beta - S_1 + S_2) + \frac{\alpha_1}{\rho}\mathbf{e}^T\mathbf{e} \tag{6.103}$$

Equation 6.101 could be rewritten as

$$e_\theta^T u_3 + \gamma_1 = 0 \tag{6.104}$$

which has a minimum norm solution

$$u_3 = -\frac{\gamma_1}{e_\theta^T e_\theta} e_\theta \tag{6.105}$$

for $e_\theta \neq 0$. Notice from Equation 6.99 that when $e_\theta = 0$, u_3 has no effect on $\dot{V}$ and a negative $\dot{V}$ is not ensured. To resolve this problem, the Lyapunov equation could be modified by adding a positive term $\frac{1}{2}u_3^T u_3$ into Equation 6.83 which becomes

$$V = \frac{1}{2}\rho \mathbf{e}^T \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \mathbf{e} + \frac{1}{2}\tilde{\mathbf{a}}^T R^{-1} \tilde{\mathbf{a}} + \frac{1}{2}u_3^T u_3 \tag{6.106}$$

then following the above derivations,

$$\dot{V} = \rho e_\theta^T u_3 + \rho e_\beta^T(-K\beta - S_1 + S_2) + u_3^T \dot{u}_3 - \frac{1}{2}\varepsilon^T \varepsilon - \frac{1}{2}\sigma \tilde{\mathbf{a}}^T R^{-1} \tilde{\mathbf{a}} \tag{6.107}$$

in which $\dot{u}_3$ will be chosen to satisfy the equation

$$\rho e_\theta^T u_3 + \rho e_\beta^T(-K\beta - S_1 + S_2) + u_3^T \dot{u}_3 = -\alpha_2 u_3^T u_3 - \alpha_1 \mathbf{e}^T \mathbf{e} \tag{6.108}$$

so that

$$\dot{V} = -\alpha_1 \mathbf{e}^T \mathbf{e} - \alpha_2 u_3^T u_3 - \frac{1}{2}\varepsilon^T \varepsilon - \frac{1}{2}\sigma \tilde{\mathbf{a}}^T R^{-1} \tilde{\mathbf{a}} \tag{6.109}$$

where $\alpha_2 > 0$. Again minimum $\|\dot{u}_3\|$ criterion is applied to assist selection of $\dot{u}_3$. By defining

$$\gamma_2 = \rho e_\theta^T u_3 + \rho e_\beta^T(-K\beta - S_1 + S_2) + \alpha_2 u_3^T u_3 + \alpha_1 \mathbf{e}^T \mathbf{e} \tag{6.110}$$

the solution $\dot{u}_3$ is

$$\dot{u}_3 = -\frac{\gamma_2}{\|u_3\|^2} u_3 \tag{6.111}$$

provided that $u_3 \neq 0$. In the modified Lyapunov function, $\dot{V} < 0$ for all $\mathbf{e}$ and $\tilde{\mathbf{a}} \neq 0$ is ensured for $u_3 \neq 0$. Since u_3 is the control input, it could be manipulated directly to have a nonzero value during control process. Or from Equation 6.111, since $\dot{u}_3$ is proportional to γ_2 defined in Equation 6.110, by choosing very small ρ, α_1, and α_2, u_3 varies slowly and could have a nonzero value. Since both actions require adjusting u_3 at the beginning and end of motion which might create disturbance to the system especially when nominal joints are at steady state, the first controller is preferable. Notice that although the first controller can not guarantee a negative $\dot{V}$ when $e_\theta = 0$, yet at $e_\theta = 0$ the nominal trajectory θ has reached steady state θ_r, and estimation should converge to the exact value in the early stage of control for a persistently exciting system, therefore, control could be ceased, which allows structural damping to dissipate residual oscillation naturally.

In Equation 6.98, the estimate update equation is defined as

$$\dot{\tilde{\mathbf{a}}} = R(\frac{1}{2}\rho Z_2^T - \rho Z_1^T \mathbf{e} - Z_0^T \varepsilon)$$

and in the least square method the update equation defined in Equation 6.50 is

$$\dot{\tilde{\mathbf{a}}} = -RZ_0^T\varepsilon$$

for a constant $\mathbf{a}$. By comparing both results, it is apparent that with a very small ρ, the estimation derived by the Lyapunov method will act like a least square result which is designed purely for estimation. Another advantage of using small ρ to improve estimation performance can be seen from Equation 6.99 which states that

$$\dot{V} = \rho\mathbf{e}^T \begin{bmatrix} u_3 \\ -K\beta - S_1 + S_2 \end{bmatrix} - \frac{1}{2}\varepsilon^T\varepsilon - \frac{1}{2}\sigma\tilde{\mathbf{a}}^T R^{-1}\tilde{\mathbf{a}}$$

in the case that $e_\theta = 0$ and $e_\beta \neq 0$ the above equation becomes

$$\dot{V} = \rho e_\beta^T(-K\beta - S_1 + S_2) - \frac{1}{2}\varepsilon^T\varepsilon - \frac{1}{2}\sigma\tilde{\mathbf{a}}^T R^{-1}\tilde{\mathbf{a}} \qquad (6.112)$$

and $\dot{V} \geq 0$ occurs at

$$\rho e_\beta^T(-K\beta - S_1 + S_2) \geq \frac{1}{2}\varepsilon^T\varepsilon + \frac{1}{2}\sigma\tilde{\mathbf{a}}^T R^{-1}\tilde{\mathbf{a}}$$

thence convergence of e_β and $\tilde{\mathbf{a}}$ are not ensured. Recalling that $\varepsilon = Z_0\tilde{\mathbf{a}}$, the above inequality could be rewritten as

$$\rho e_\beta^T(-K\beta - S_1 + S_2) \geq \frac{1}{2}\tilde{\mathbf{a}}^T(Z_0^T Z_0 + \sigma R^{-1})\tilde{\mathbf{a}}$$

which could be interpreted geometrically as a *ball* of $\tilde{\mathbf{a}}$ bounded by $\rho e_\beta^T(-K\beta - S_1 + S_2)$. So when $\tilde{\mathbf{a}}$ enters the ball defined by ρ, convergence of estimation is not justified because of $\dot{V} \geq 0$. But a small ρ will reduce the size of unjustified region and result in better estimation. Hence, a small ρ will be adopted in the later case studies. The adaptive control system block diagram is shown in Figure 6.2.

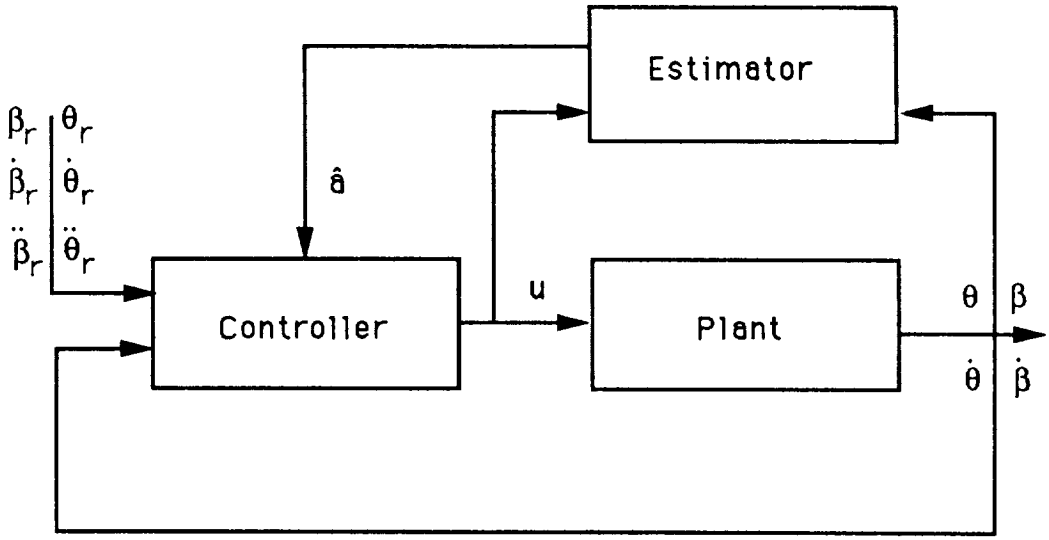


Figure 6.2 Adaptive Control Block Diagram

6.4 Numerical Case Studies: Case 1

Verification of the designed adaptive controller is conducted numerically on the motion control of two flexible manipulators carrying uncertain payload. The first model is a three DOF manipulator modeled with three lumped link compliances as depicted in Figure 6.3, whose system parameters are listed in following tables.

Table 6.2 3-Link Model Kinematic Parameters

	Link Length (m) (x, y, z)	Center of Mass (m) (x, y, z)	Offset Angle (deg)
Link 1	(0, 0, 1)	(0, 0, -0.5)	0
Link 2	(1.2, 0, 0)	(0.5, 0, 0)	90
Link 3	(1.4, 0, 0)	(0.5, 0, 0)	0

Table 6.3 3-Link Model Inertial Parameters

	Mass (kg)	Moment of Inertia ($kg.m^2$) (I_{xx}, I_{yy}, I_{zz})
Link 1	300	(0, 0, 30)
Link 2	680	(6, 53, 44)
Link 3	450	(50, 8, 50)

Table 6.4 3-Link Model Actuator Parameters

	Joint 1	Joint 2	Joint 3
Inertia ($10^{-3} \cdot kg \cdot m^2$)	4.2	2.1	2.1
Damping ($N \cdot m/(rad/s)$)	0.4	0.4	0.3
Resistance (ohm)	0.8	0.8	0.8
Torque Constant ($volt/(rad/s)$)	20	20	14
Gear ratio	100	100	100
Back emf Constant ($N \cdot m/amp$)	0.5	0.5	0.4

and the exact payload and initial estimate are

Table 6.5 Actual Payload parameters and Initial Estimates

	m	mr_x	mr_y	mr_z	I_{pxx}	I_{pxy}	I_{pxz}	I_{pyy}	I_{pyz}	I_{pzz}
$a_{1\to10}$	90	9	9	9	10	5	5	10	5	10
$\hat{a}_{1\to10}(t=0)$	0	0	0	0	0	0	0	0	0	0

The stiffness matrix is $K = \text{diag}[(4000000)_3]$ in N/m, where $(\cdot)_n$ denotes n repeated diagonal element. The simulation procedure is itemized in the following steps.

6.4.1 Step 1: Selection of Nominal Reference Trajectories

The reference trajectory selected for each nominal joint is similar to the fifth order polynomial function used in the last chapter, which is repeated below

$$\begin{aligned} \theta_r(t,t_f) &= \Delta\theta f(t,t_f) + \theta_{r0} \\ f(t,t_f) &= 6\frac{t^5}{t_f^5} - 15\frac{t^4}{t_f^4} + 10\frac{t^3}{t_f^3} \end{aligned} \tag{6.113}$$

where t_f is the traveling time of reference joint displacement, and $f(t,t_f)$ is a normalized polynomial designed with zero velocity and acceleration at $t = 0$ and $t = t_f$ to avoid shocks that might cause structural resonance. In the above equation, $\Delta\theta$ represents the reference joint displacement, and θ_{r0} is the starting point. In the simulation, an initial positional error is added to check the tracking performance, that is, the actual joint will start from a point denoted by θ_0 that is different from the reference initial θ_{r0}. The positional parameters used in this example are tabulated in the next table.

Table 6.6 3-Link Model Trajectory Parameters

	$t_f(sec)$	$\Delta\theta(deg)$	θ_{r0}	θ_0	$\theta_0 - \theta_{r0}$
Joint 1	10	60	0	6	6
joint 2	10	70	0	6	6
Joint 3	10	50	0	6	6

6.4.2 Step 2: Selection of Designed Acceleration

From Equation 6.71, the designed accelerations are defined as

$$\ddot{\theta}_d = \ddot{\theta}_r - K_{p\theta}(\dot{\theta} - \dot{\theta}_r) - K_{I\theta}(\theta - \theta_r)$$

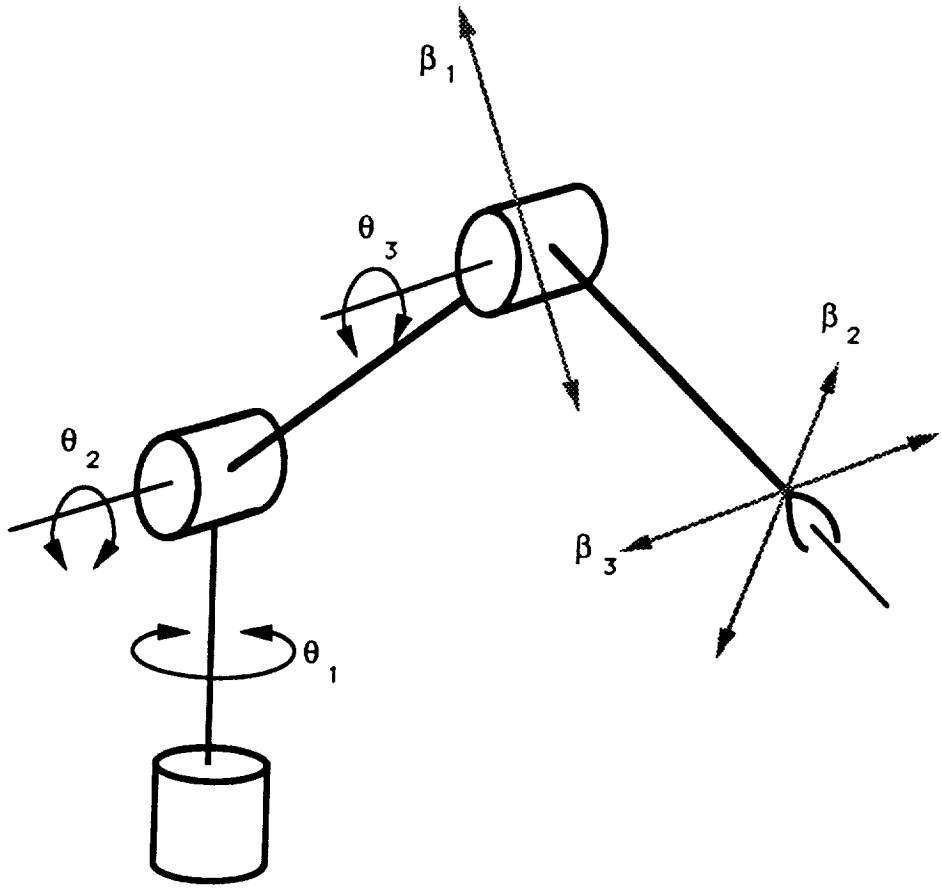


Figure 6.3 Three-Link Robot Modeled with Three Link Compliances

$$\ddot{\beta}_d = \ddot{\beta}_r - K_{p\beta}(\dot{\beta} - \dot{\beta}_r) - K_{I\beta}(\beta - \beta_r)$$

in which θ_r is the desired trajectory given in the last step, and $\dot{\theta}_r$ and $\ddot{\theta}_r$ come from consecutive time derivatives of Equation 6.113. In the designed vibratory mode acceleration, the desired vibratory mode value β_r and its time derivatives, $\dot{\beta}_r$ and $\ddot{\beta}_r$, are set to zero for oscillation elimination. Therefore, the diagonal gain matrices $K_{p\theta}$, $K_{I\theta}$, $K_{p\beta}$, and $K_{I\beta}$, which will assign stable poles to Equations 6.72 and 6.73, need to be defined. The poles chosen here are (-3, -4) for each nominal motion and (-1, -2) for every vibratory mode, so the corresponding gain matrices are $K_{p\theta} = \text{diag}[(7)_3]$, $K_{I\theta} = \text{diag}[(12)_3]$, $K_{p\beta} = \text{diag}[(3)_3]$, and $K_{I\beta} = \text{diag}[(2)_3]$. With the above parameters, the error states e_θ and e_β are evaluated from Equations 6.72 and 6.73 which are

$$e_\theta = (\dot{\theta} - \dot{\theta}_r) + K_{p\theta}(\theta - \theta_r) + K_{I\theta}\int(\theta - \theta_r)dt$$

$$e_\beta = (\dot{\beta} - \dot{\beta}_r) + K_{p\beta}(\beta - \beta_r) + K_{I\beta}\int(\beta - \beta_r)dt$$

where θ is a $\mathcal{R}^{n_\theta}$ vector of nominal joint displacements and β is a $\mathcal{R}^{n_\beta}$ vector of vibratory mode amplitudes.

6.4.3 Step 3: Construction of Control Input

The control input u is the conglomerate of Equations 6.78, 6.79, 6.94, and 6.105 as cited below

$$\begin{aligned}
u &= u_1 + u_2 \\
u_1 &= [\Sigma_1\ \Lambda^T]_k \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (f_1)_k + [\hat{\Sigma}_1\ \hat{\Lambda}^T]_u \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (\hat{f}_1)_u \\
u_2 &= u_3 - \frac{1}{2}[\dot{\Sigma}_1\ \dot{\Lambda}^T]_k \mathbf{e} - \frac{1}{2}[\hat{\dot{\Sigma}}_1\ \hat{\dot{\Lambda}}^T]_u \mathbf{e} \\
u_3 &= -\frac{\gamma_1}{e_\theta^T e_\theta} e_\theta
\end{aligned}$$

where γ_1 is defined in Equation 6.103 in which $\alpha_1 = \rho$ is used in the simulation. In the above computation of u_3, a safeguard is added to avoid dividing by zero, that is, when the denominator $e_\theta^T e_\theta$ reaches a predefined range close to zero, u_3 is set to zero. This precaution is to prevent numerical errors but might affect the resultant $\dot{V}$ value which according to the Lyapunov's second method needs to be negative to guarantee the convergence of both controlled motion and payload estimation. Yet, since $e_\theta \to 0$ means nominal motion approaches desired state, and the convergence of parameter estimation, which is approximate to the result of least square method after choosing a small ρ, relies primarily on the condition of persistent excitation. Therefore, the major impact of setting u_3 to zero is on vibration elimination. However, this situation generally occurs at the end of nominal motion, hence the residual oscillation could be left to be dissipated passively by structural damping. A modification of the above approach is updating u_3 by Equation 6.111, which could have a nonzero divisor by resetting u_3, but this will create a disturbance on nominal motion, therefore, it is not used in simulations. After constructing control input u, motion response is found numerically from system dynamics defined in Equation 6.67.

6.4.4 Step 4: Update of Parameter Estimation

For constant payload inertia $\mathbf{a}$, $\dot{\hat{\mathbf{a}}} = \ddot{\hat{\mathbf{a}}}$ where $\ddot{\hat{\mathbf{a}}}$ is given by Equation 6.98, then the update equation of estimate $\hat{\mathbf{a}}$ is

$$\dot{\hat{\mathbf{a}}} = R(\frac{1}{2}\rho Z_2^T - \rho Z_1^T \mathbf{e} - Z_0^T \varepsilon)$$

in which Z_2 is defined in Equation 6.89, Z_1 is introduced in Equation 6.80, Z_0 is the integral of Z_0' shown in Equation 6.84, and ε is presented in Equation 6.85 where U is the integral of U' defined in Equation 6.84. According

to the previous discussion, since a smaller ρ produces better estimate convergence, $\rho = 0.001$ is chosen in simulation. Additionally, in the simulation equal weight is placed on all data used in constructing R^{-1}, i.e., no exponential forgetting in estimation is introduced ($\sigma = 0$), then from Equation 6.48, R is updated by

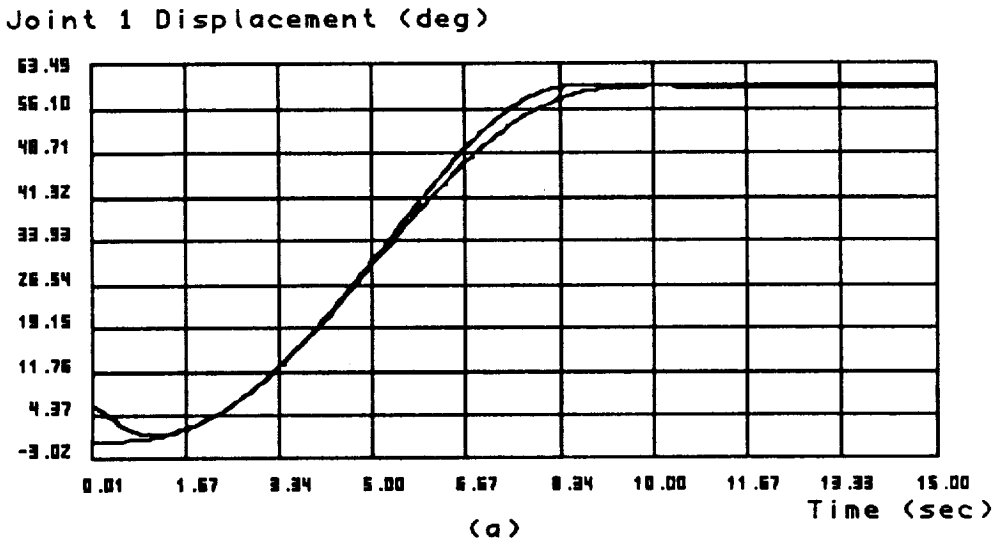
$$\dot{R} = -RZ_0^T Z_0 R$$

The reason of choosing $\sigma = 0$ is that if system ceases excitation during simulation, according to Equation 6.57, a nonzero σ will turn R^{-1} into a semidefinite matrix quickly and hence cause R to increase drastically. Eventually, a singular R^{-1} makes R go to infinity and destroys the estimation process. In general practice, estimation is terminated once persistent excitation stops, so an exponential forgetting factor could be employed to speed up convergence, but that is not implemented in our case studies in order to observe the kinematic effect on payload excitation. Therefore, $\sigma = 0$ is used here to put equal weight on all collected data. The initial value $R(0)$ used in the simulation is a 10×10 diagonal matrix with ten repeated diagonal elements of value 1000. The integration step in the following simulations is 1 msec.

6.4.5 Simulation Results on a 3-Link Manipulator Model

The first model simulation results are reported in the following figures. Figures 6.4 (a), (b), and (c) are the traces of nominal joint displacements. Notice that the desired and actual displacements are separated by an initial position error. Figures 6.4 (d), (e), and (f) are link deformations, where the final steady state deformations in (d) and (e) are due to gravitation force. Figures 6.4 (g), (h), and (i) are control voltages of joint 1 to 3. Figures 6.4 (j) to (s) are plots of exact and estimate of payload inertia, and Figure 6.4

(t) shows the sum of the diagonal element, or trace, of R, as an examination of persistent excitation. As discussed before, for $\sigma = 0$, R^{-1} grows rapidly under persistent excitation, hence R and its trace diminishes swiftly and ideally becomes zero. However, the reduction of the trace of R in Figure 6.4 (t) stops at 706.7, hence the system is not properly excited in this case, therefore estimation results are impaired. In Figures 6.4 (j) to (m), estimates of mass and center of mass location approach actual values, but estimates of moment of inertia in Figures 6.4 (n) to (s) fail to reach exact values, which means that in the controlled motion payload is insufficiently activated to reflect its true value. Since the simulated model is a three-link manipulator without wrist, the gripper has limited angular movement to probe the moment of inertia of payload, therefore, little information could be collected to reconstruct the true identities. In the next case, a six DOF manipulator is simulated, which will show improved estimation of moment of inertia with additional angular movement of the wrist.



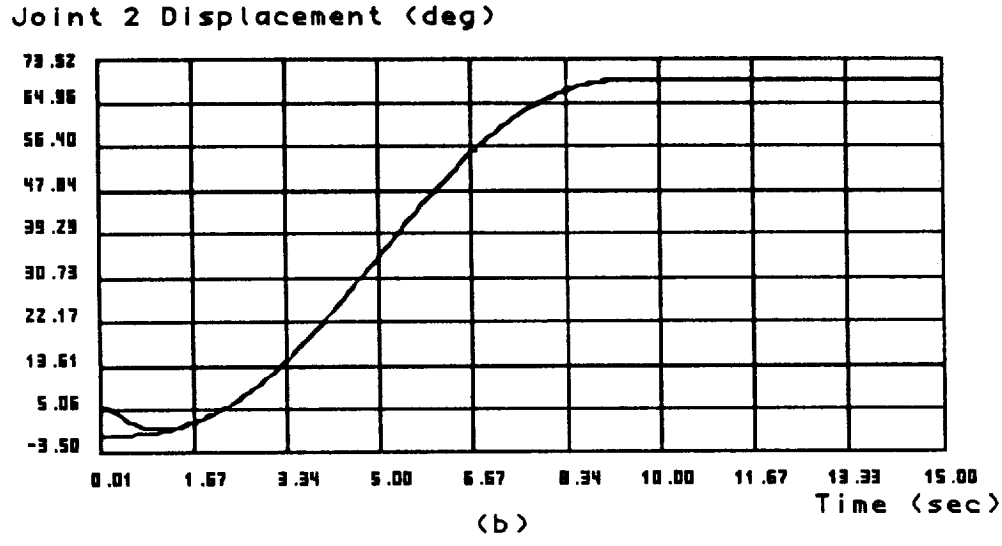
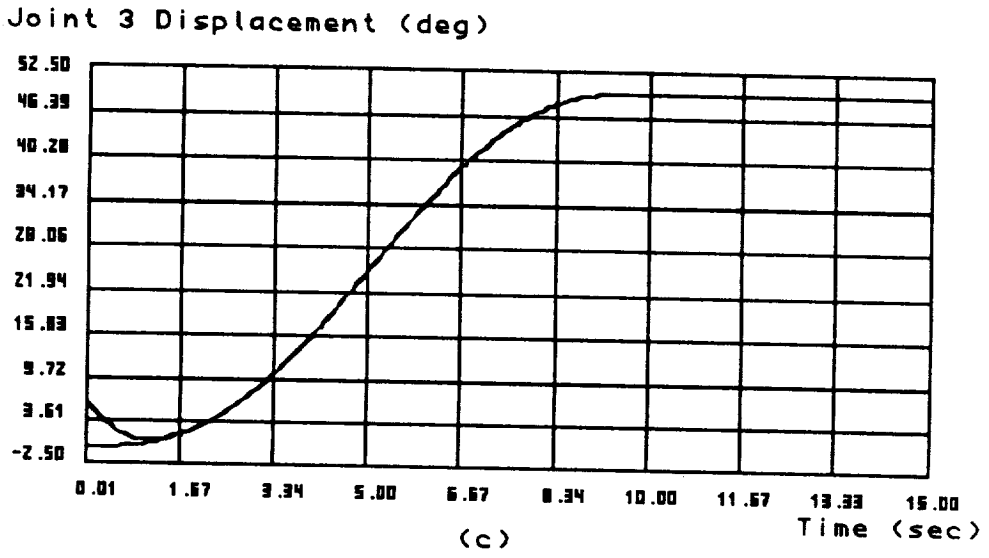


Figure 6.4: Adaptive Control of a 3-Link Robot



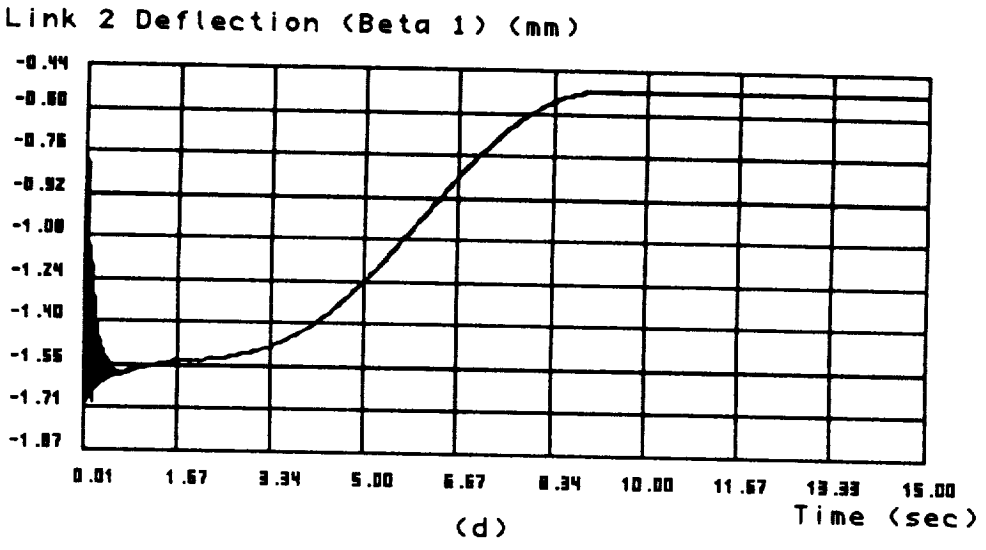
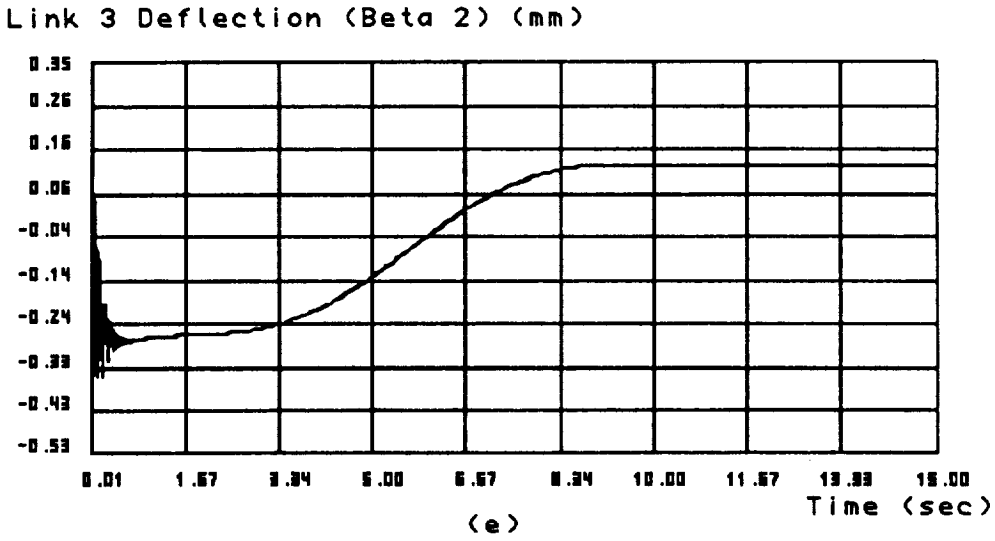


Figure 6.4: Adaptive Control of a 3-Link Robot



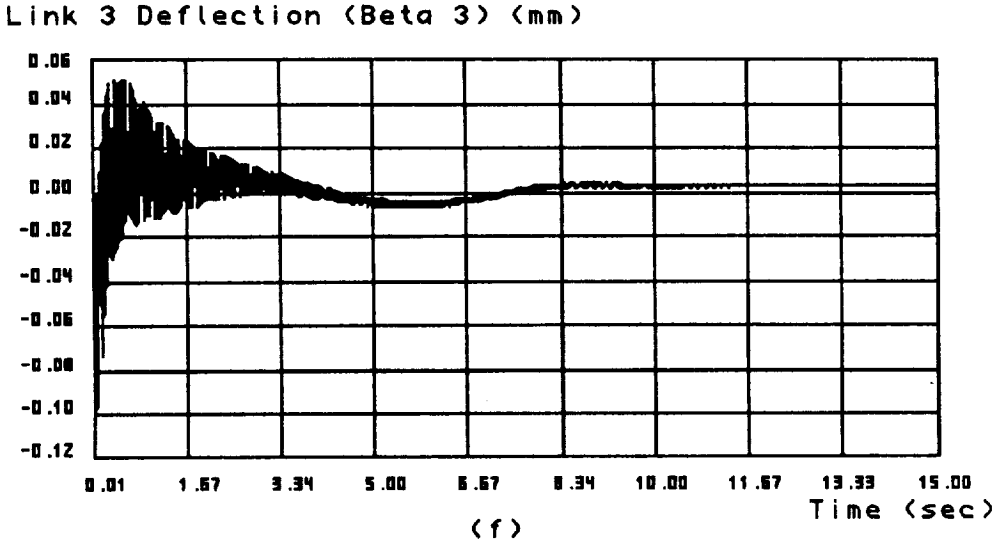
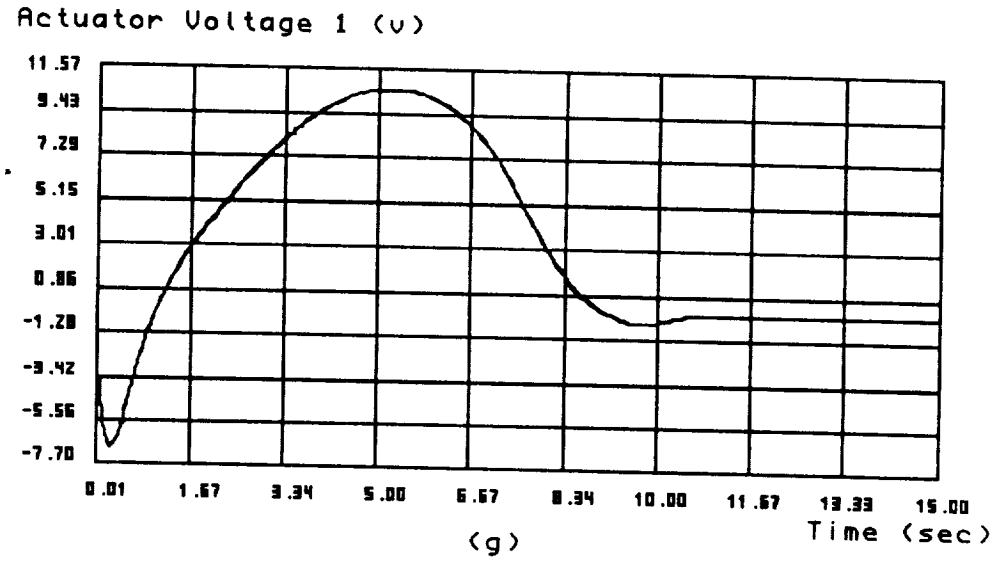


Figure 6.4: Adaptive Control of a 3-Link Robot



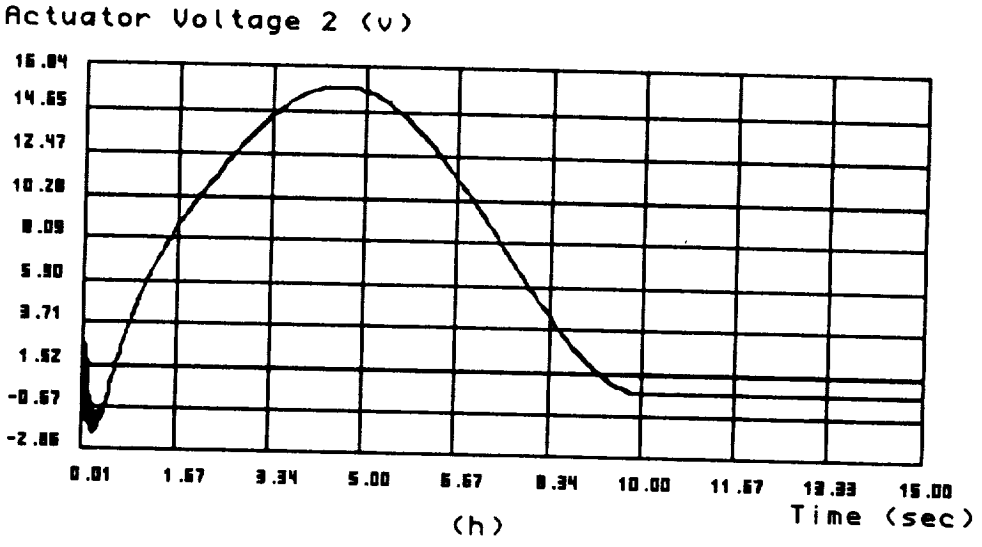
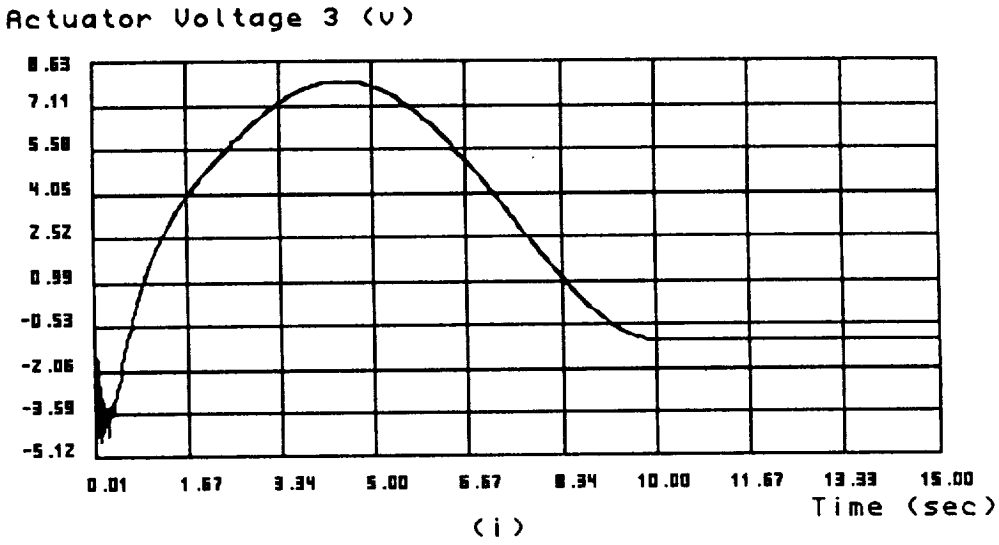


Figure 6.4: Adaptive Control of a 3-Link Robot



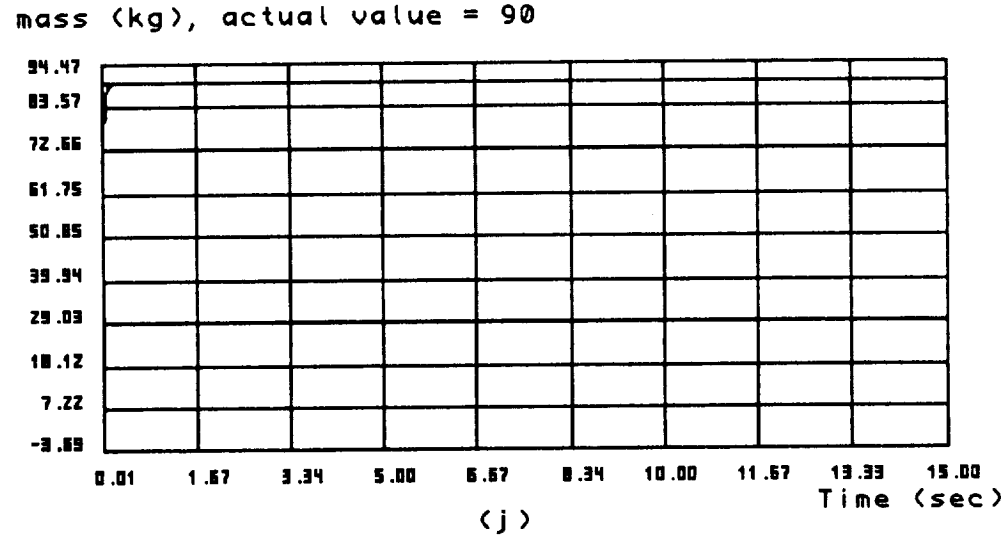
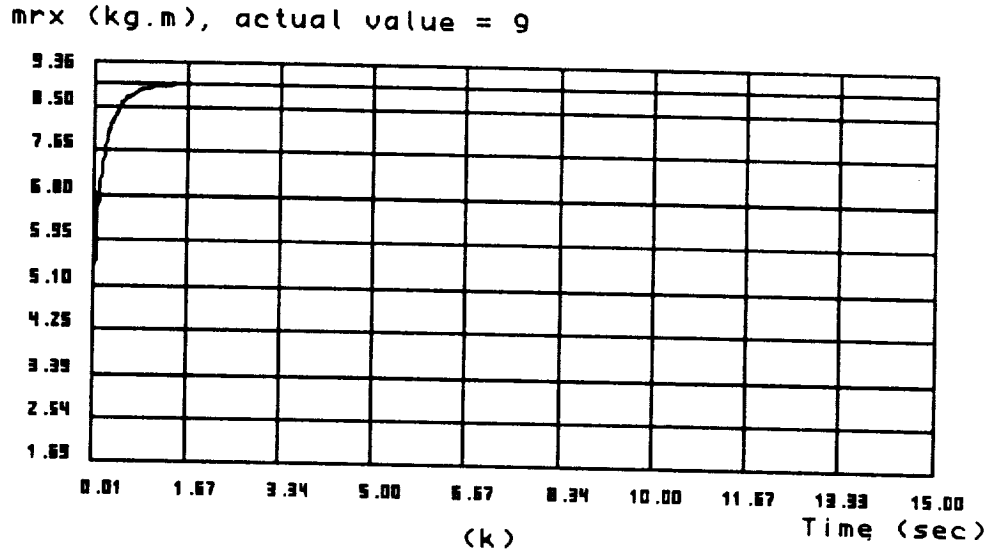


Figure 6.4: Adaptive Control of a 3-Link Robot



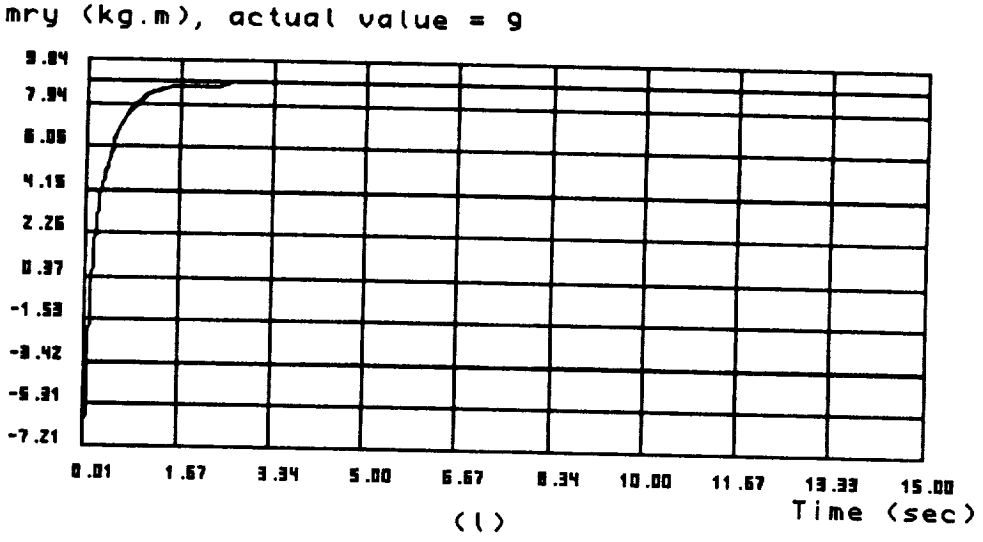
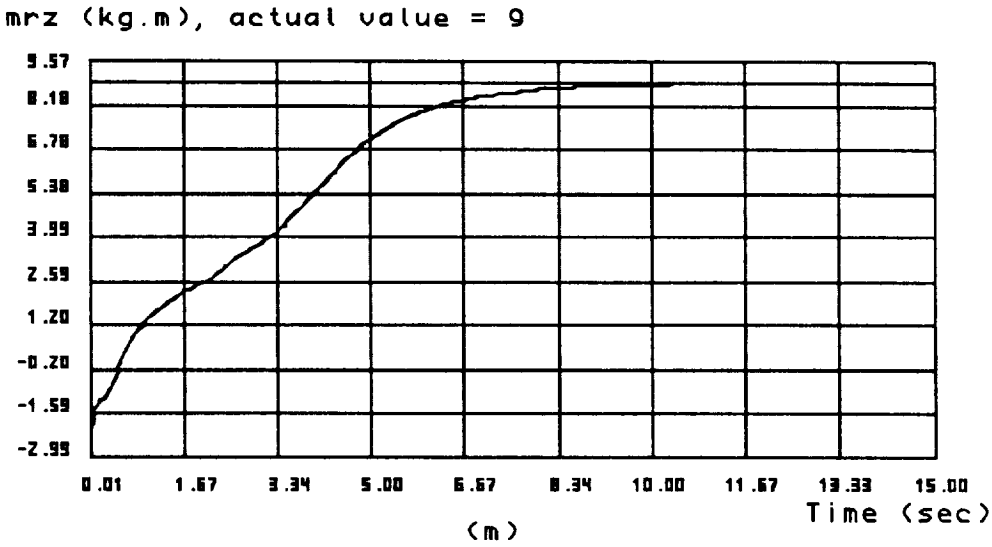


Figure 6.4: Adaptive Control of a 3-Link Robot



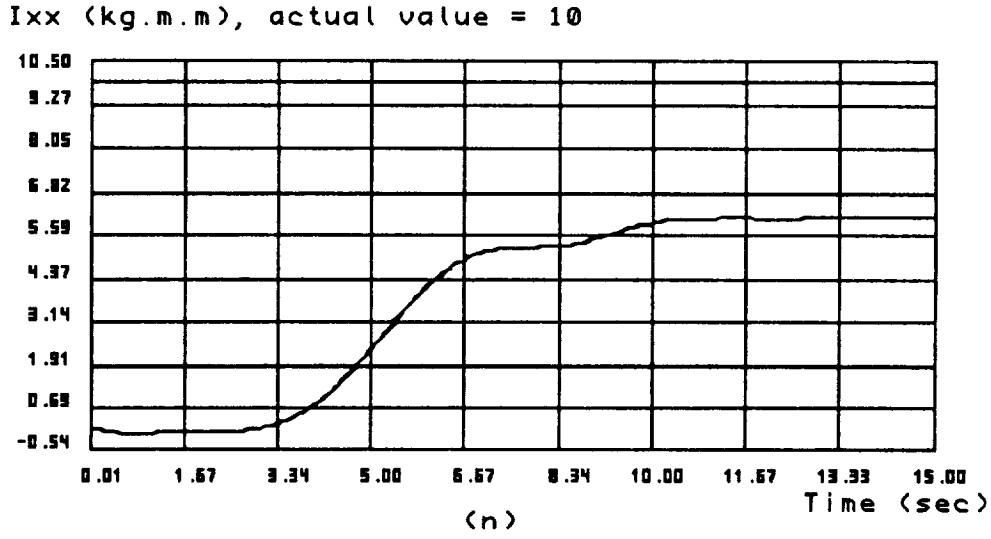
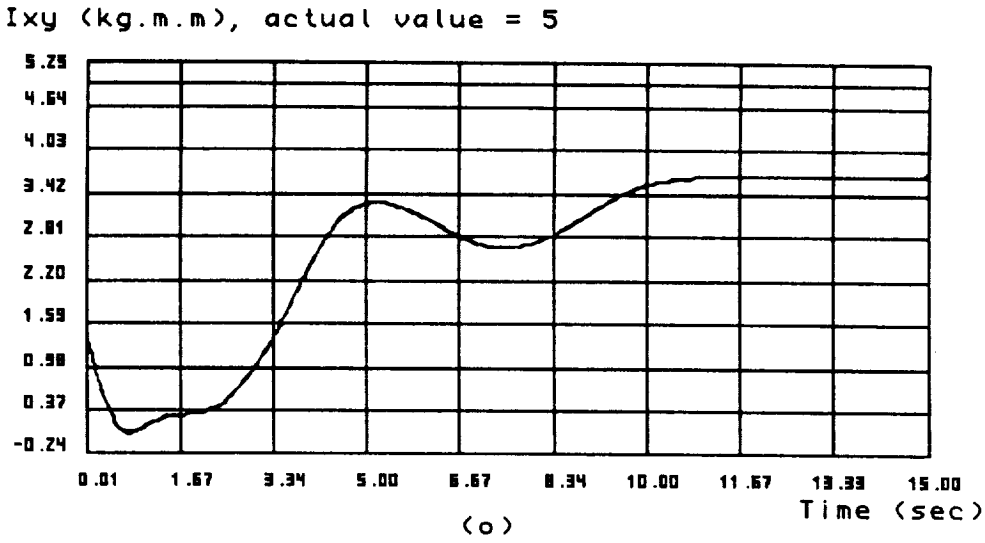


Figure 6.4: Adaptive Control of a 3-Link Robot



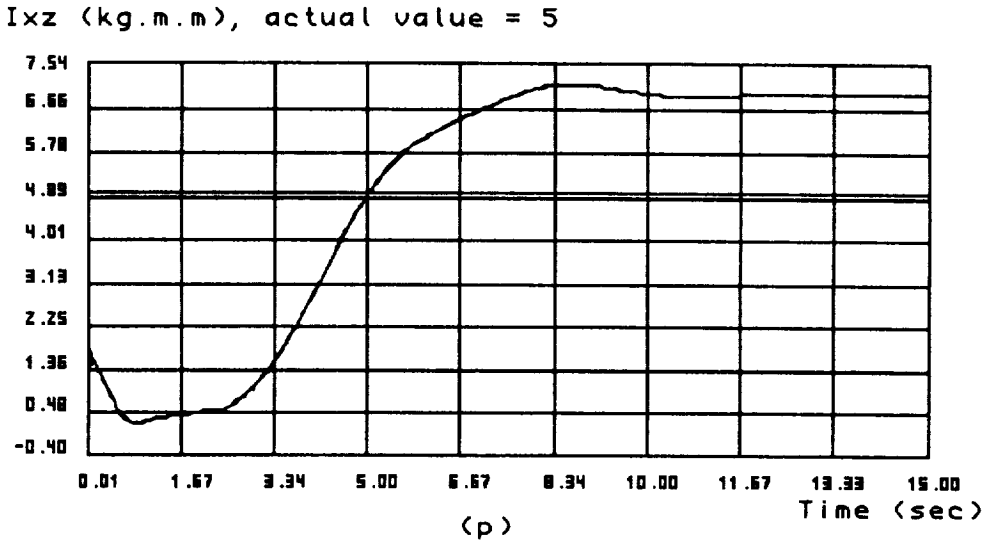
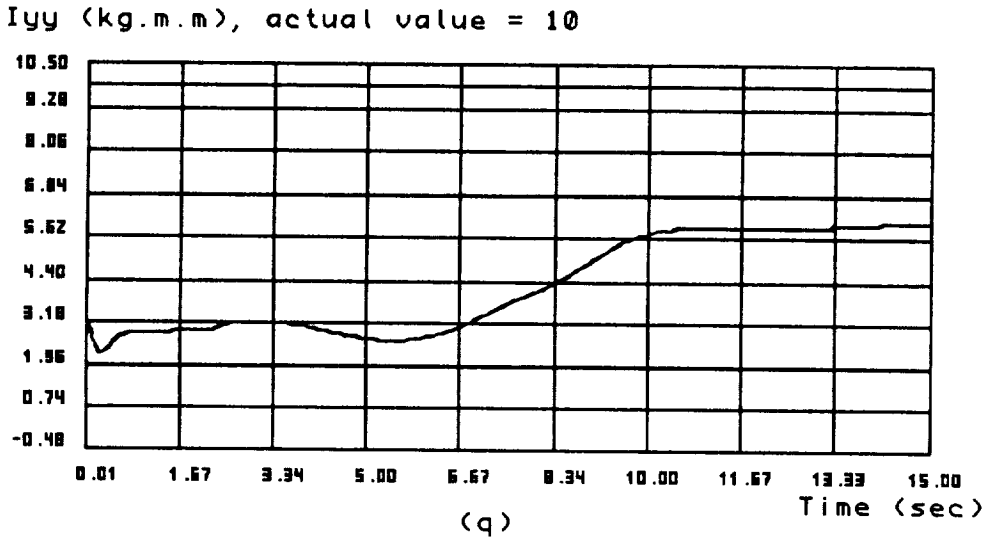


Figure 6.4: Adaptive Control of a 3-Link Robot



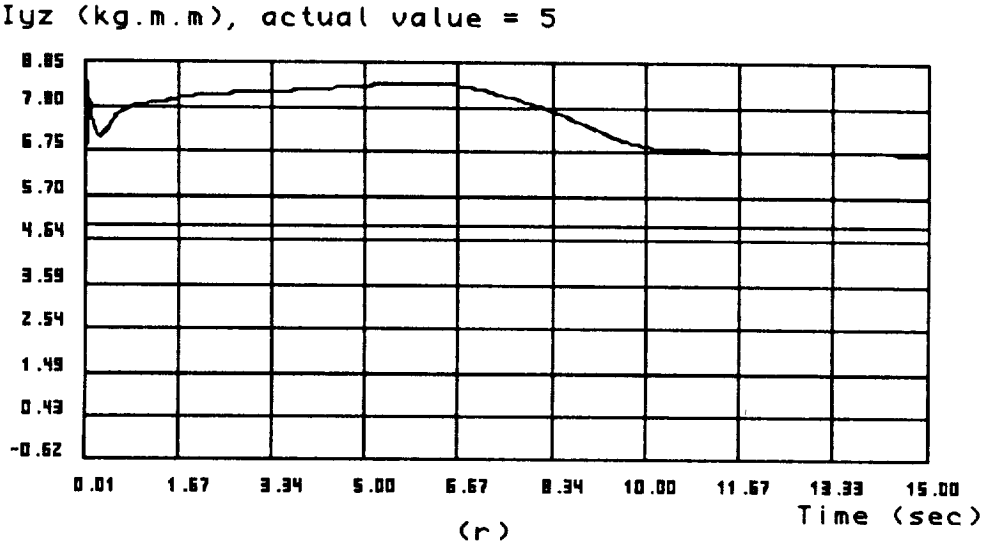
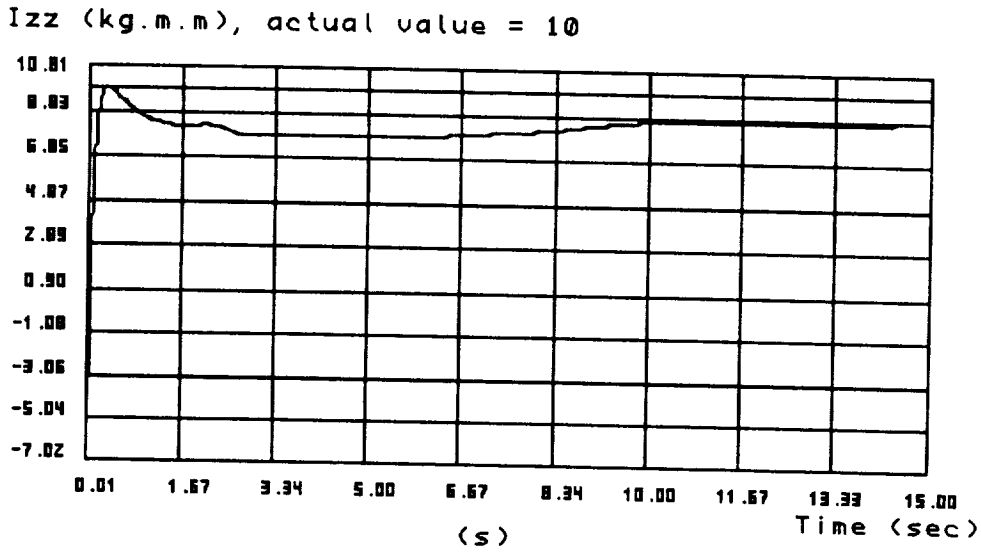


Figure 6.4: Adaptive Control of a 3-Link Robot



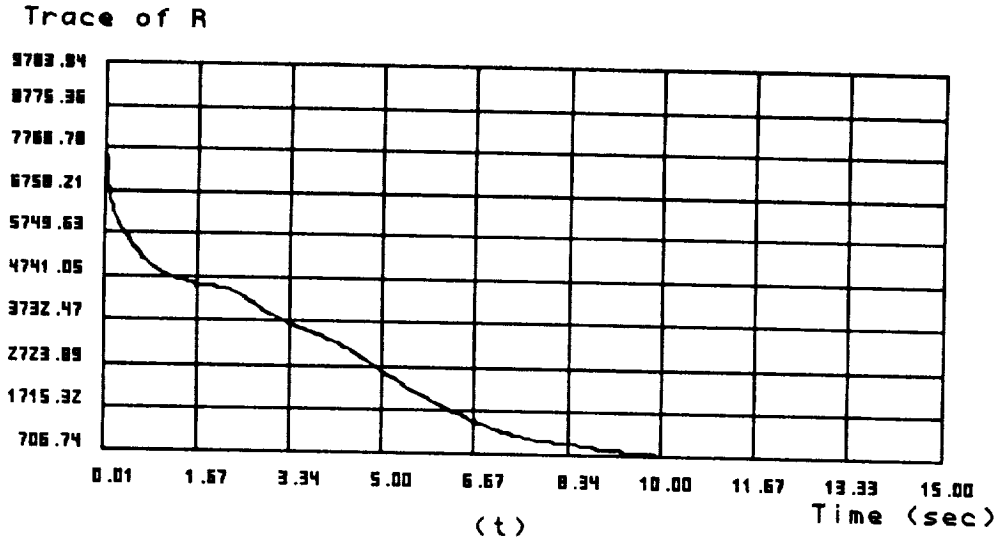


Figure 6.4: Adaptive Control of a 3-Link Robot

6.4.6 Case 2: Simulation on a 6-Link Manipulator Model

The second case simulates the motion control of a six DOF manipulator modeled with three lumped link compliances as depicted in Figure 6.5. The parameters of this model are listed in the following tables.

Table 6.7 6-Link Model Kinematic Parameters			
	Link Length (m) (x, y, z)	Center of Mass (m) (x, y, z)	Offset Angle (deg)
Link 1	(0, 0, 0.8128)	(0, 0, -0.4318)	0
Link 2	(1.1776, 0, 0)	(0.508, -0.0254, 0)	90
Link 3	(0, 0, 0)	(0.1016, -0.1778, 0)	0
Link 4	(0, 0, 1.397)	(0, 0, -0.508)	90
Link 5	(0, 0, 0)	(0, 0, 0)	-60
Link 6	(0, 0, 0.1524)	(0, 0, -0.1016)	60

Table 6.8 6-Link Model Inertial Parameters		
	Mass (kg)	Moment of Inertia ($kg.m^2$) (I_{xx}, I_{yy}, I_{zz})
Link 1	317.5	(0, 0, 29.3)
Link 2	680.4	(5.9, 52.7, 43.9)
Link 3	453.6	(49.7, 7.61, 49.7)
Link 4	68	(0.59, 0.59, 0.35)
Link 5	36.3	(0.23, 0.23, 0.06)
Link 6	27.2	(0.12, 0.12, 0.06)

Table 6.9 6-Link Model Actuator Parameters						
Joint	1	2	3	4	5	6
Inertia ($10^{-3} \cdot kg \cdot m^2$)	4.2	2.1	2.1	1.3	1.3	0.8
Damping ($N \cdot m/(rad/s)$)	0.4	0.4	0.3	0.4	0.3	0.3
Resistance (ohm)	0.8	0.8	0.8	0.8	0.8	0.8
Torque Constant ($volt/(rad/s)$)	20	20	14	11	8	8
Gear ratio	100	100	100	80	30	10
Back emf Const. ($N \cdot m/amp$)	0.5	0.5	0.4	0.3	0.3	0.2

and the gain matrix are: $K_{p\theta} = \text{diag}[(7)_6]$, $K_{I\theta} = \text{diag}[(12)_6]$, $K_{p\beta} = \text{diag}[(3)_3]$, and $K_{I\beta} = \text{diag}[(2)_3]$. $R(0)$, α, and stiffness matrix are the same as the last simulation, and $\sigma = 0$. The nominal trajectories and initial states are given in the next table.

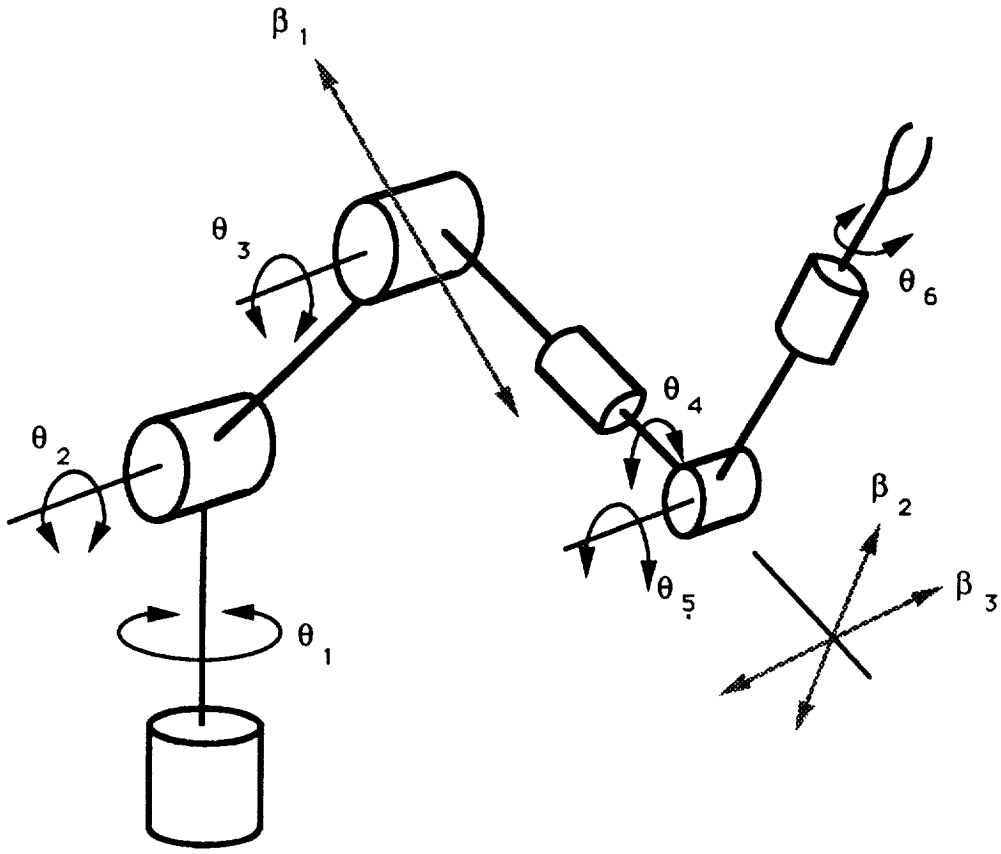


Figure 6.5: Six-Link Robot Modeled with Three Link Compliances

Table 6.10 6-Link Model Trajectory Parameters					
	$t_f(sec)$	$\Delta\theta(deg)$	θ_{r0}	θ_0	$\theta_0 - \theta_{r0}$
Joint 1	10	60	0	10	10
joint 2	10	70	0	10	10
Joint 3	10	50	0	10	10
Joint 4	10	40	0	10	10
Joint 5	10	50	0	10	10
Joint 6	10	60	0	10	10

Two sets of initial estimates are tested on this model. The first set has zero initial estimates as shown in the next table.

Table 6.11 Actual Payload parameters and Initial Estimates										
	m	mr_x	mr_y	mr_z	I_{pxx}	I_{pxy}	I_{pxz}	I_{pyy}	I_{pyz}	I_{pzz}
$a_{1\to10}$	90	9	9	9	10	5	5	10	5	10
$\hat{a}_{1\to10}(t=0)$	0	0	0	0	0	0	0	0	0	0

The second set has overestimated initial values as listed in the following table.

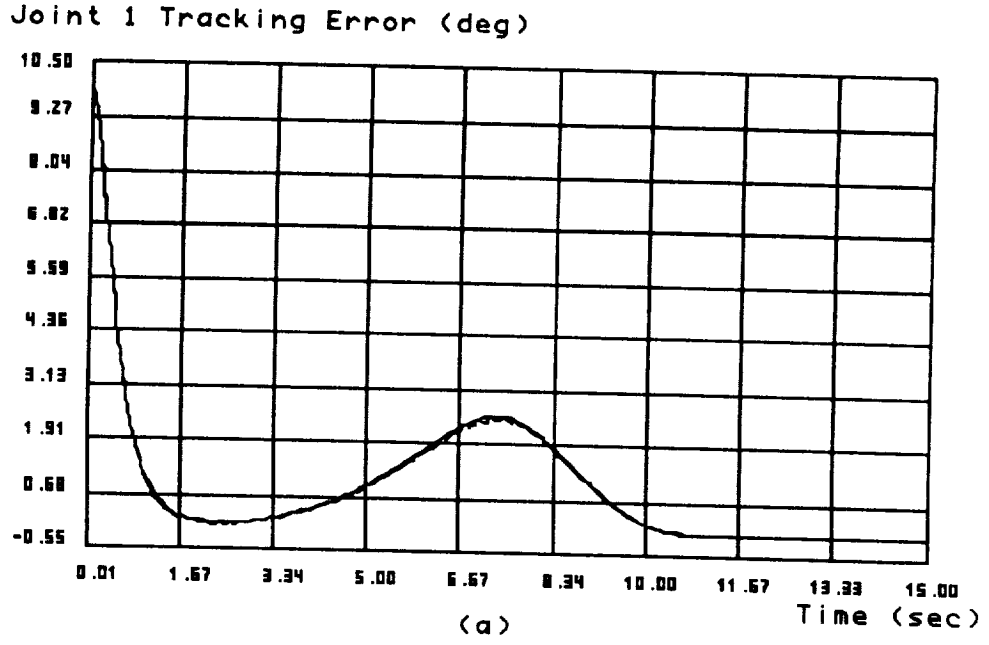
Table 6.12 Actual Payload parameters and Initial Estimates										
	m	mr_x	mr_y	mr_z	I_{pxx}	I_{pxy}	I_{pxz}	I_{pyy}	I_{pyz}	I_{pzz}
$a_{1\to10}$	90	9	9	9	10	5	5	10	5	10
$\hat{a}_{1\to10}(t=0)$	120	18	18	18	15	8	8	15	8	15

The purpose of using two sets of data is to test the controller's adaptive capacity to different initial estimates. Besides that a comparison of both results will give a better understanding on the phenomenon of persistent excitation and how estimation affects controller performance. Both simulations follow the same steps listed in the three-link case and the results are presented in Figures 6.6 (a) to (z). In order to produce a compact presentation, each figure carries two sets of data where a solid line represents response of zero initial case and a broken line indicates performance of overestimated case. Figures 6.6 (a) to (f) show nominal joints 1 to 6 tracking errors, i.e., $\theta - \theta_r$. In these figures, the initial positional errors are deliberately arranged according to Table 6.10 to test controller tracking capability. Figures 6.6 (g) to (i)

are vibratory mode deflections. The input voltages are displayed in Figures 6.6 (j) to (o). Figures 6.6 (p) to (y) are estimate errors, $\tilde{\mathbf{a}}$, of ten payload inertial parameters, and Figure 6.6 (z) is the trace of R which is treated as an indicator of persistent excitation. From the nominal and vibratory motions in Figures 6.6 (a) to (i), controller stability is verified. Additionally, estimate convergence is revealed from Figures 6.6 (p) to (y) where estimate errors approach zero in the first 1.67 sec. Also, the trace of R reduces form 10,000 to less than 32 for zero initial case and 220 for overestimated case at the same period of time. Both traces of R keep decreasing until reaching a value 4 for zero initial case and 16 for overestimated case. As stated before, for a persistently exciting system, R will become zero eventually. Therefore, in the six-link model, the controlled system is persistently exciting, which is essential to the convergence of the payload estimates. Such results are in contrast to that of the third-link case where the trace of R stops at 707 and payload estimates fail to converge, especially the moment of inertia values. Apparently, the wrist in the six-link model generates additional angular motion crucial to the excitation of the moment of inertia contents. Such connections provide us an insight of the relationship between system physical motion and excitation conditions required by analytical estimation. Despite the satisfactory performance of both zero initial and overestimated cases, some observations regarding simulation results will be discussed below.

For the first three joints, both cases show almost identical response as displayed in Figures 6.6 (a) to (c), but the difference shows up in the wrist motion according to Figures 6.6 (d) to (f) where the zero initial estimate case suffers a large overshoot especially within the first two seconds. Similar distinctions also appear in the actuator control voltages shown by Figures 6.6 (m) and (o). Such results are due to a substantial payload inconsistency be-

tween the model and plant. By comparing Tables 6.8 and 6.11, the mass and moment of inertia of links 4, 5, and 6 are far less than actual payload values, so payload dominates the dynamic effect of wrist motion. Furthermore, in the motion control payload is uncertain and estimated on-line, therefore wrist control becomes sensitive to estimate precision. According to Figures 6.6 (p) to (y) results, the first two seconds is the transient period of estimation in zero initial estimate case, depicted by solid line, yields large estimate errors. Inevitably, the wrist motion is affected in this transient period. Interestingly, the vigorous wrist overshoot of the zero initial case accidently creates better excitation to payload than that of the overestimated case, therefore, the former has a smaller and faster declining R value as shown in Figure 6.6 (z). By contrast, the inertia of the first three links are substantially larger than payload magnitude, therefore their motion controls are robust to imprecise estimates. However, both zero initial and overestimated cases have similar structural deflections, and the steady state deflections in Figures 6.6 (g) to (h) are due to the weight of the structure. Notice that in Figures 6.6 (p) to (s) mass and center of mass location estimates converge to actual values quickly, similar tendencies are also observed from three-link case results, i.e., Figures 6.4 (j) to (m). Obviously, payload mass is easy to excite and could be estimated precisely in few time steps.



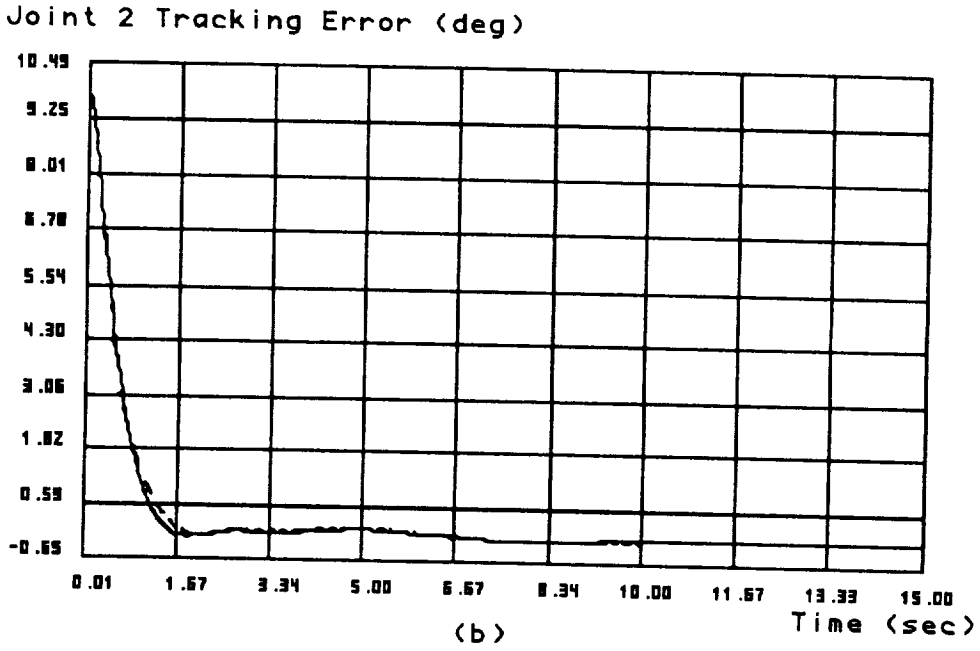
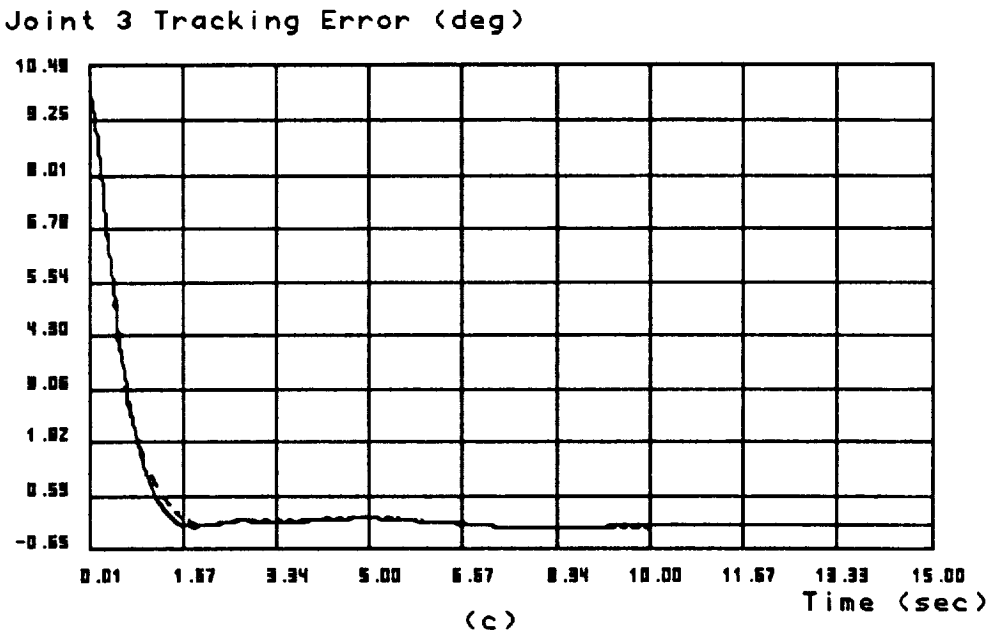


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



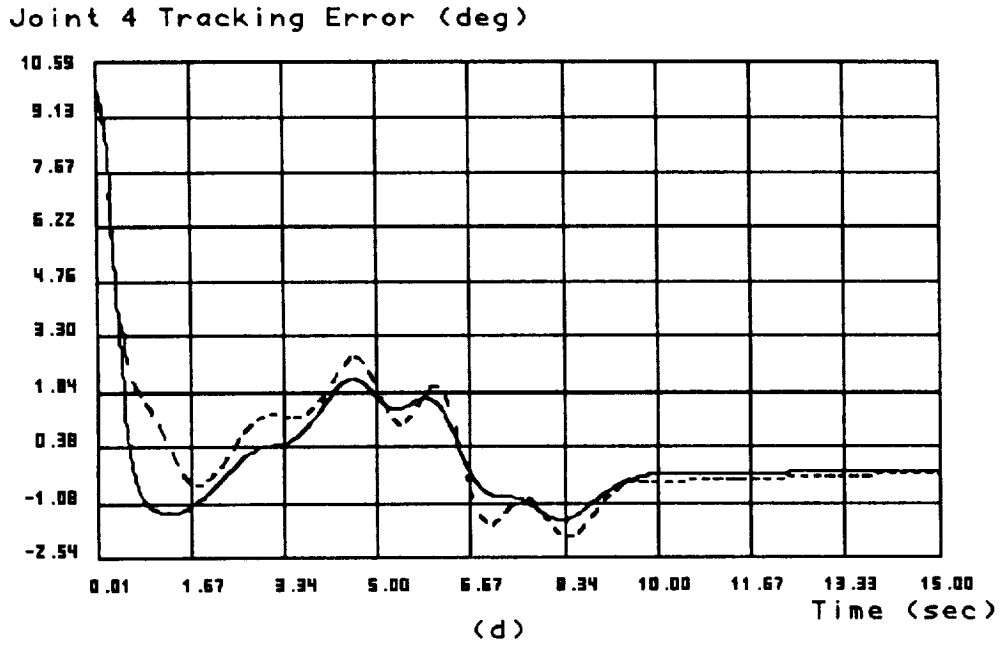
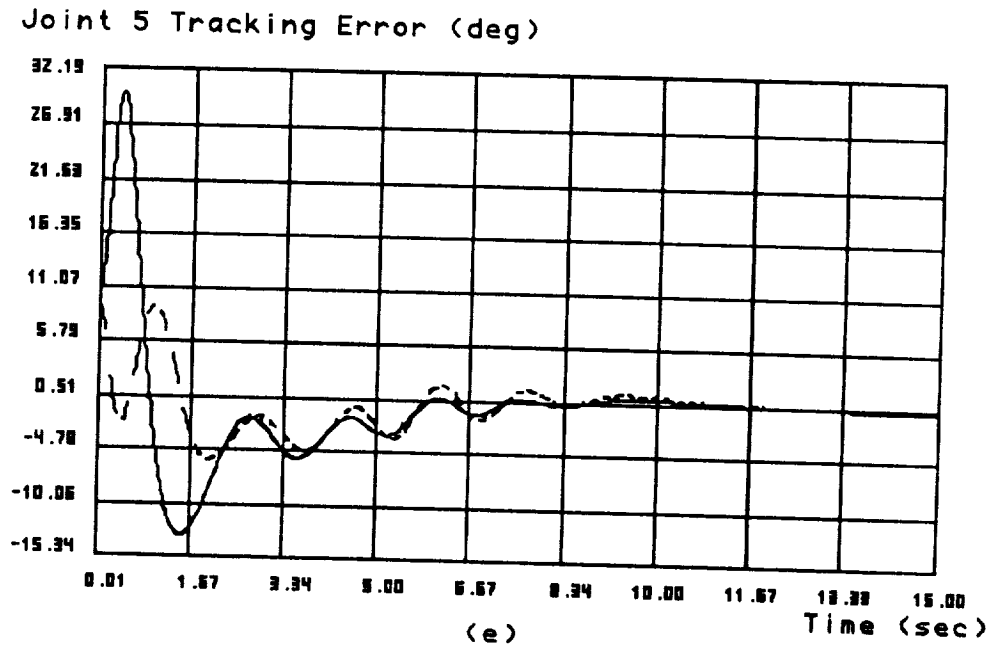


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



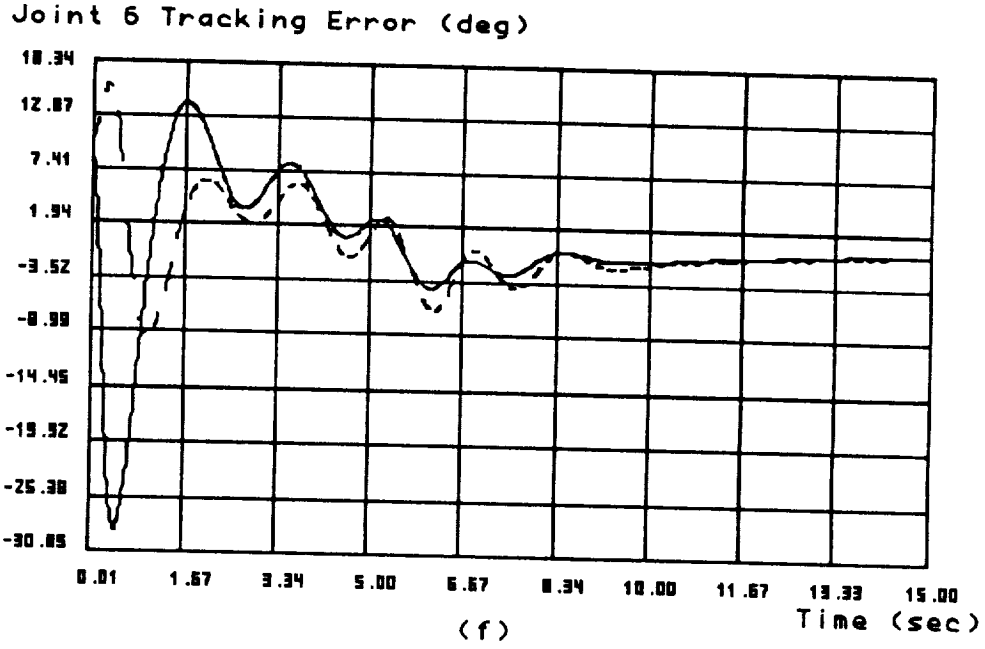
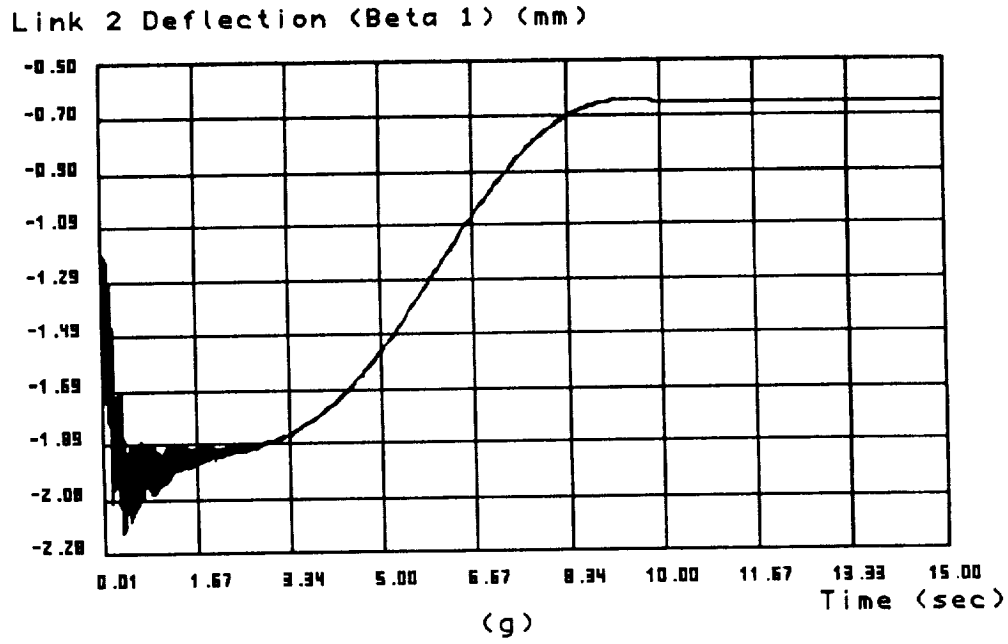


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



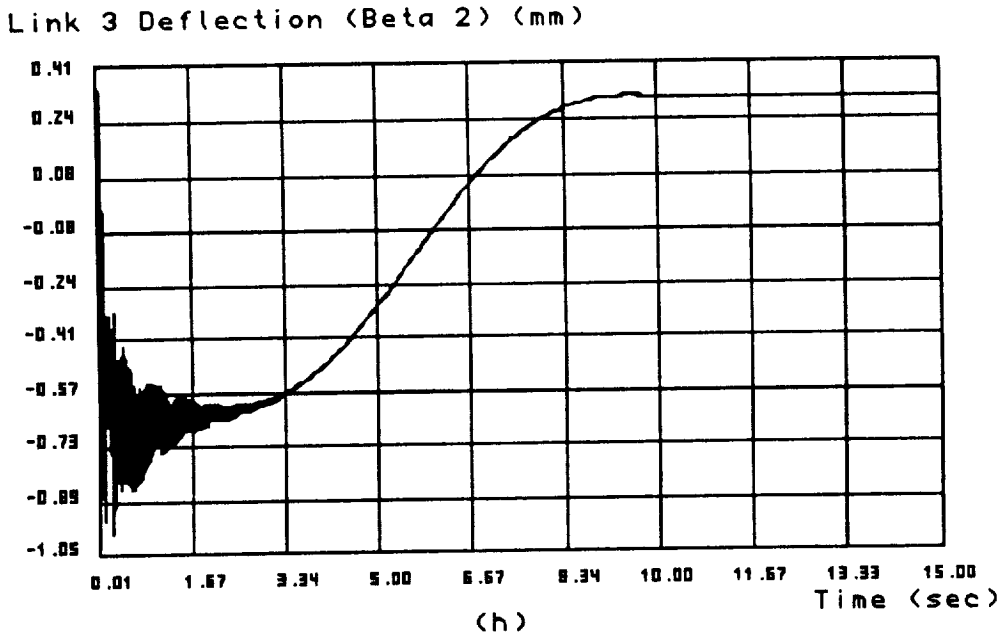
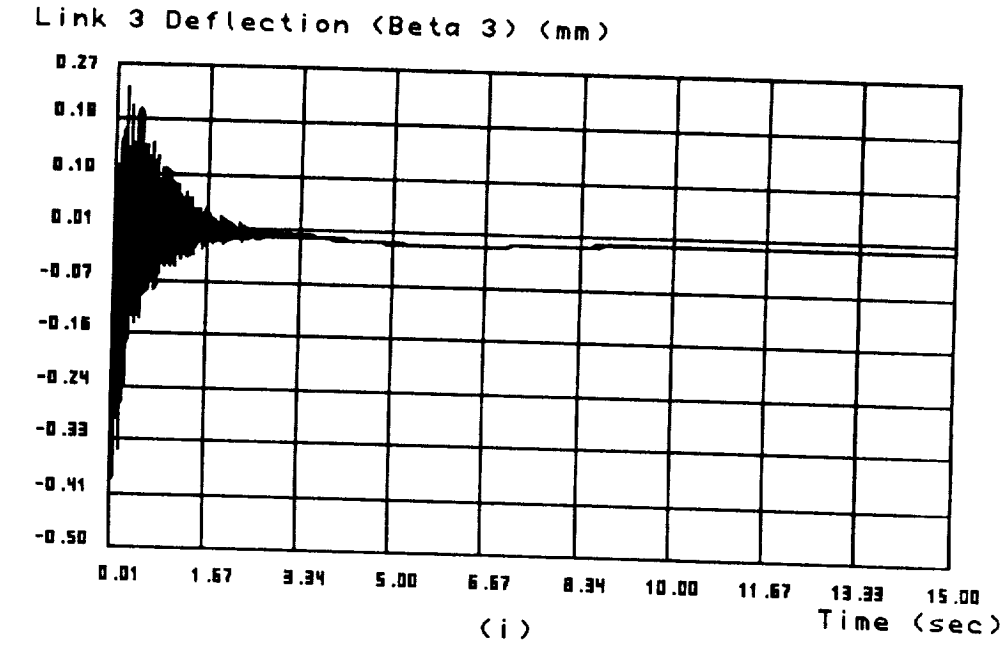


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



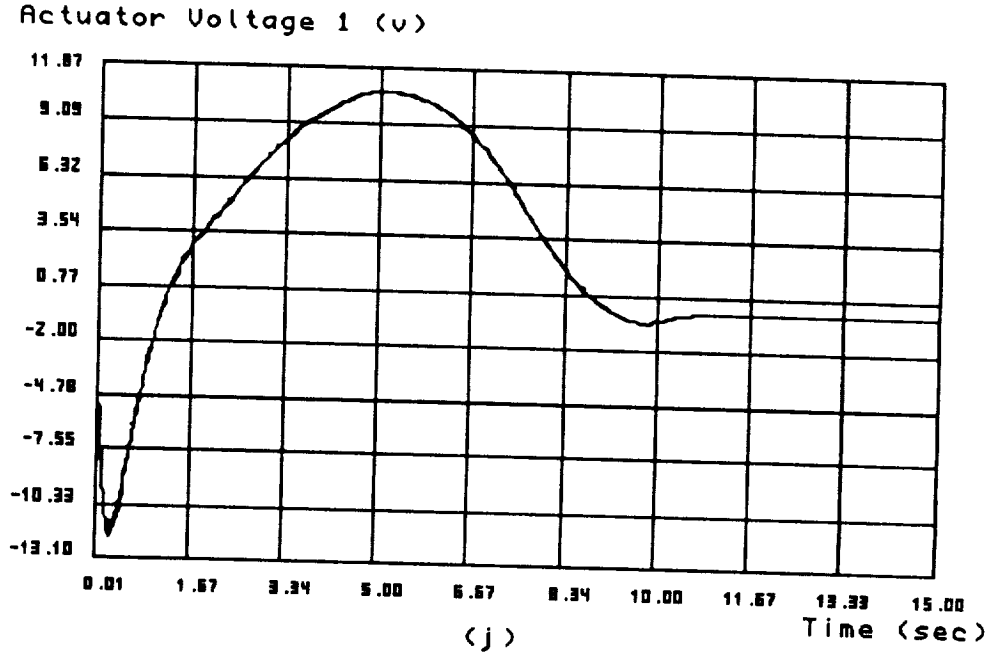
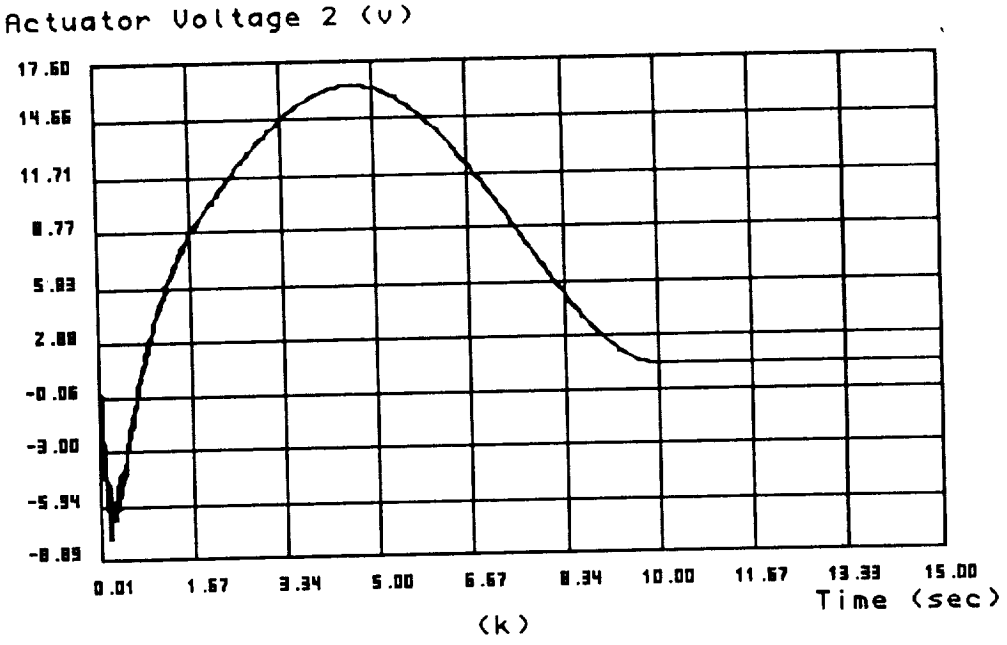


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



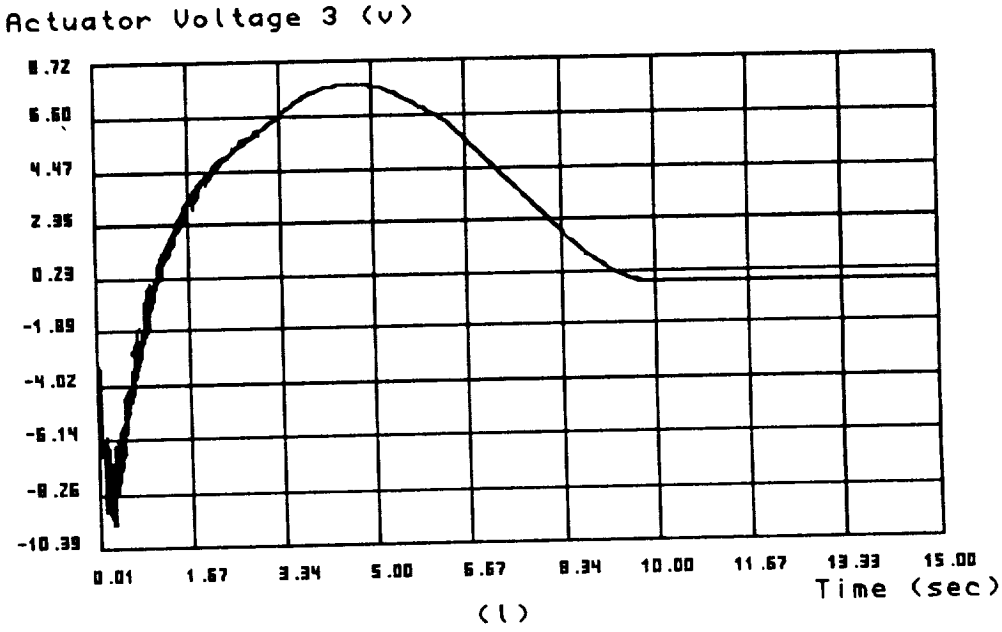


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

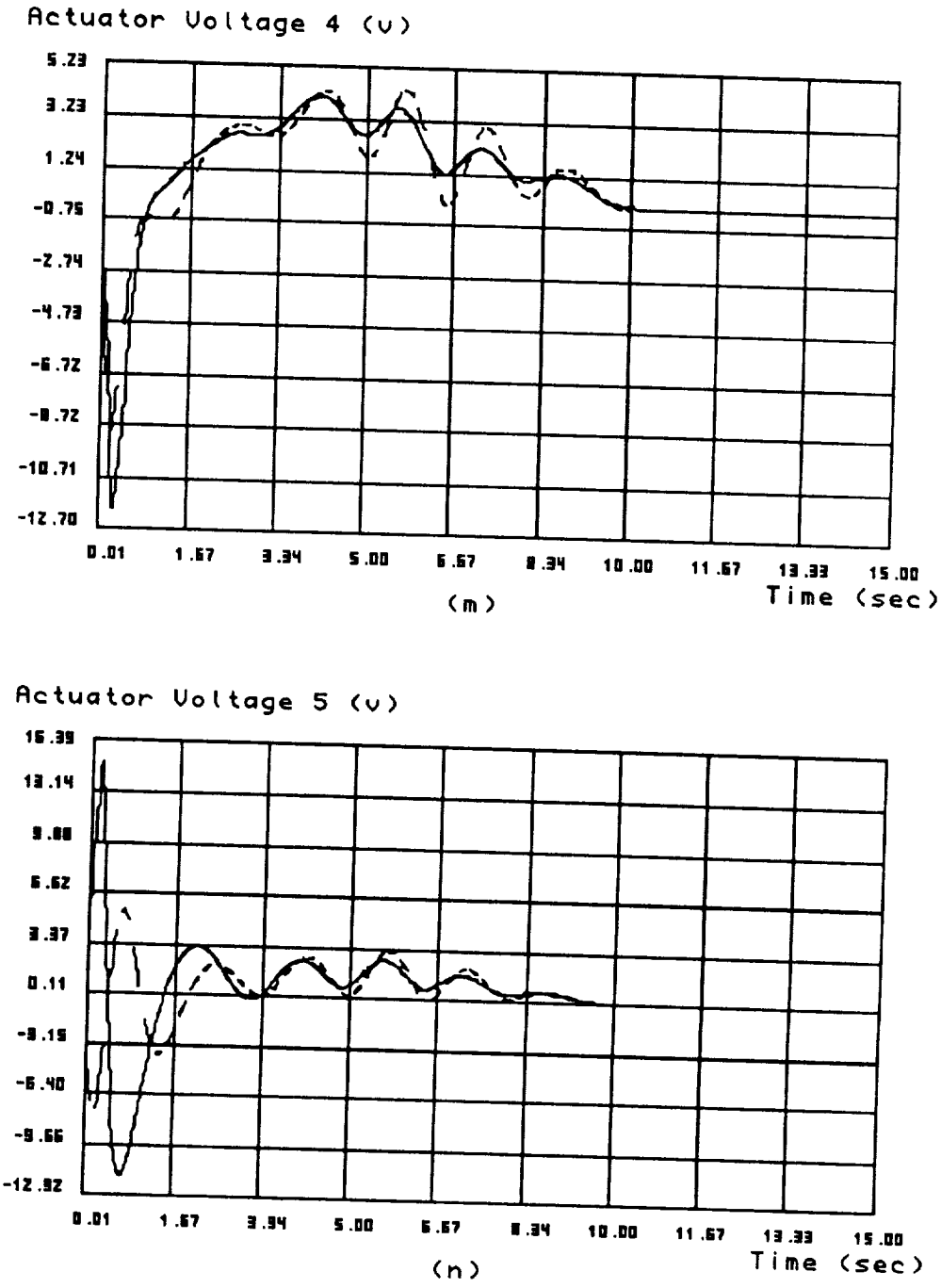
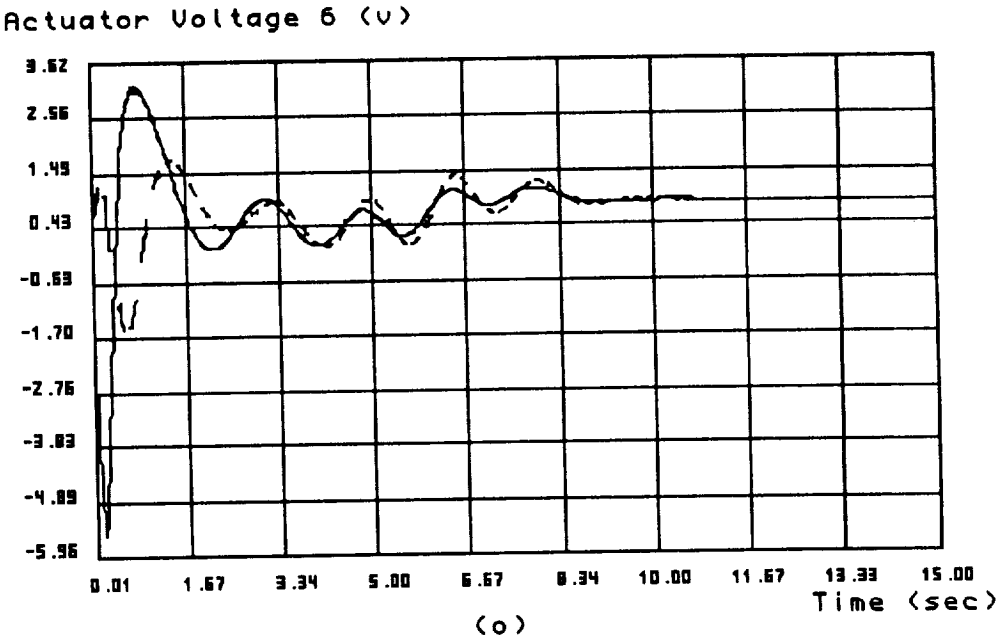


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



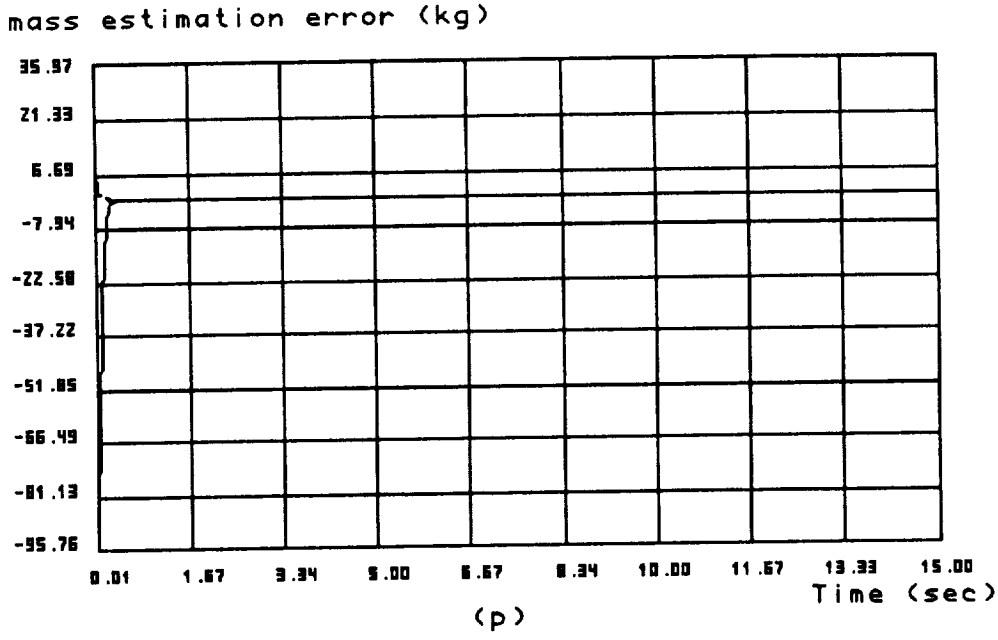


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

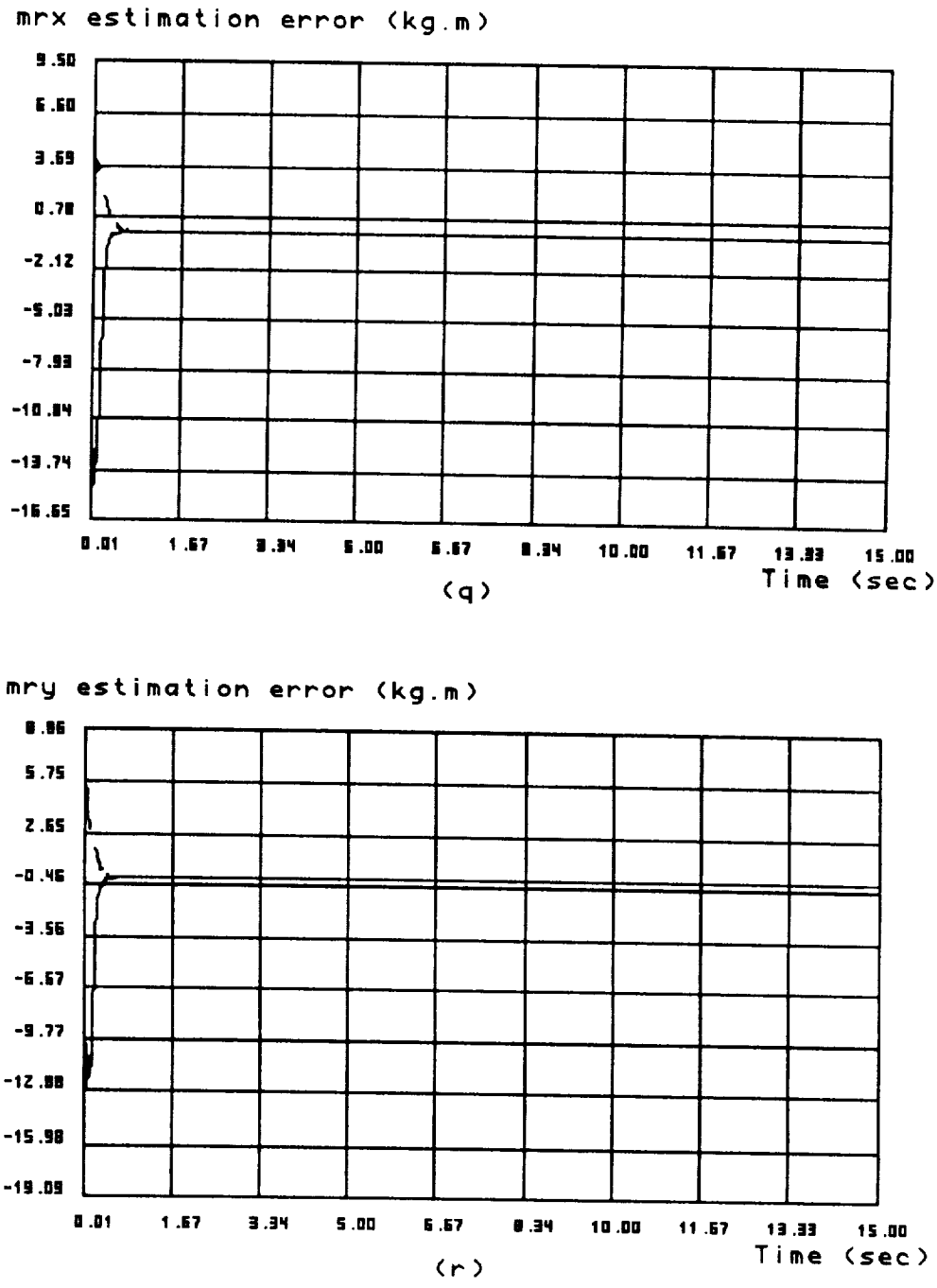
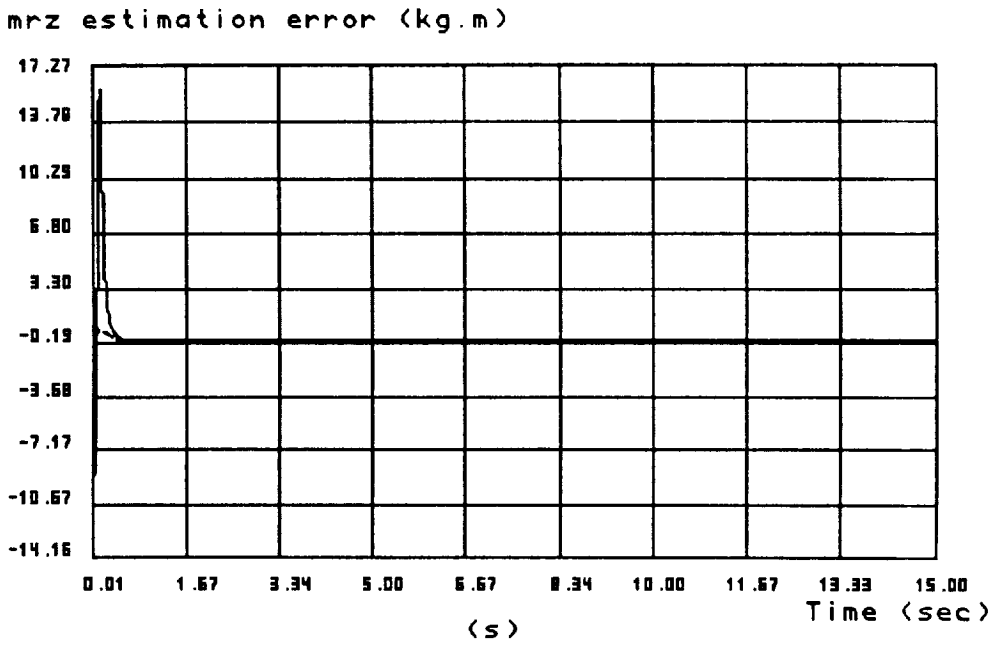


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



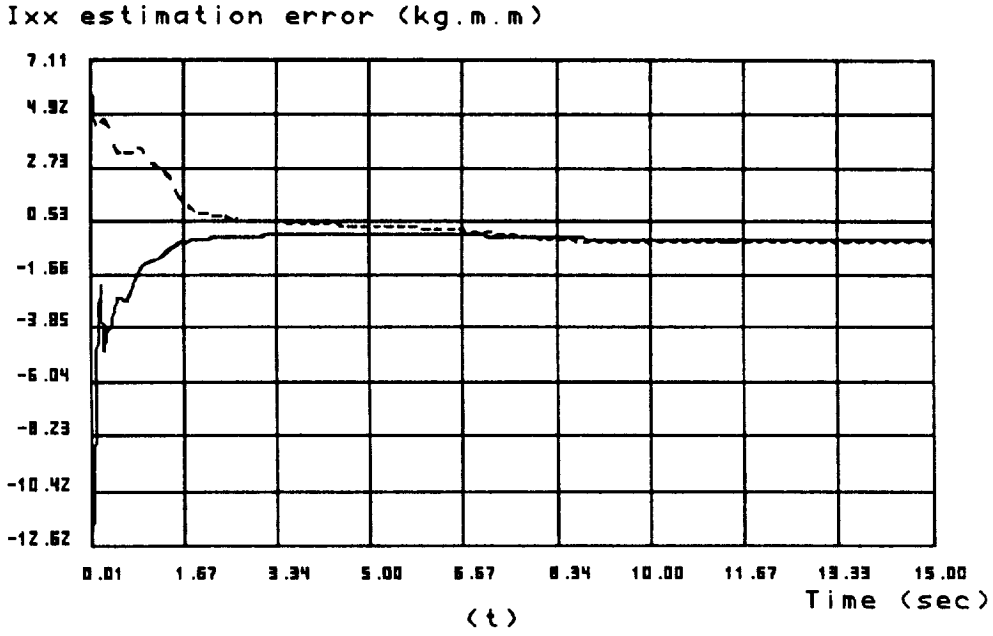


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

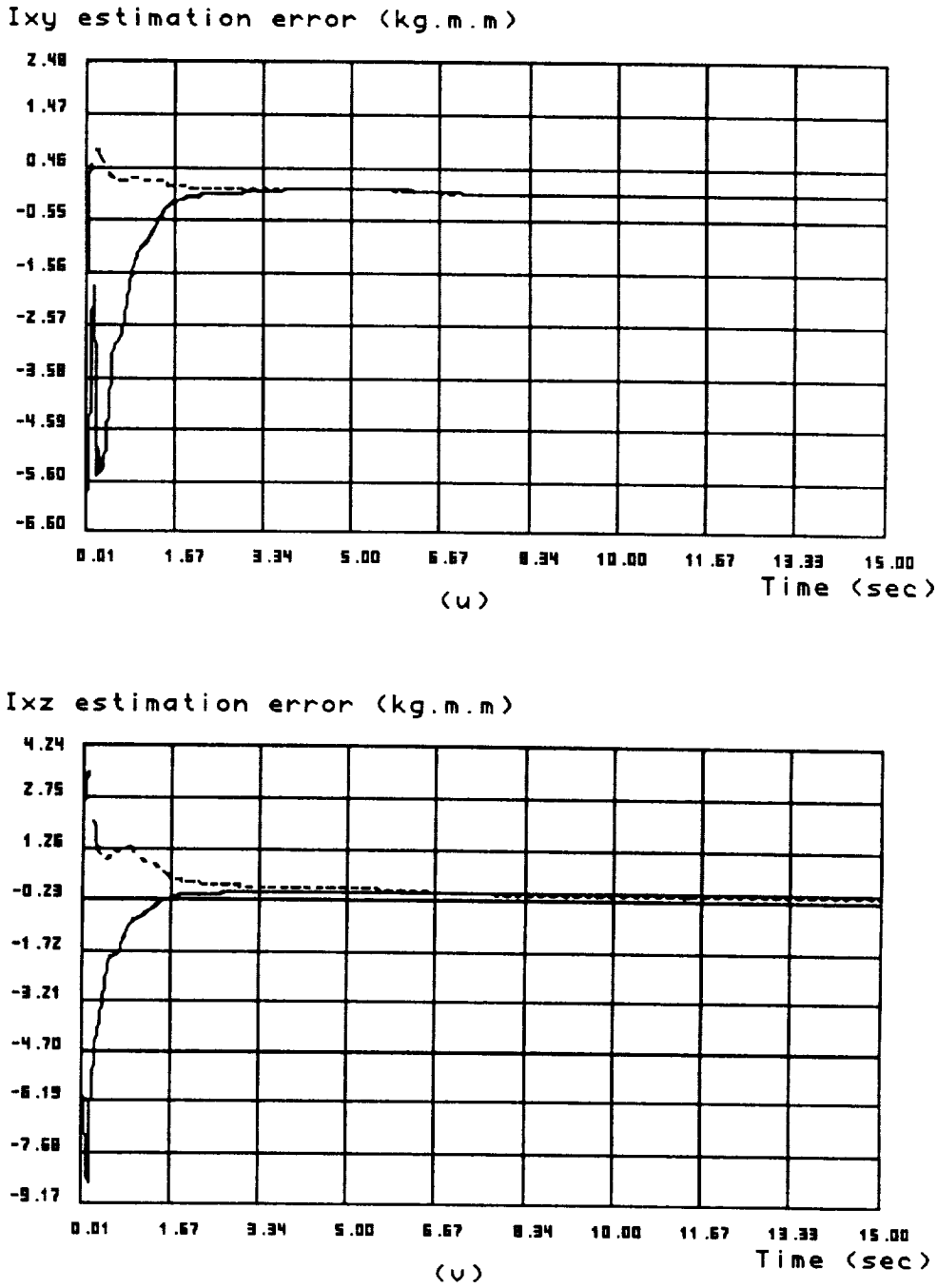


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

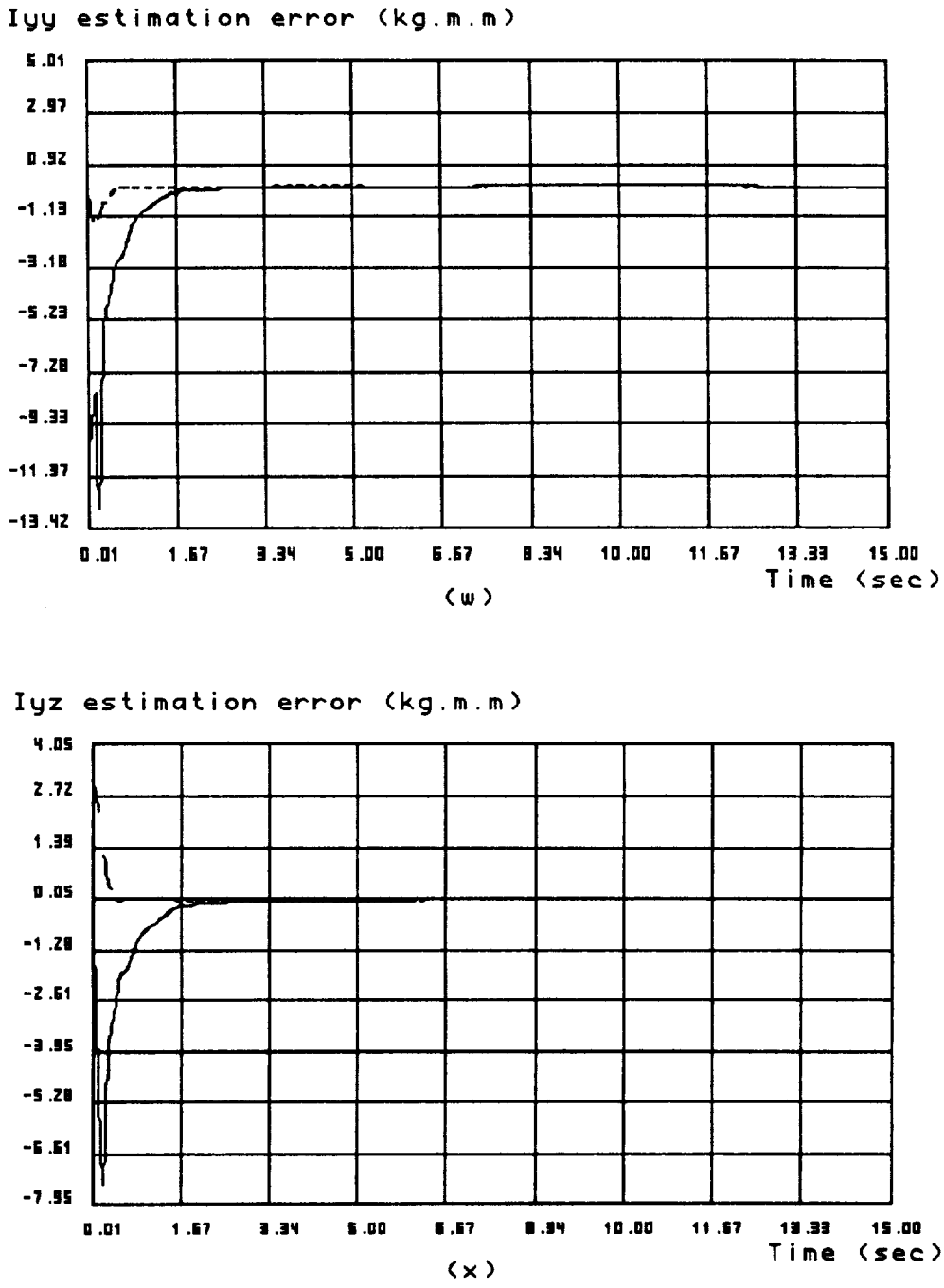


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

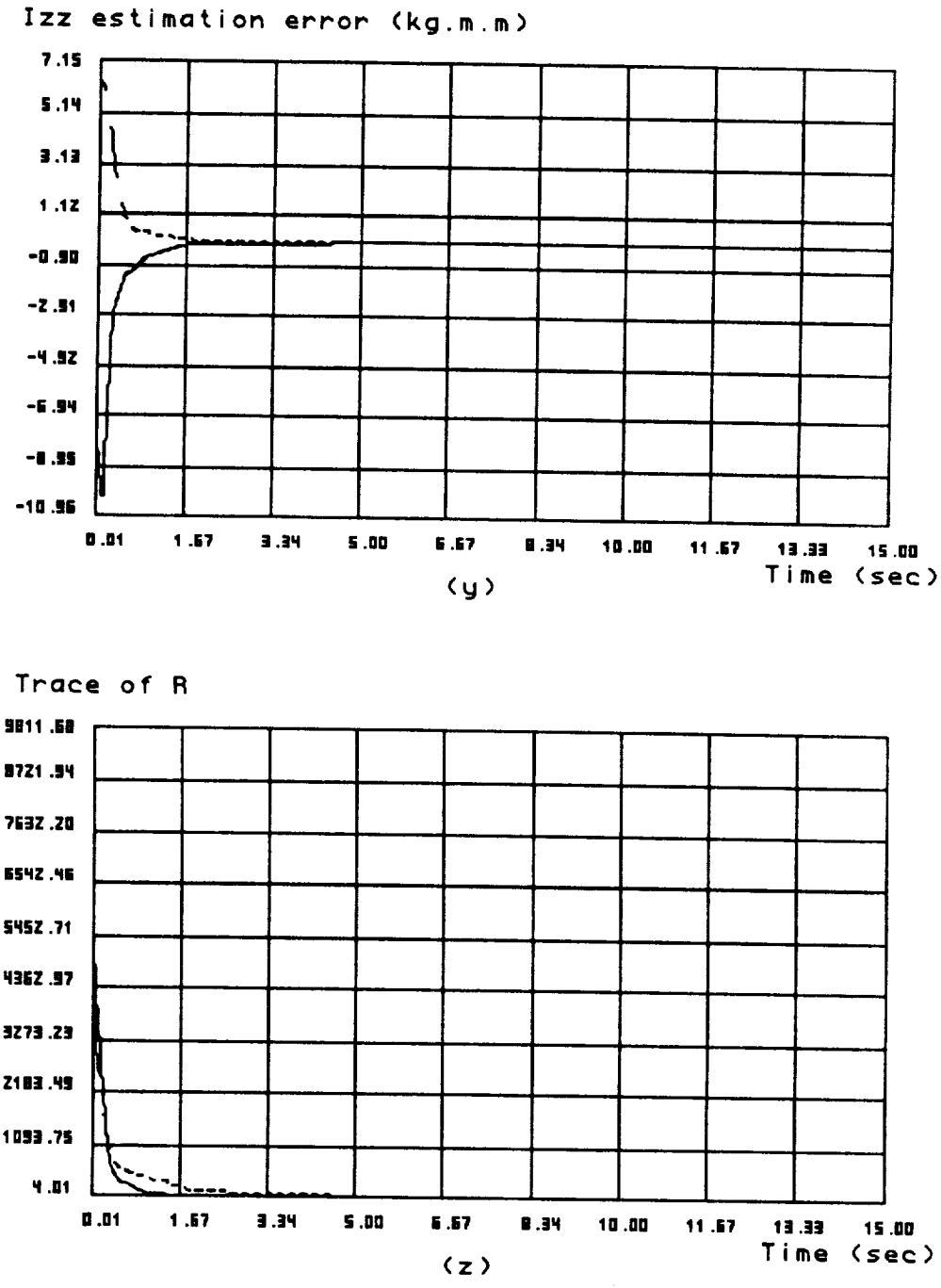


Figure 6.6: Adaptive Control of a 6-Link Robot with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

6.5 Effect of Update Rate on Adaptive Controller Performance

In the real-time implementation of the designed adaptive control algorithm, controlled system response will be affected by update delay. Since robotic dynamics are highly nonlinear, computation of dynamic parameters demands finite amount of time. Details of real-time computation of robot dynamic parameters are reported by [Wander and Tesar, 1987]. In physical implementation, the controller has to use a sequence of periodically updated dynamic parameters instead of continuously updated information to generate control command. Consequently, the update interruption creates a disturbance to the stability of the proposed controller and affects the controlled system response. In this section, update rate effect on controller performance will be analyzed, and from the analyzed results suggestions will be given to remedy the update disturbance. In the following analysis, subscript o denotes a updated value, and Δ indicates an update rate error at every instance.

In Equation 6.78 the control input u is composed of

$$u = u_1 + u_2 \tag{6.114}$$

Now, u_1 defined in Equation 6.79 is replaced by a periodically updated form given by

$$u_1 = [\Lambda_1 \ \Sigma^T]_{ko} \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (f_1)_{ko} + [\hat{\Lambda}_1 \ \hat{\Sigma}^T]_{uo} \begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (\hat{f}_1)_{uo} \tag{6.115}$$

where the subscript o indicates a given update value. With this new u_1, the system dynamics in Equation 6.82 become

$$\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix} \begin{bmatrix} \dot{e}_\theta \\ \dot{e}_\beta \end{bmatrix} = \begin{bmatrix} u_2 \\ -K\beta - S_1 \end{bmatrix} + Z_1\tilde{\mathbf{a}} + \begin{bmatrix} \Delta_1 \\ 0 \end{bmatrix} \tag{6.116}$$

where Δ_1 represents an error of Equation 6.82 due to update delay and is defined by

$$\begin{aligned}\Delta_1 = & \left([\Lambda_1\ \Sigma^T]_{ko} - [\Lambda_1\ \Sigma^T]_k\right)\begin{bmatrix}\ddot{\theta}_r\\ \ddot{\beta}_r\end{bmatrix} + (f_1)_{ko} - (f_1)_k \\ & + \left([\hat{\Lambda}_1\ \hat{\Sigma}^T]_{uo} - [\hat{\Lambda}_1\ \hat{\Sigma}^T]_u\right)\begin{bmatrix}\ddot{\theta}_r\\ \ddot{\beta}_r\end{bmatrix} + (\hat{f}_1)_{uo} - (\hat{f}_1)_u \end{aligned} \quad (6.117)$$

Then, this error is transported to the time derivative of the Lyapunov function given by Equation 6.93, that is

$$\begin{aligned}\dot{V} = & \rho\mathbf{e}^T\left\{\begin{bmatrix}u_2\\ -K\beta - S_1\end{bmatrix} + \begin{bmatrix}\Delta_1\\ 0\end{bmatrix} + \frac{1}{2}\begin{bmatrix}\dot{\Lambda}_1 & \dot{\Sigma}^T\\ \dot{\Sigma} & \dot{\Lambda}_2\end{bmatrix}_k\mathbf{e} + \frac{1}{2}\begin{bmatrix}\dot{\hat{\Lambda}}_1 & \dot{\hat{\Sigma}}^T\\ \dot{\hat{\Sigma}} & \dot{\hat{\Lambda}}_2\end{bmatrix}_u\mathbf{e}\right\} \\ & + \rho\mathbf{e}^T Z_1\tilde{\mathbf{a}} + \tilde{\mathbf{a}}^T R^{-1}\dot{\tilde{\mathbf{a}}} + \frac{1}{2}\tilde{\mathbf{a}}^T(\dot{R^{-1}})\tilde{\mathbf{a}} - \frac{1}{2}\rho Z_2\tilde{\mathbf{a}} \end{aligned} \quad (6.118)$$

Accordingly, u_2 originally given in Equation 6.94 is now constructed by updated parameters shown in the following equation

$$u_2 = u_3 - \frac{1}{2}[\dot{\Lambda}_1\ \dot{\Sigma}^T]_{ko}\mathbf{e} - \frac{1}{2}[\dot{\hat{\Lambda}}_1\ \dot{\hat{\Sigma}}^T]_{uo}\mathbf{e} \quad (6.119)$$

this new definition reduces Equation 6.118 to

$$\begin{aligned}\dot{V} = & \rho\mathbf{e}^T\begin{bmatrix}u_3\\ -K\beta - S_1 + S_2\end{bmatrix} + \rho\mathbf{e}^T\begin{bmatrix}\Delta_1 + \Delta_2\\ 0\end{bmatrix} \\ & + \tilde{\mathbf{a}}^T(R^{-1}\dot{\tilde{\mathbf{a}}} - \frac{1}{2}\rho Z_2^T + \rho Z_1^T\mathbf{e}) - \frac{1}{2}\sigma\tilde{\mathbf{a}}^T R^{-1}\tilde{\mathbf{a}} + \frac{1}{2}\varepsilon^T\varepsilon \end{aligned} \quad (6.120)$$

where

$$\begin{aligned}\Delta_2 = & -\frac{1}{2}\left([\dot{\Lambda}_1\ \dot{\Sigma}^T]_{ko} - [\dot{\Lambda}_1\ \dot{\Sigma}^T]_k\right)\mathbf{e} \\ & - \frac{1}{2}\left([\dot{\hat{\Lambda}}_1\ \dot{\hat{\Sigma}}^T]_{uo} - [\dot{\hat{\Lambda}}_1\ \dot{\hat{\Sigma}}^T]_u\right)\mathbf{e} \end{aligned} \quad (6.121)$$

In the adaptive controller design, u_3 is initially solved from the equation

$$e_\theta^T u_3 + \gamma_1 = 0 \quad (6.122)$$

now, the solution is updated by

$$u_3 = -\frac{\gamma_{1o}}{e_\theta^T e_\theta} e_\theta \tag{6.123}$$

with γ_{1o} defined by

$$\gamma_{1o} = e_\beta^T(-K\beta - S_{1o} + S_{2o}) + \frac{\alpha_1}{\rho}\mathbf{e}^T\mathbf{e} \tag{6.124}$$

where

$$\begin{aligned} S_{1o} &= [\Sigma\ \Lambda_2]_{ko}\begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (f_2)_{ko} + [\hat{\Sigma}\ \hat{\Lambda}_2]_{uo}\begin{bmatrix} \ddot{\theta}_d \\ \ddot{\beta}_d \end{bmatrix} + (\hat{f}_2)_{uo} \\ S_{2o} &= \frac{1}{2}[\dot{\Sigma}\ \dot{\Lambda}_2]_{ko}\mathbf{e} + \frac{1}{2}[\dot{\hat{\Sigma}}\ \dot{\hat{\Lambda}}_2]_{uo}\mathbf{e} \end{aligned} \tag{6.125}$$

which creates an error Δ_3 to Equation 6.122, i.e.,

$$\begin{aligned} e_\theta^T u_3 + \gamma_1 &= e_\beta^T(S_{1o} - S_1 - S_{2o} + S_2) \\ &= \Delta_3 \end{aligned} \tag{6.126}$$

where

$$\Delta_3 = e_\beta^T(S_{1o} - S_1 - S_{2o} + S_2) \tag{6.127}$$

With the additional errors, Equation 6.120 becomes

$$\begin{aligned} \dot{V} &= -\alpha_1\mathbf{e}^T\mathbf{e} + \rho\Delta_3 + \rho e_\theta^T(\Delta_1 + \Delta_2) \\ &\quad + \tilde{\mathbf{a}}^T(R^{-1}\dot{\tilde{\mathbf{a}}} - \frac{1}{2}\rho Z_2^T + \rho Z_1^T\mathbf{e}) - \frac{1}{2}\rho\tilde{\mathbf{a}}^T R^{-1}\tilde{\mathbf{a}} + \frac{1}{2}\varepsilon^T\varepsilon \end{aligned} \tag{6.128}$$

The analysis is now focused on the parameter estimation part in the above $\dot{V}$ equation which is also affected by update delay. According to Equation 6.98, the parameter estimation increment is computed by the update form

$$\dot{\tilde{\mathbf{a}}} = R_o(\frac{1}{2}\rho Z_{2o}^T - \rho Z_{1o}^T\mathbf{e} - Z_{0o}^T\varepsilon_o) \tag{6.129}$$

whose effect on the original design is derived below

$$\begin{aligned}
& \tilde{\mathbf{a}}^T(R^{-1}\dot{\tilde{\mathbf{a}}} - \frac{1}{2}\rho Z_2^T + \rho Z_1^T\mathbf{e}) \\
= \; & \tilde{\mathbf{a}}^T[R^{-1}R_o(\frac{1}{2}\rho Z_{2o}^T - \rho Z_{1o}^T\mathbf{e} - Z_{0o}^T\varepsilon_o) - \frac{1}{2}\rho Z_2^T + \rho Z_1^T\mathbf{e}] \\
= \; & \tilde{\mathbf{a}}^T[R^{-1}(R+\Delta R)(\frac{1}{2}\rho Z_{2o}^T - \rho Z_{1o}^T\mathbf{e} - Z_{0o}^T\varepsilon_o) - \frac{1}{2}\rho Z_2^T + \rho Z_1^T\mathbf{e}] \\
= \; & \tilde{\mathbf{a}}^T[R^{-1}\Delta R(\frac{1}{2}\rho Z_{2o}^T - \rho Z_{1o}^T\mathbf{e} - Z_{0o}^T\varepsilon_o) + \frac{1}{2}(\rho Z_{2o}^T - \rho Z_2^T) \\
& + (-\rho Z_{1o}^T + \rho Z_1^T)\mathbf{e} + (-Z_{0o}^T + Z_0^T)\varepsilon_o] - \tilde{\mathbf{a}}^T Z_0^T\varepsilon_o \\
= \; & \tilde{\mathbf{a}}^T\rho\Delta_4 + \tilde{\mathbf{a}}^T\Delta_5 - \varepsilon^T\varepsilon
\end{aligned} \tag{6.130}$$

where errors ΔR, Δ_4, and Δ_5 are

$$\begin{aligned}
\Delta R &= R_o - R \\
\Delta_4 &= \frac{1}{2}R^{-1}\Delta R Z_{2o}^T - R^{-1}\Delta R Z_{1o}^T\mathbf{e} + \frac{1}{2}(Z_{2o}^T - Z_2^T) - (Z_{1o}^T - Z_1^T)\mathbf{e} \\
\Delta_5 &= -R^{-1}\Delta R Z_{0o}^T\varepsilon_o - (Z_{0o}^T - Z_0^T)\varepsilon_o - Z_0^T(\varepsilon_o - \varepsilon) \\
&= -R^{-1}\Delta R Z_{0o}^T\varepsilon_o - \Delta Z_0^T\varepsilon_o - Z_0^T\Delta\varepsilon
\end{aligned} \tag{6.131}$$

with

$$\begin{aligned}
\Delta Z_0 &= Z_{0o} - Z_0 \\
\Delta\varepsilon &= \varepsilon_o - \varepsilon
\end{aligned} \tag{6.132}$$

A substitution of these results into Equation 6.128, $\dot{V}$ results in

$$\begin{aligned}
\dot{V} = \; & -\alpha_1\mathbf{e}^T\mathbf{e} - \frac{1}{2}\rho\tilde{\mathbf{a}}^T R^{-1}\tilde{\mathbf{a}} - \frac{1}{2}\varepsilon^T\varepsilon \\
& + \rho[\Delta_3 + e_\theta^T(\Delta_1 + \Delta_2) + \tilde{\mathbf{a}}^T\Delta_4] + \tilde{\mathbf{a}}^T\Delta_5
\end{aligned} \tag{6.133}$$

Apparently, the errors caused by periodical update will create additional uncertainty on the stability analysis of $\dot{V}$. However, since ρ is a control parameter decided by users, it could be very small to reduce the error term

$$\rho[\Delta_3 + e_\theta^T(\Delta_1 + \Delta_2) + \tilde{\mathbf{a}}^T\Delta_4]$$

in the above $\dot{V}$ expression, which suggests that ρ could serve as a parameter to improve controller performance. Besides this new function, small ρ has also been used to enhance parameter estimation precision in earlier case studies. But the $\dot{V}$ in Equation 6.133 is still under the influence of $\tilde{\mathbf{a}}^T\Delta_5$ where Δ_5 is defined in Equation 6.131. Since Δ_5 relates primarily to the update errors of R, Z_0, and ε, they are further analyzed in the followings. Recalling the definition of Z' in Equation 6.62 that

$$\begin{aligned} Z'_{0o} &= \left[\cdots \left(I_i^* + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix}\right) \cdots\right]_o ; \; i \in \{1, 2, \cdots, 10\} \\ &= \left[\cdots \left(I_i^* + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix}\right) \cdots\right] \\ &\quad + \left\{\left[\cdots \left(I_i^* + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix}\right) \cdots\right]_o - \left[\cdots \left(I_i^* + \begin{bmatrix} (f_1)_i \\ (f_2)_i \end{bmatrix}\right) \cdots\right]\right\} \\ &\stackrel{\text{def}}{=} Z'_0 + \Delta Z'_0 \end{aligned} \tag{6.134}$$

and also from Equation 6.84

$$\begin{aligned} U'_o &= \begin{bmatrix} u \\ -K\beta \end{bmatrix} - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_{ko} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_{ko} \\ &= \begin{bmatrix} u \\ -K\beta \end{bmatrix} - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_{k} \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_{k} \\ &\quad - \left(\begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_{ko} - \begin{bmatrix} \Lambda_1 & \Sigma^T \\ \Sigma & \Lambda_2 \end{bmatrix}_{k}\right) \begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} - \left(\begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_{ko} - \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}_{k}\right) \\ &\stackrel{\text{def}}{=} U' + \Delta U' \end{aligned} \tag{6.135}$$

In order to remove the acceleration dependence of Z'_{0o} and U'_o, two integrations over time are executed which produce

$$\begin{aligned} Z_{0o} &= \int (Z'_0 + \Delta Z'_0) dt \\ &\stackrel{\text{def}}{=} Z_0 + \Delta Z_0 \end{aligned} \tag{6.136}$$

$$\begin{aligned} U_o &= \int (U' + \Delta U') dt \\ &\stackrel{\text{def}}{=} U + \Delta U \end{aligned} \tag{6.137}$$

where $\Delta(\cdot)$ represents a update error. Finally, the update errors of ε and R, from Equations 6.85 and 6.48, are

$$\begin{aligned} \varepsilon_o &= Z_{0_o}\hat{\mathbf{a}} - U_o \\ &= (Z_0 + \Delta Z_0)\hat{\mathbf{a}} - (U + \Delta U) \\ &= (Z_0\hat{\mathbf{a}} - U) + (\Delta Z_0\hat{\mathbf{a}} - \Delta U) \\ &\stackrel{\text{def}}{=} \varepsilon + \Delta\varepsilon \end{aligned} \tag{6.138}$$

$$\begin{aligned} \dot{R}_o &= \sigma R_o - R_o Z_{0_o}^T Z_{0_o}^T R_o \\ &= \sigma R_o - R_o(Z_0 + \Delta Z_0)^T(Z_0 + \Delta Z_0)R_o \\ &= \sigma R_o - R_o(Z_0^T Z_0)R_o - R_o(\Delta Z_0^T)(Z_0)R_o \\ &\quad -R_o(Z_0^T)(\Delta Z_0)R_o - R_o(\Delta Z_0^T)(\Delta Z_0)R_o \end{aligned} \tag{6.139}$$

and

$$\Delta R = R_o - R$$

Unfortunately, from the above analytical results, no free parameter like ρ could be chosen to reduce the effect of ΔR, ΔZ_0, and $\Delta\varepsilon$ on Δ_5 and hence on $\dot{V}$ values given in Equation 6.133. Furthermore, a small ρ as suggested above means that the estimation process in Equation 6.129 behaves like a least square method result which is also affected by the errors of R, Z_0, and ε. So some practical approaches must be adopted to solve these problems. One solution proposed here is to stop estimation once the system ceases persistent excitation. The reason is that when the system stops excitation, the matrix R and its computed value R_o are small and ideally zero, so the error ΔR is small. Also, the previous case study shows that payload estimates approach to the exact values when R stops decreasing. This means that ε and ε_o and therefore $\Delta\varepsilon$ are almost zero. Hence the resultant Δ_5 is small,

consequently its effect on $\dot{V}$ in Equation 6.133 is reduced. One question left to be answered is what criterion should be used to decide that a system has ceased excitation. In the previous case studies, the trace of R is used as an indicator of persistent excitation, so the decrement rate of the trace could serve the purpose. Practically, when the decrement rate of the trace is less than a predefined value, the estimation process is stopped and the controller will use the last payload estimates to compute control command for the rest of motion.

6.6 Case Studies with 100 Hz Update Rate

The update effect will be demonstrated on the six-link model. Again, both zero initial estimate and overestimated examples are repeated except that now control parameters have a 100 Hz update rate while the integration step remains 1 msec. All system and control parameters are the same as before. The major changes are that $R(0)$ has a value 10,000 for its diagonal elements and $\rho = 0.000001$ is used to reduced update disturbance. In both simulations, the estimation is terminated when the trace of R_o has a decrement less than 0.5 over an update step. This 0.5 value is selected based on our experience from previous simulation results. Notice that $R(0)$ is 10 times the value used in the last case study. This selection is to have a better approximation to the exact $R(0)$ which is actually an infinite matrix. However, it should be pointed out that a large $R(0)$ will make estimation too sensitive to initial estimation errors. Both simulation results are reported in Figures 6.7 (a) to (z). In each figure, a solid line represents the zero initial estimate case result, and a broken link depicts the overestimated case response. By comparing Figures 6.7 with Figures 6.6 where update effect is omitted, it

appears that by choosing a small ρ and terminating estimation the update rate disturbance is reduced. The simulations results in Figures 6.7 are similar to that in Figure 6.6. The major differences are that the moment of inertia estimates in the overestimated case do not converge to the exact values when estimation is terminated, as indicated by the broken lines in Figures 6.7 (t) to (y). According to simulation data, the zero initial case ceases estimation at 1.07 sec where the trace of R is 41.9; the overestimated case stops estimation at 1.89 sec where the trace of R is 147. Again, the zero initial estimate case has better excitation than that of the overestimated case, which is due to the large overshoot of wrist motion shown in Figures 6.7 (d) to (f). The case study results confirm our suggestion that update delay impact on adaptive controller performance could be remedied by using small ρ and termination of estimation.

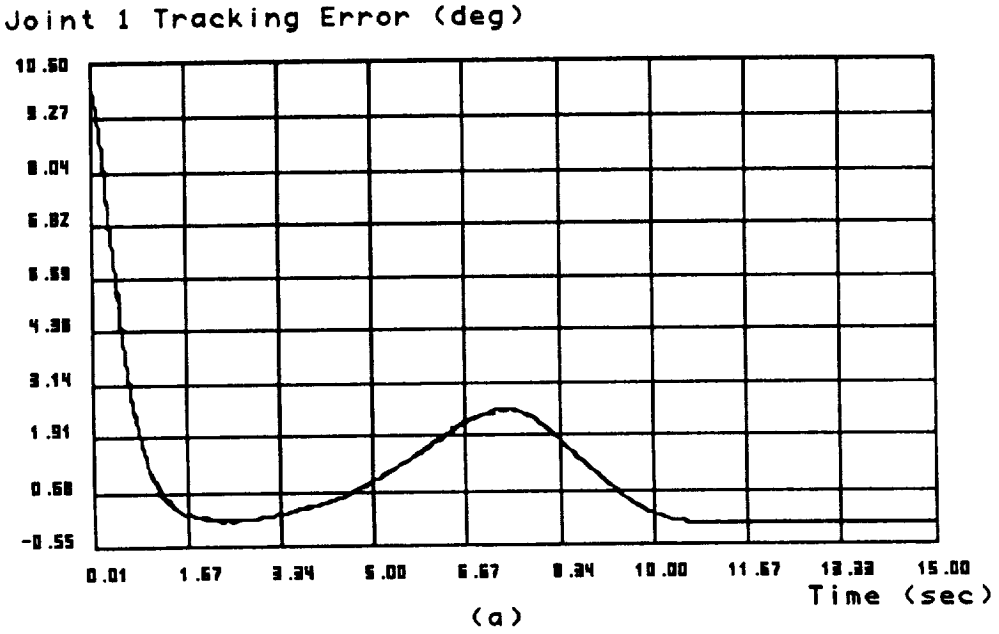
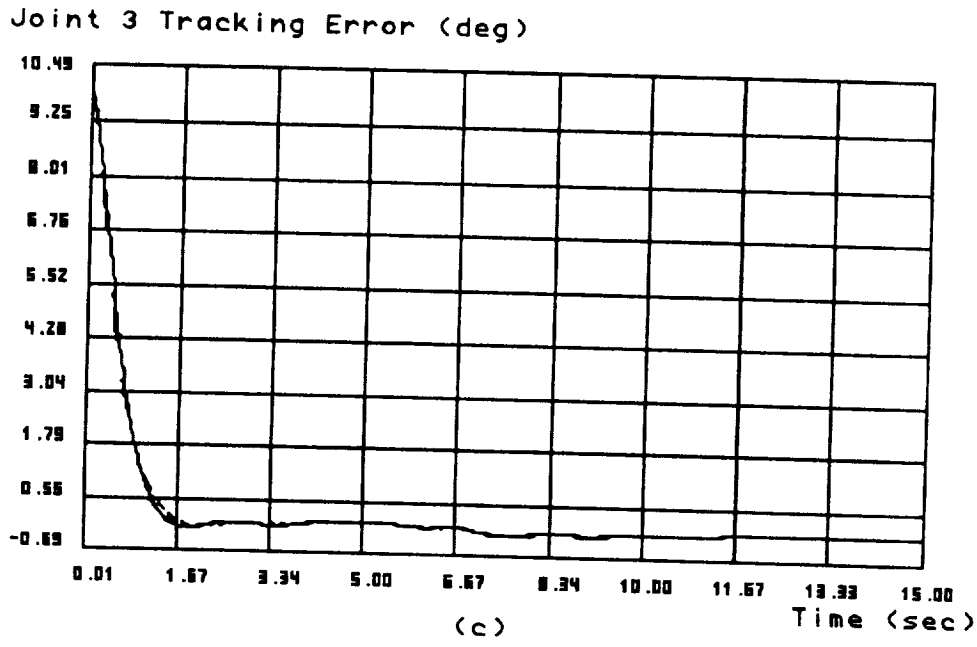


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



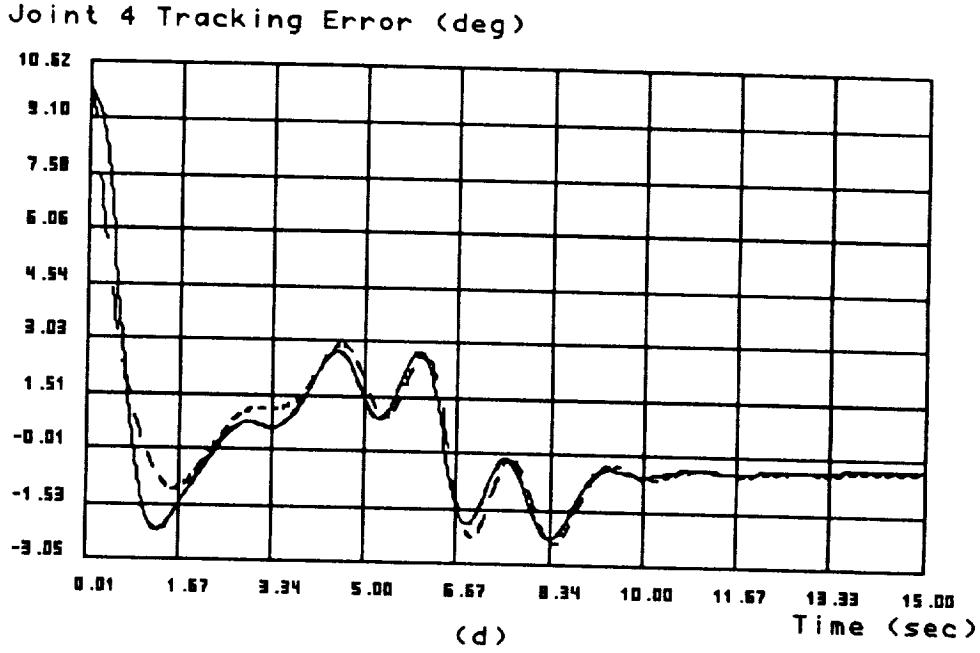
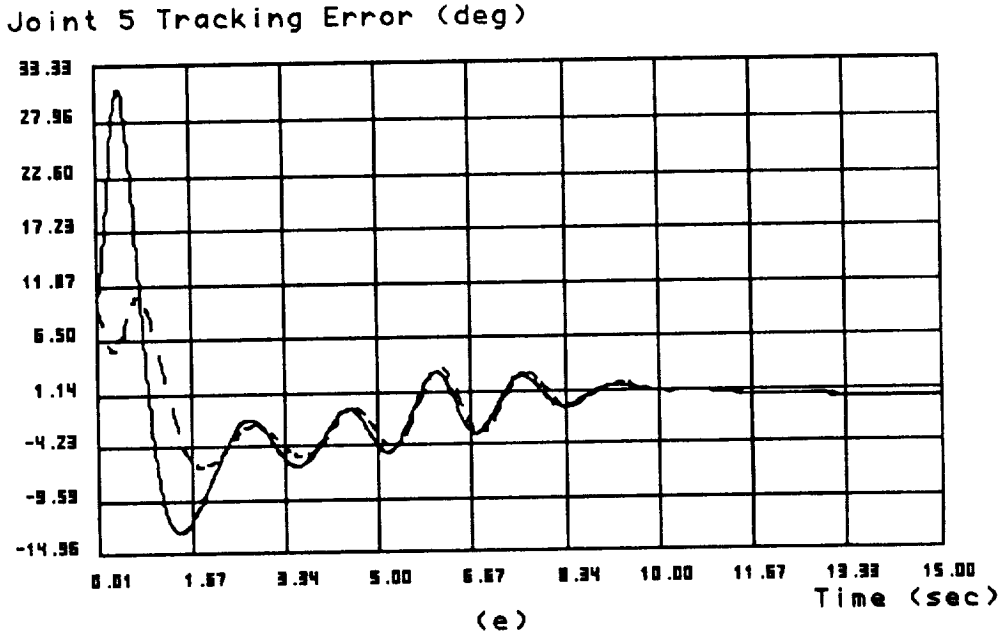


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



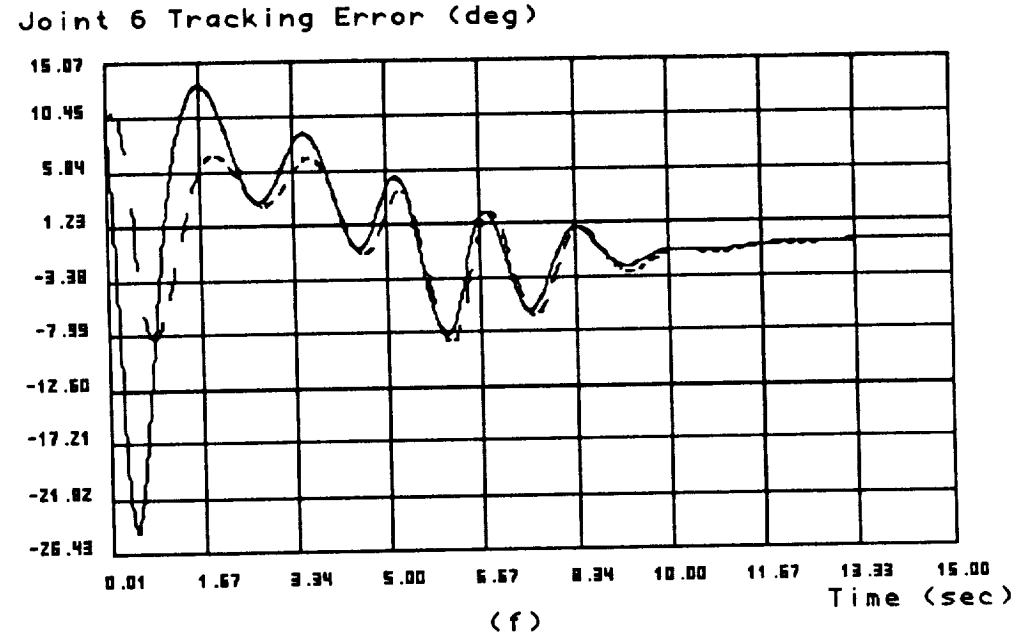


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

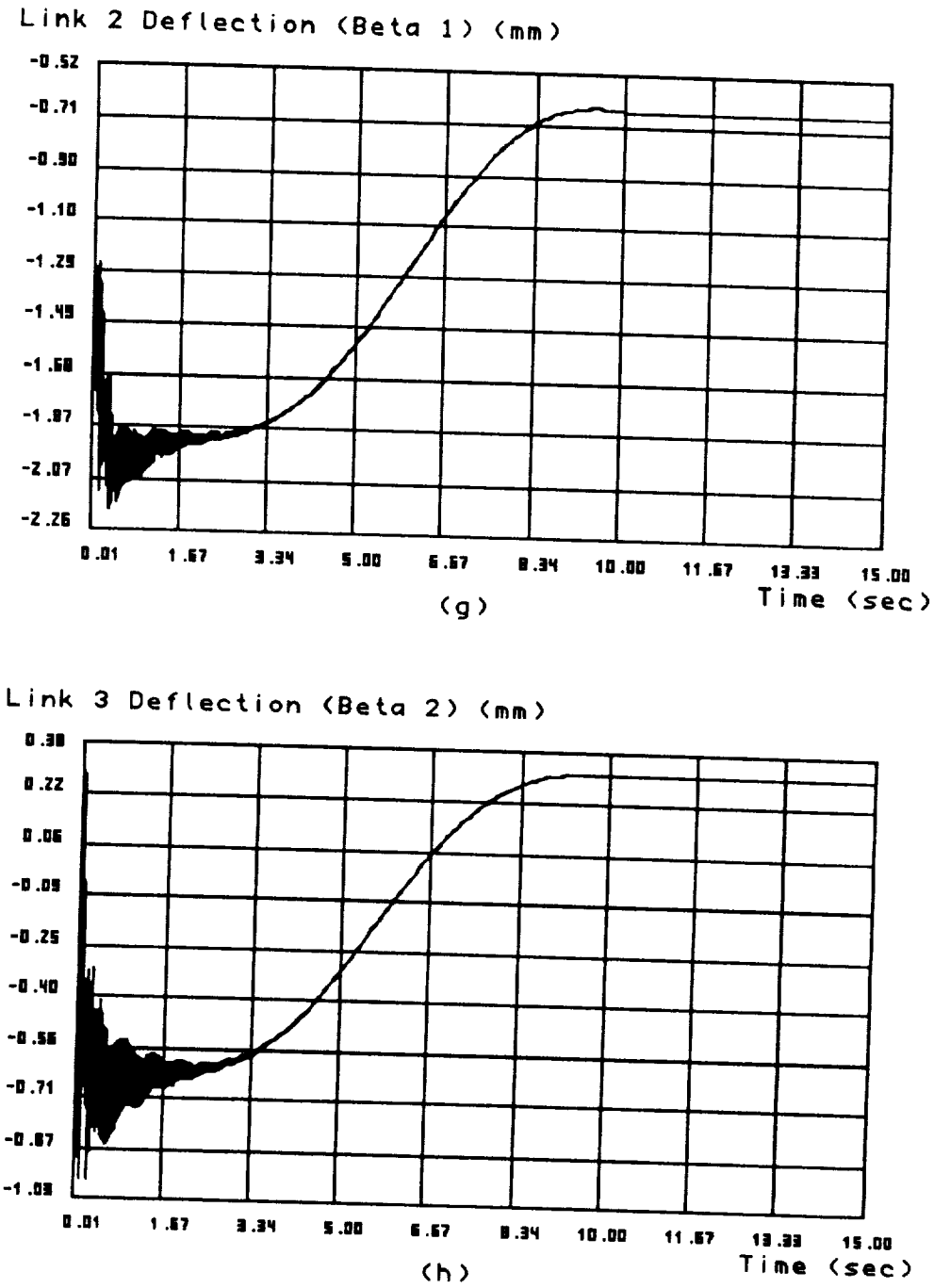
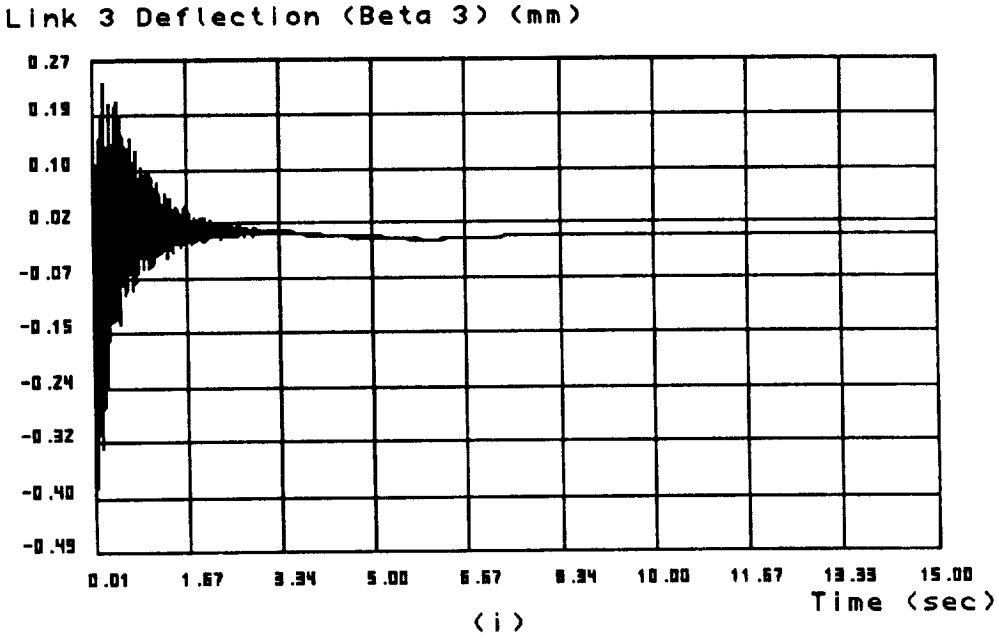


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



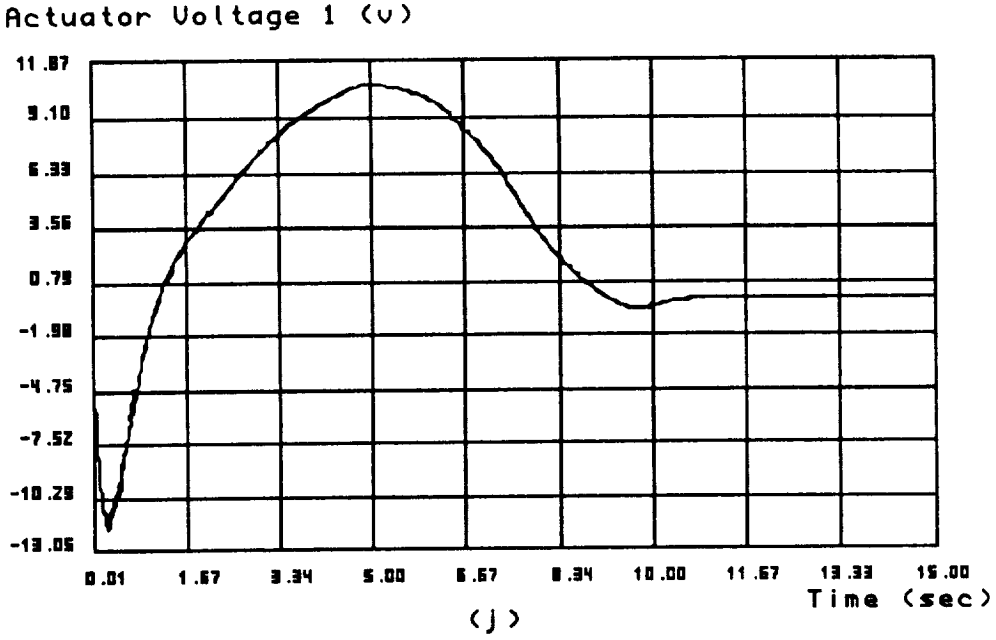


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

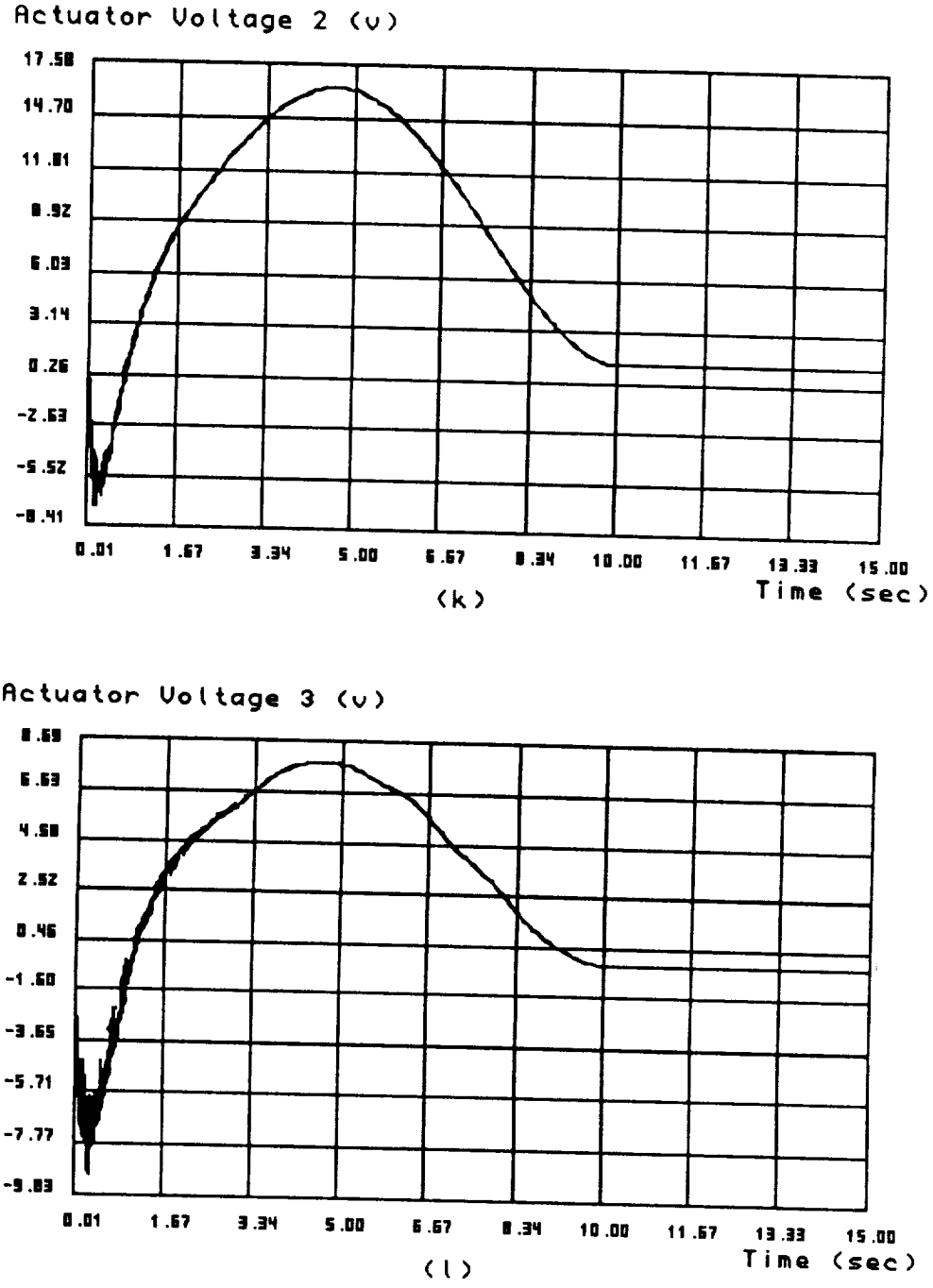
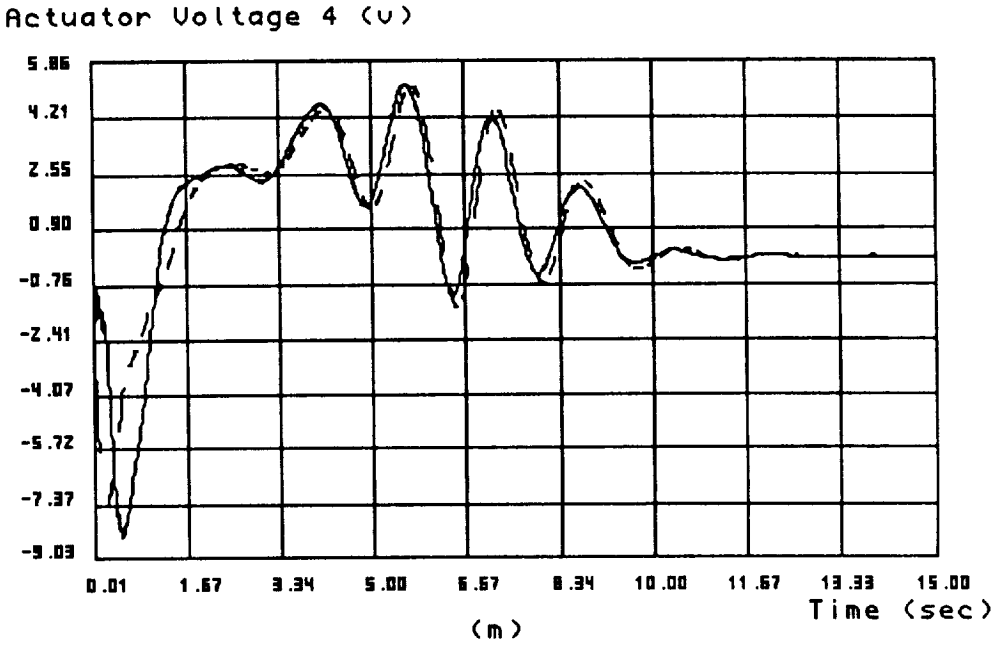


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



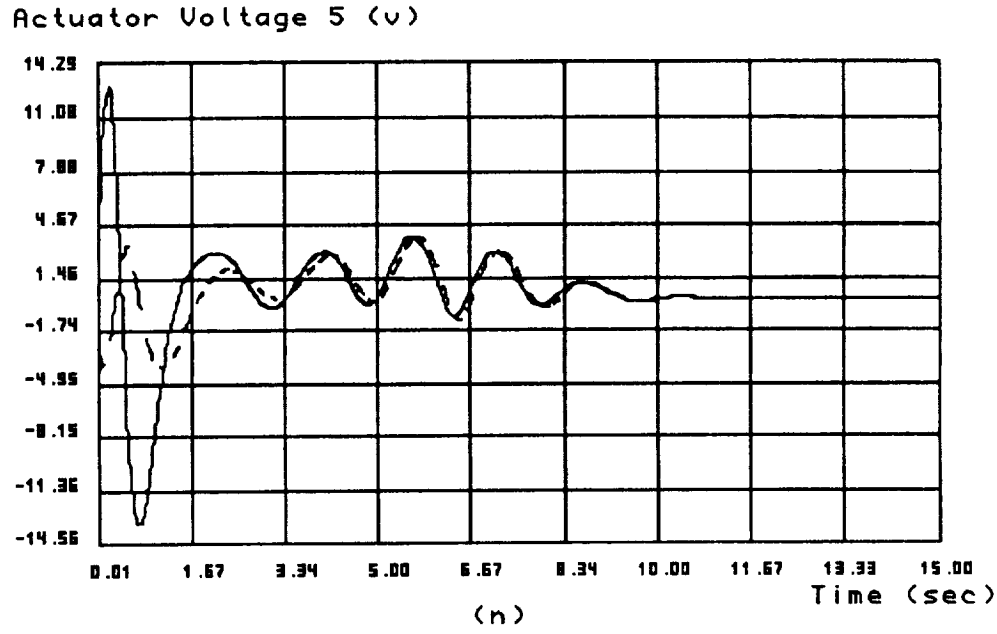
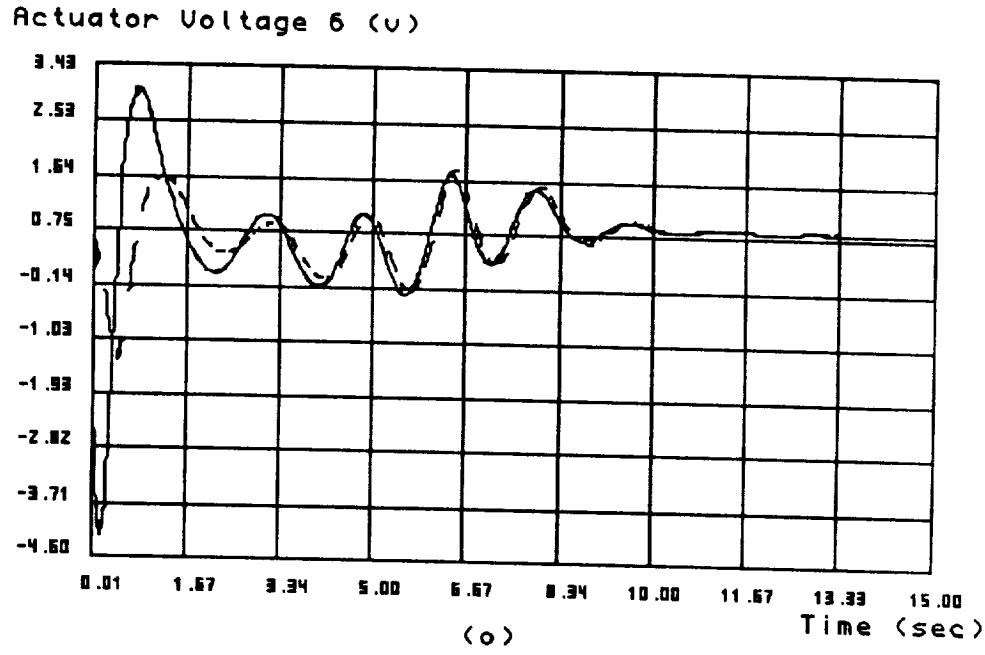


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



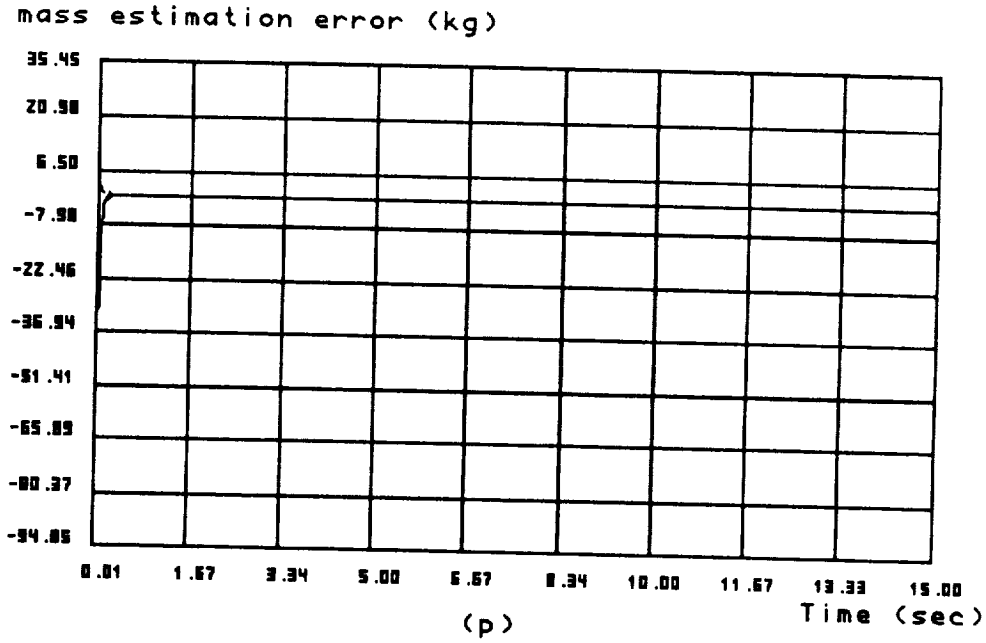


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

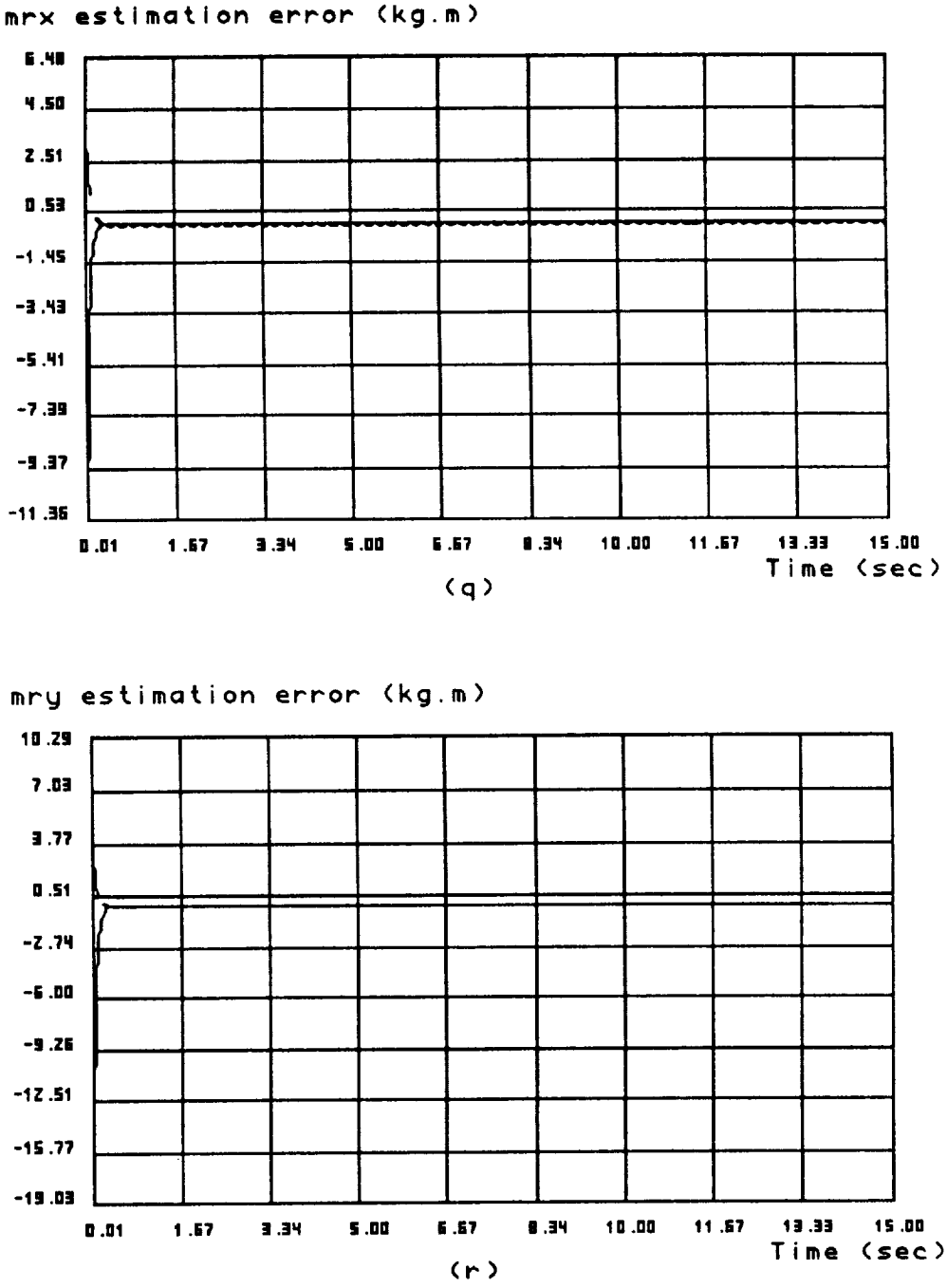


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

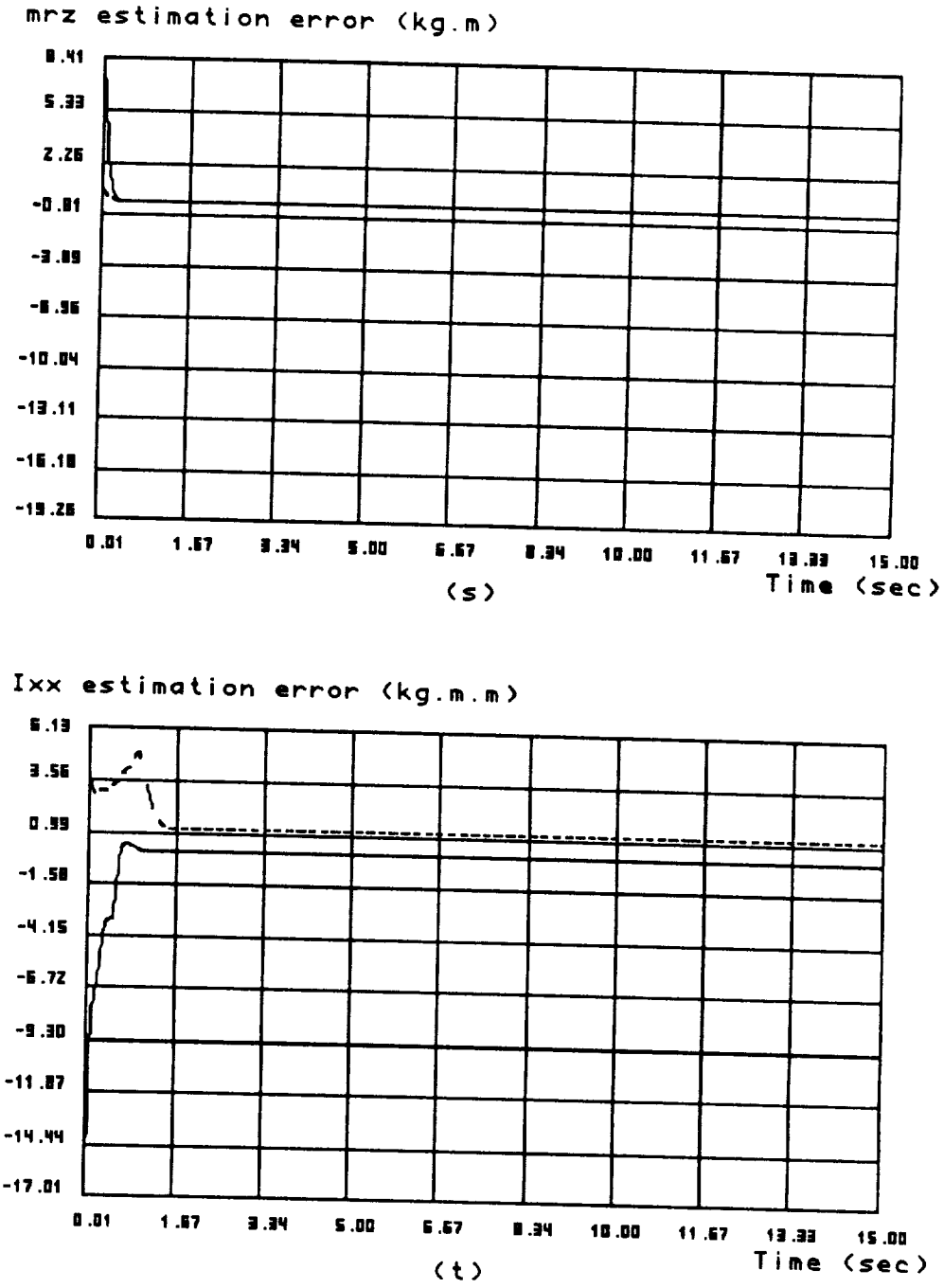
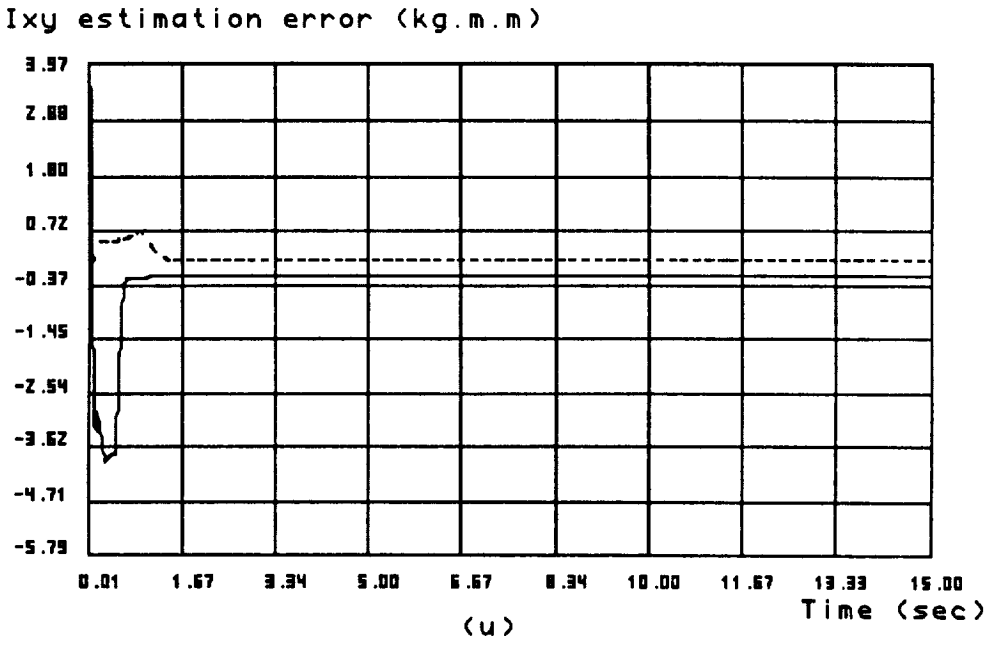


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)



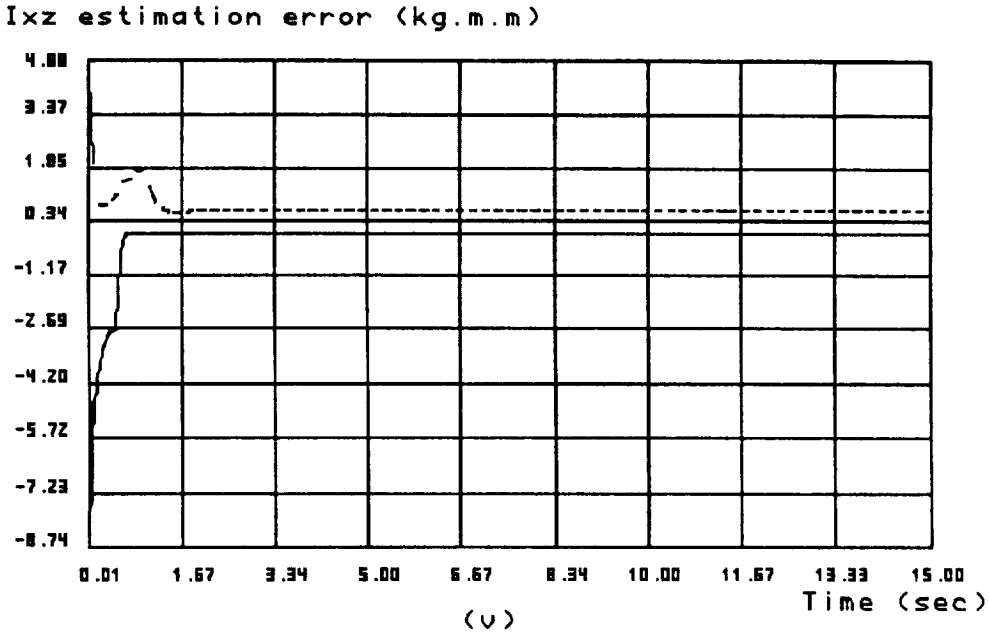


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

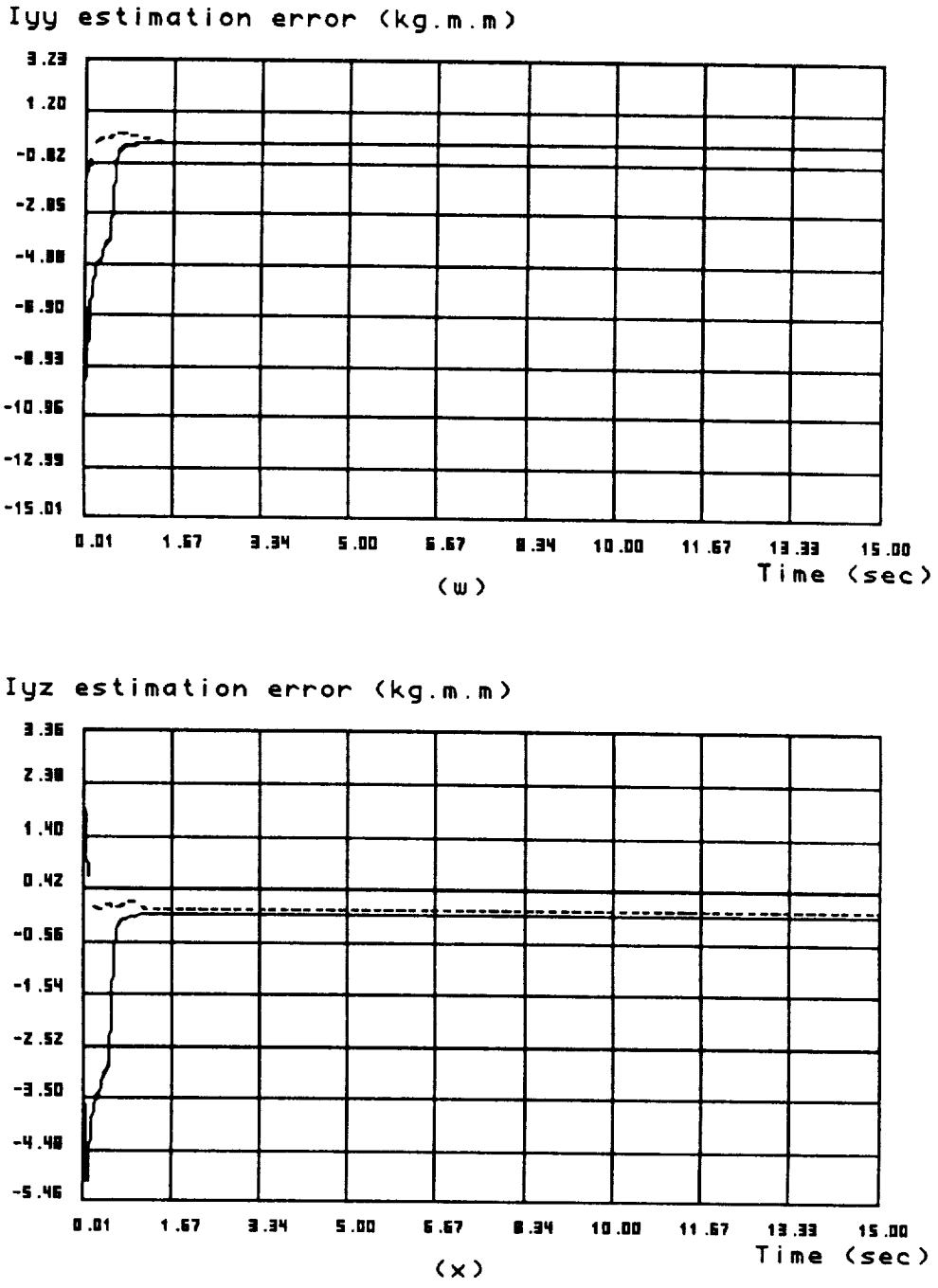


Figure 6.7: Adaptive Control of a 6-Link Robot at 100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

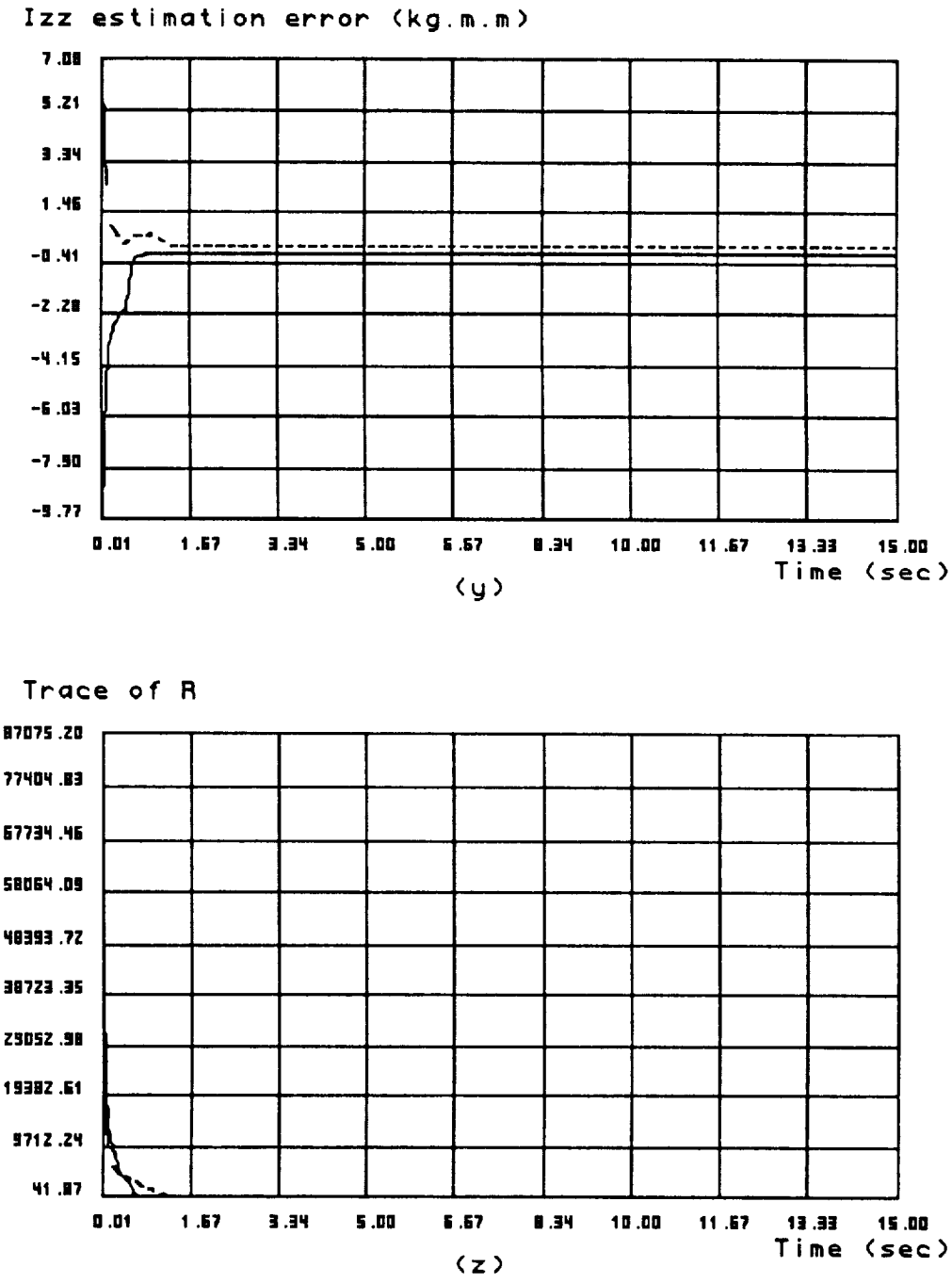


Figure 6.7: Adaptive Control of a 6-Link Robot at100 Hz Update Rate with Zero Initial Estimates (Solid Line) and Overestimated Initial Values (Broken Line)

6.7 Comparison Between Adaptive and Non-Adaptive Control

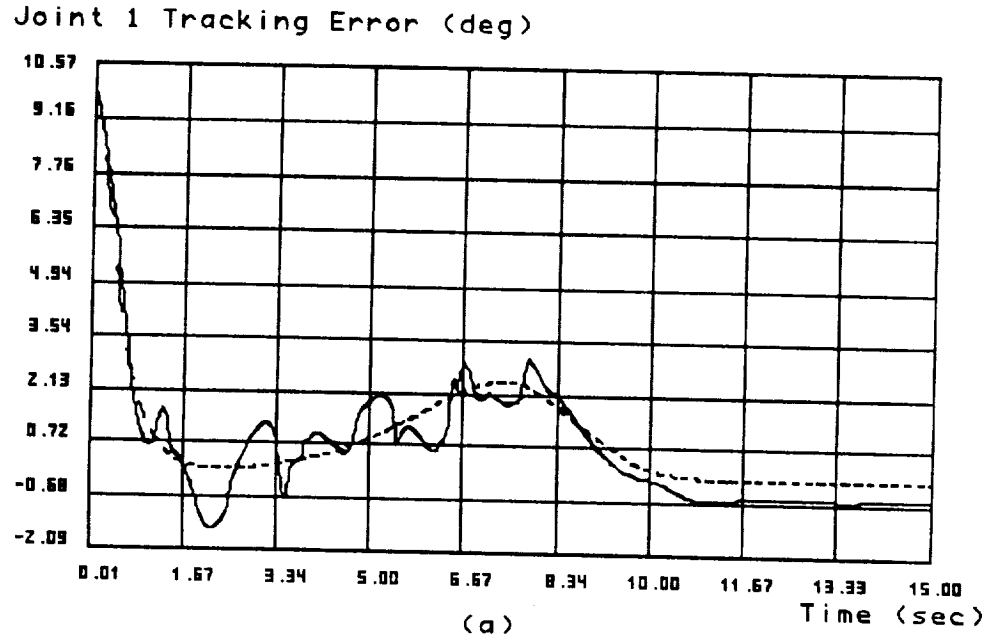
In order to illustrate the merit of adaptive control in uncertain systems, we conduct the following comparison. Now, the parameter estimation in Figure 6.2 is disconnected from control system. Therefore, by setting $\dot{\tilde{\mathbf{a}}} = \tilde{\mathbf{a}} = 0$ for all t, the above adaptive controller becomes a regular controller built for the system with well-known inertial parameters including the payload. Without the parameter estimation, the new regular controller is employed to control the six-link model with overestimated payload. This means that the regular controller constructs control commands using a set of payload values that are larger than exact values. The simulation results are displayed by solid line in Figures 6.8 (a) to (f) for nominal mode tracking errors, Figures 6.8 (g) to (i) for link deflections, and Figures 6.8 (j) to (o) for control voltages. For comparison purposes, the adaptive control results of the overestimated case are also presented by broken line. Apparently, without adaptive effort, the regular controller has very poor performance in this case. The overshoots of wrist motion and the residual oscillation of β_3 in Figures 6.8 (i) are unacceptable. Also, the wrist control voltages are beyond reasonable values. However, another aspect regarding this example is that without proper design a controller might bring system instability to our studied models, which means that the examples used in this report are not intrinsically robust. A carelessly designed controller would fail the assigned control task. Of course, according to the Chapter 5 results, a well-designed robust controller which contains no adaptive loop is applicable to a system with uncertain parameters. Therefore, experimental work should be conducted to compare the performance between adaptive and non-adaptive controllers. Some comparison criteria are suggested below, which are general for rigid and compliant

robotic systems:

1. settling time: nominal motion and residual oscillation
2. reference motion tracking precision
3. required actuator energy
4. capability of uncertainty estimation
5. estimation convergence speed of adaptive algorithm
6. real-time computation effort
7. robustness to unmodeled disturbance
8. applicability to compliant structure
9. applicability to modularized structure
10. required on-line measurement
11. tuning capability such as selection of poles in PID, P matrix in the Lyapunov function, ρ in the adaptive control law, and so forth.

For a given robot and task, different control algorithms should be tested experimentally. The results could be tabulated in a matrix format following the above listed criteria. This table will help users to choose the best controller for a specified task. Unfortunately, due to nonlinearity of robot dynamics, selected control parameters like PID gains or the P matrix are difficult to parameterize explicitly to analyze their effects on controlled system performance. So, one experimental result could not be extended directly to another

task. Therefore, a collection of various task results will be very useful to create a general table containing different controller performance, which could serve as an indicator of the preferable control algorithm for a given task.



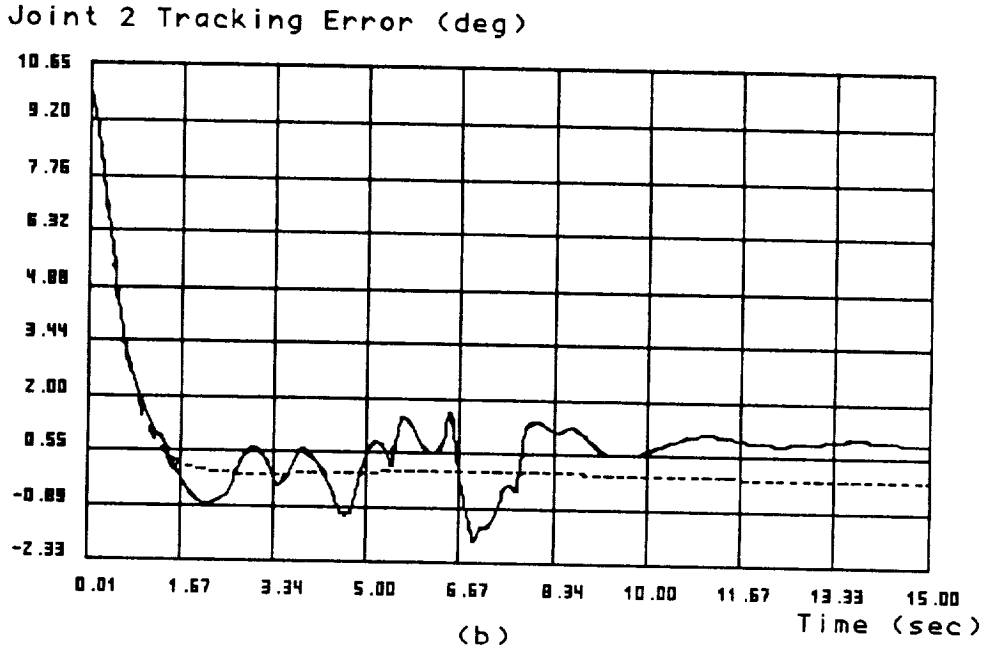
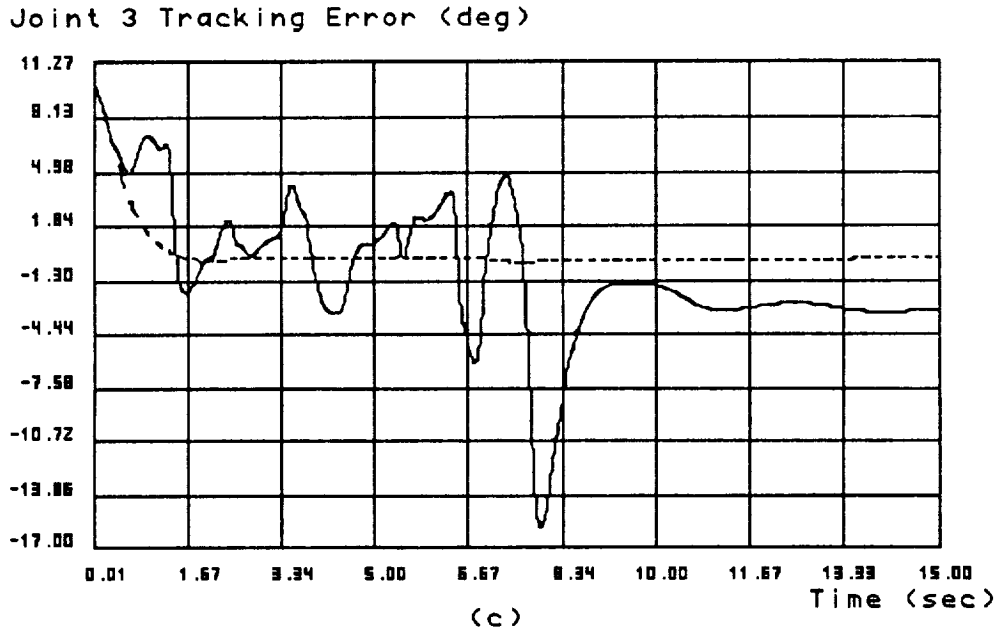


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at 100 Hz Update Rate



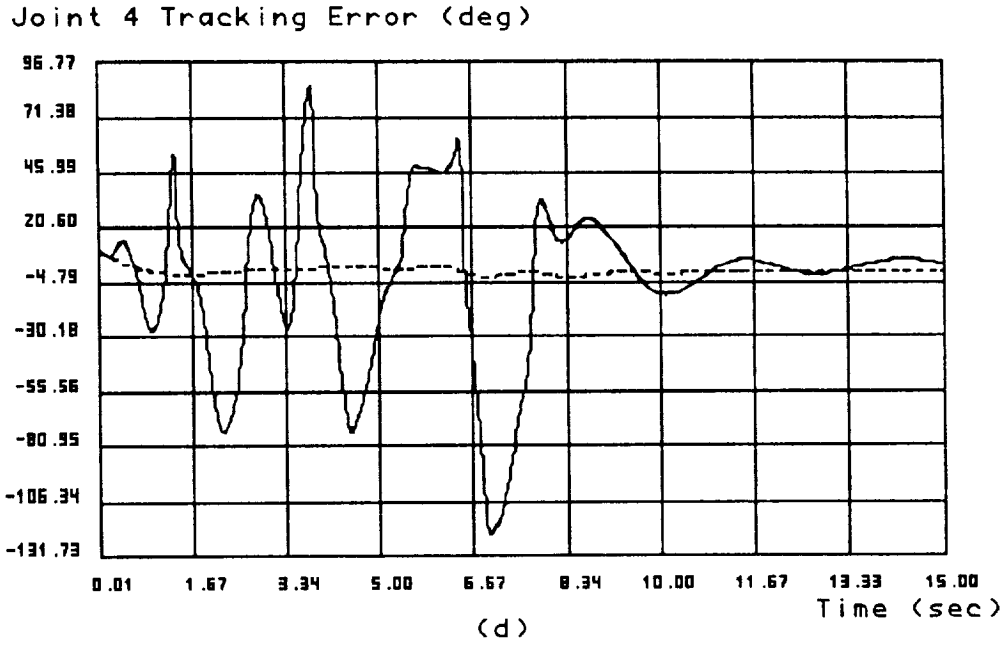
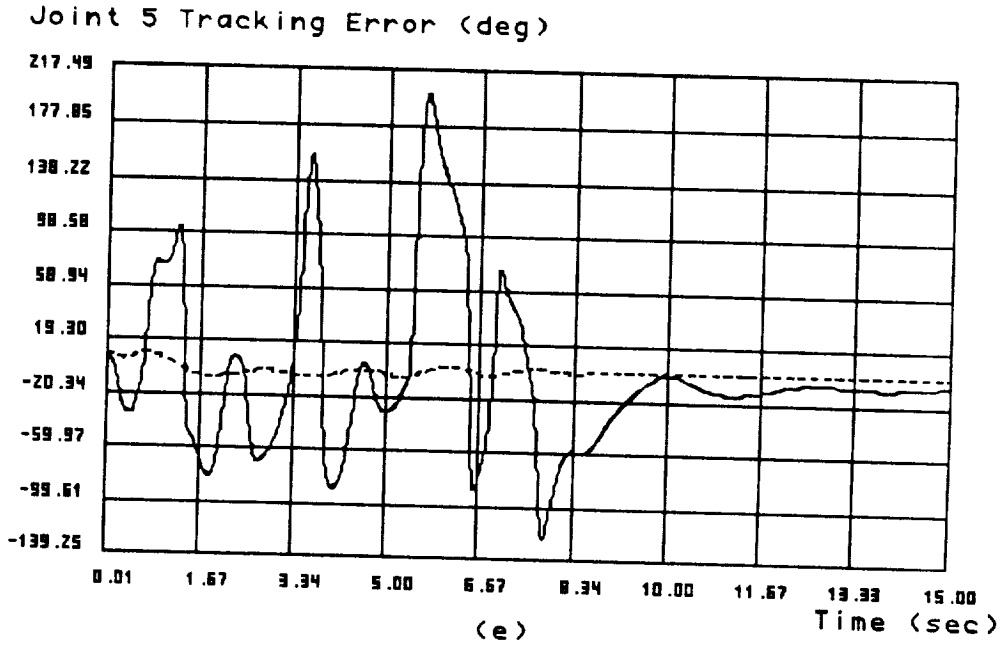


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at 100 Hz Update Rate



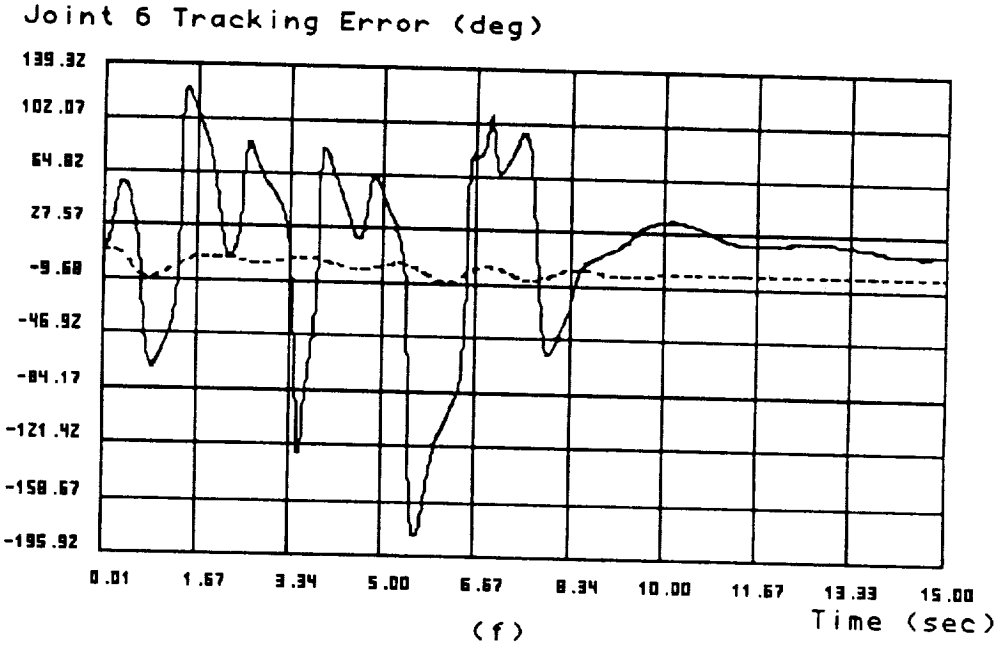
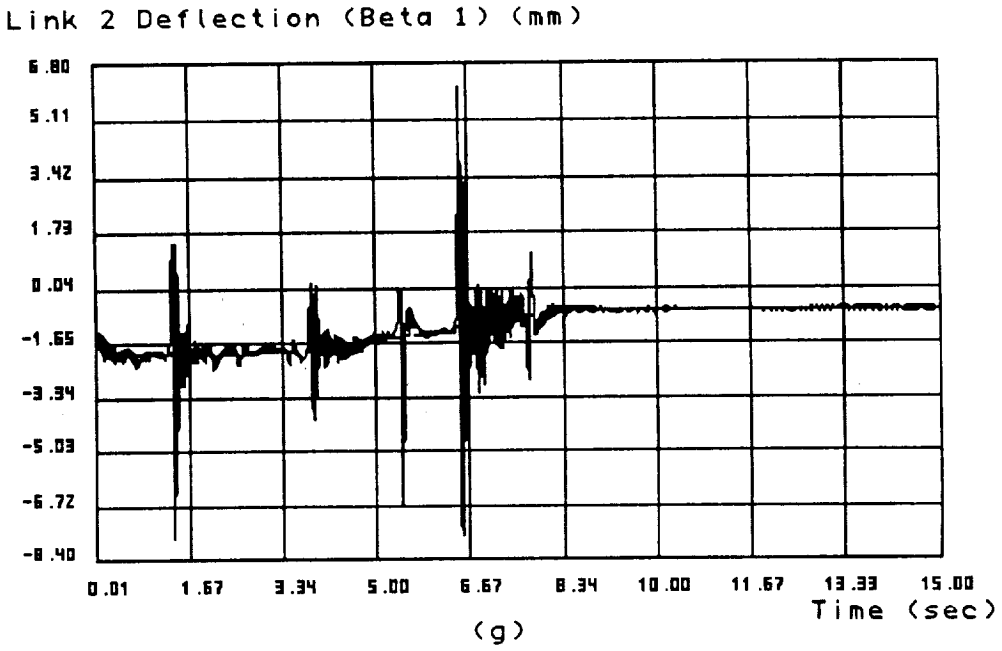


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at 100 Hz Update Rate



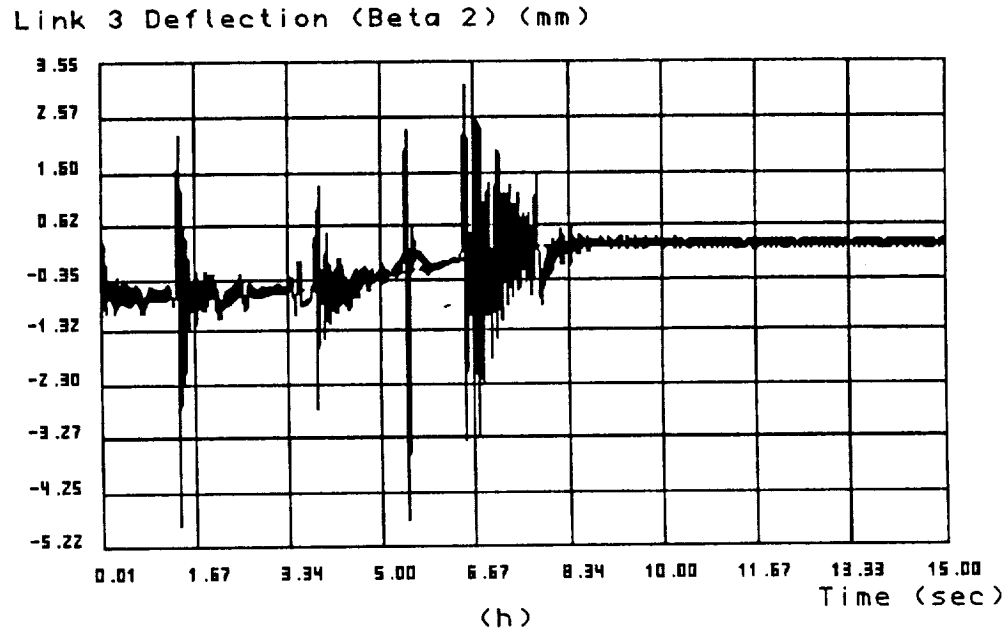
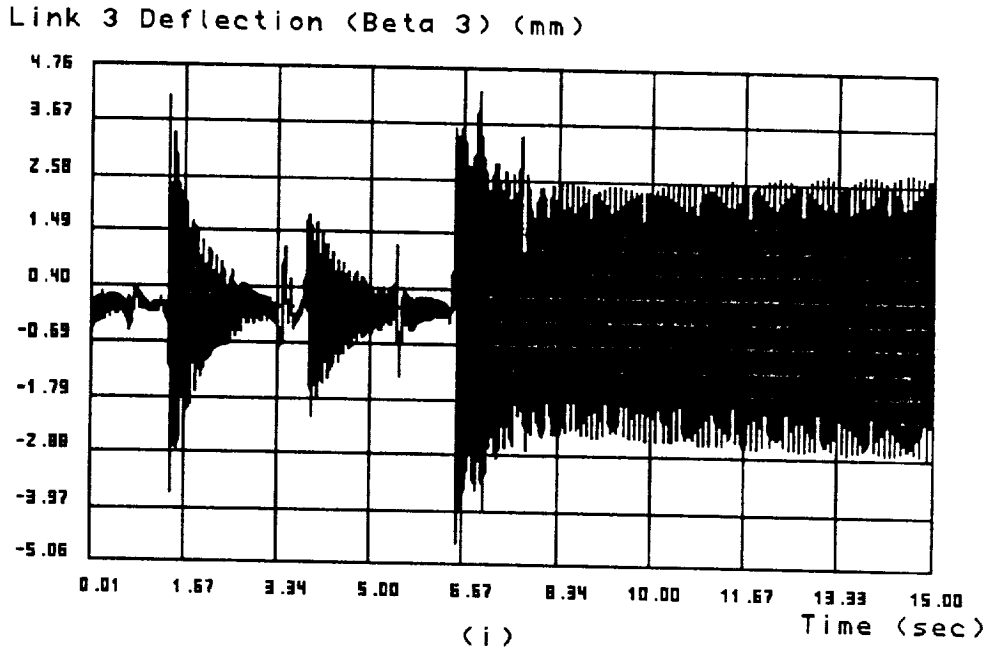


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at 100 Hz Update Rate



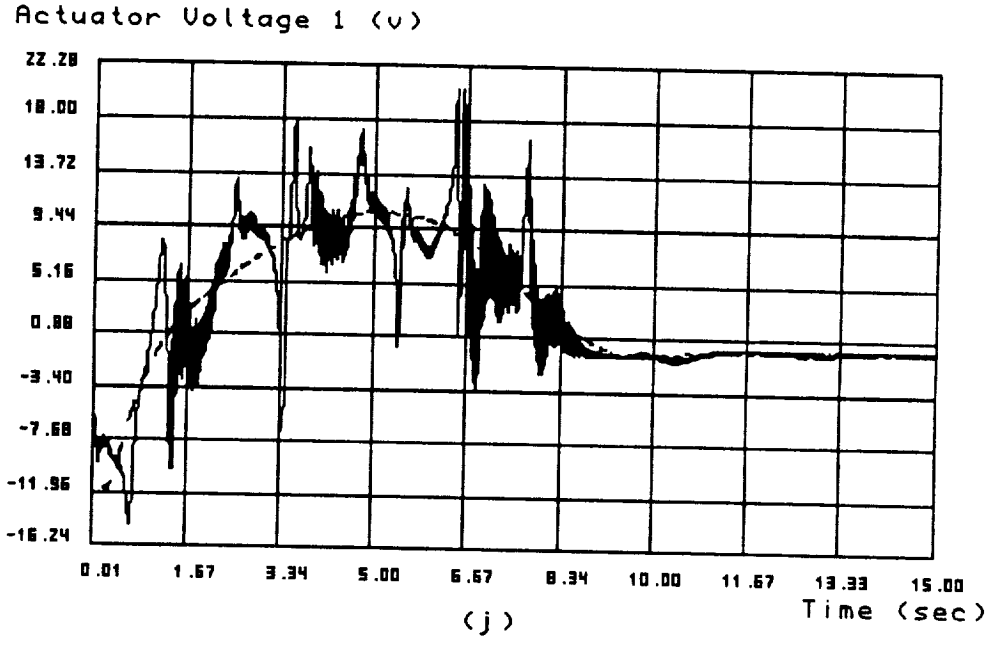
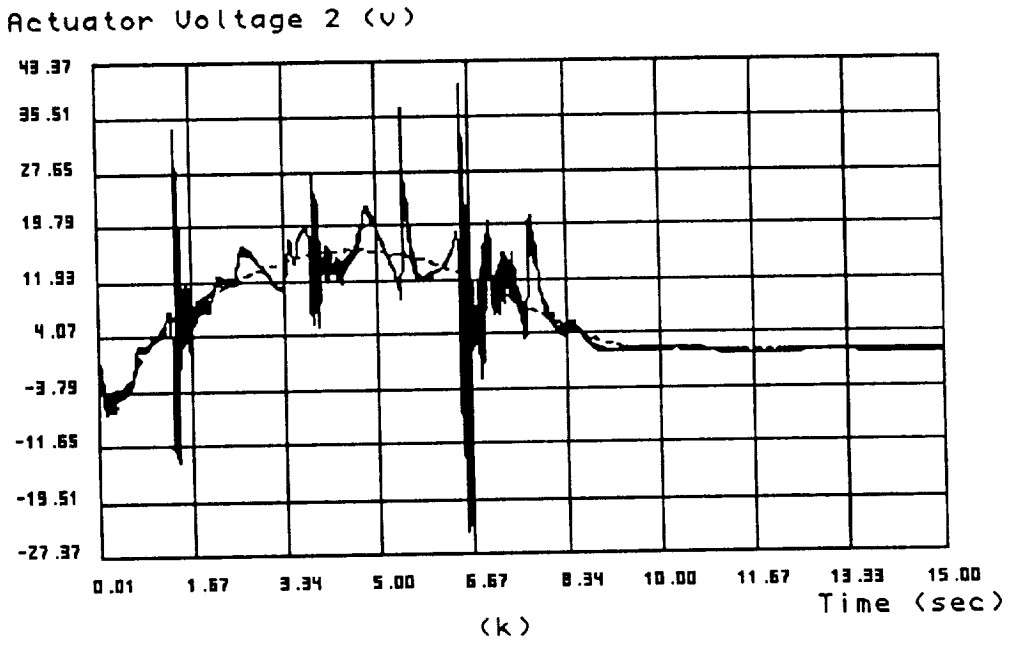


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at 100 Hz Update Rate



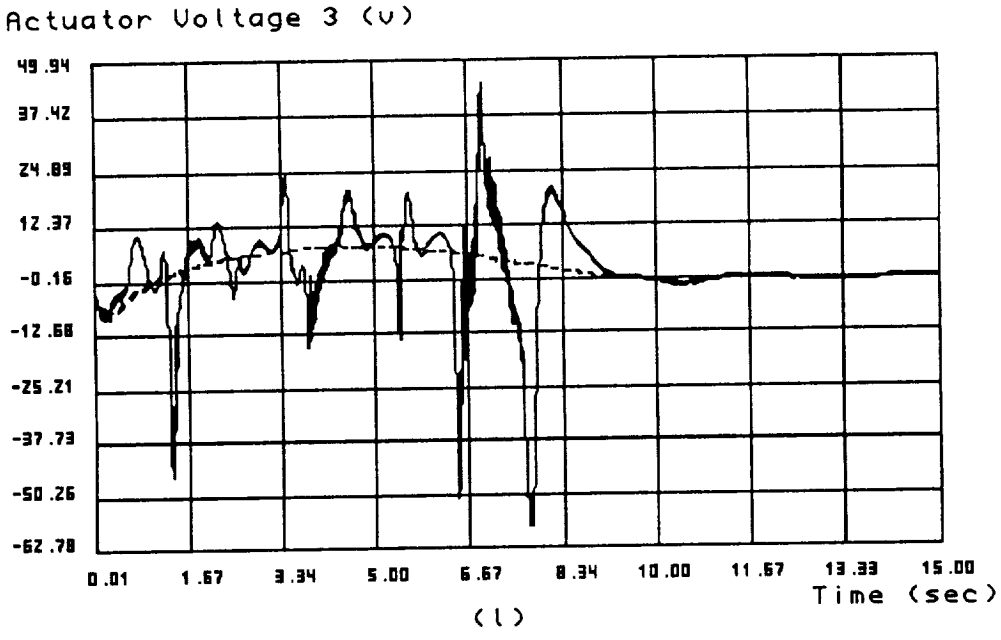


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at 100 Hz Update Rate

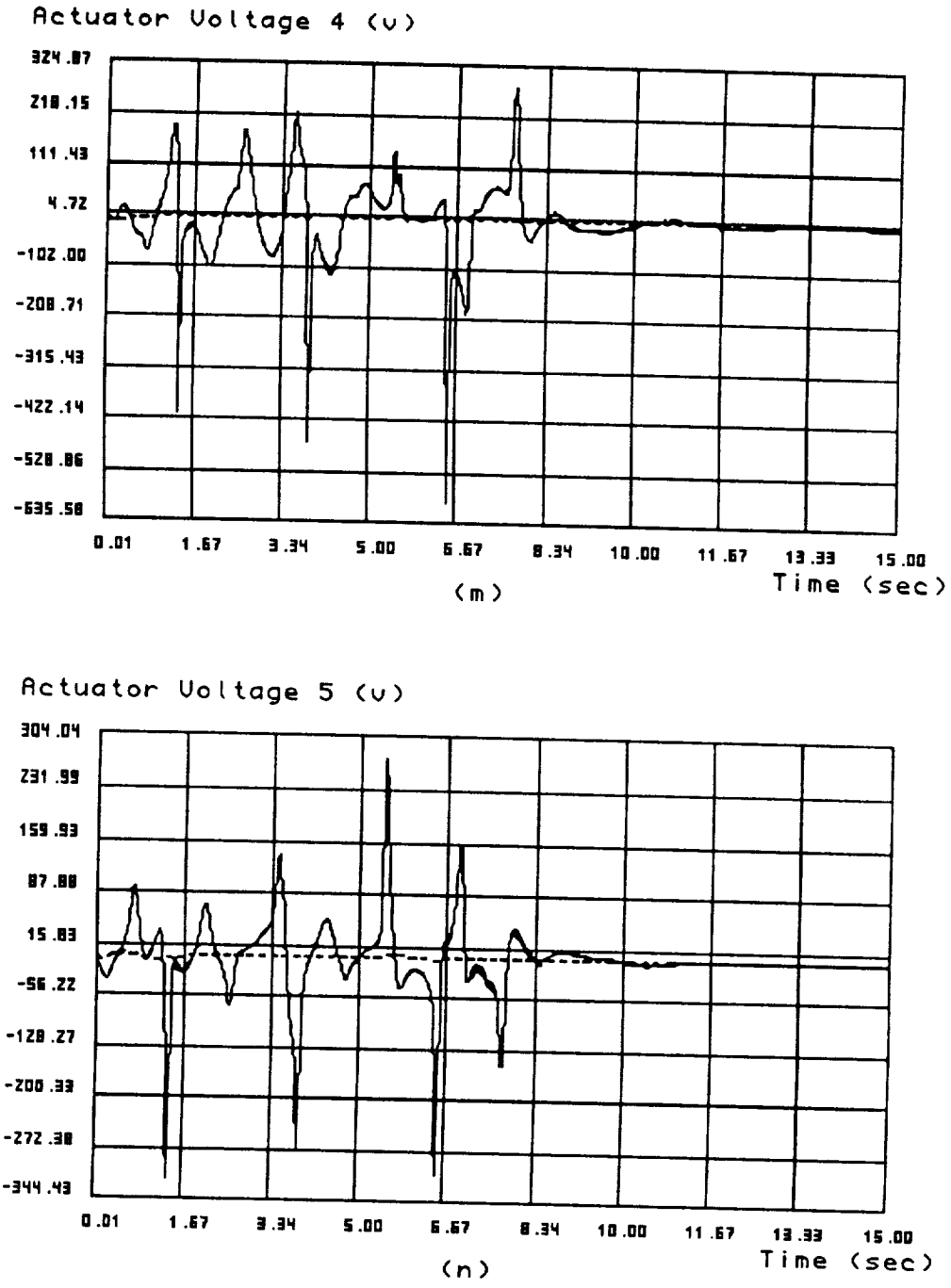


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at 100 Hz Update Rate

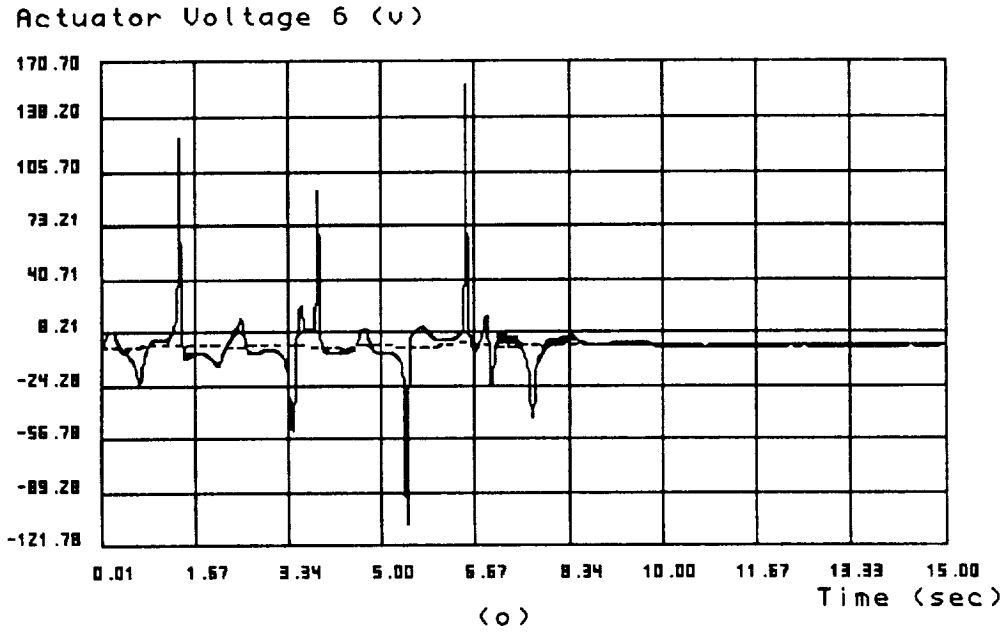


Figure 6.8: Comparison Between Adaptive (Broken Line) and Nonadaptive (Solid Line) Control of a 6-Link Robot at100 Hz Update Rate

6.8 Summary

In this chapter, the Newton-Euler and Lagrange methods are introduced to extract link and payload inertia terms from nonlinear and coupled manipulator dynamics. By doing so, the final dynamic formulation could be expressed as a linear equation in inertial parameters composed of mass, center of mass location, and moment of inertia of individual link and payload. Based on this linear relation, a least square method is introduced to estimate the constant but uncertain inertial parameters. The results could be applied to on-site calibration of lumped manipulator inertia when drift of modeled values is suspected or calibration is required for a new assembled modularized structure. The convergence of the least square method is shown by a Lyapunov type function and a direct integration of estimate for a persistently exciting system. The requirement of persistent excitation to ensure accuracy of estimate is also discussed.

For compliant manipulators carrying an uncertain payload, an adaptive algorithm is introduced to control nominal tracking, vibration elimination, and on-line payload estimation. The adaptive control law is applicable to rigid robotic manipulators as well as lumped and distributed compliant manipulators. Three computer simulations are reported to verify the designed results. The first simulation is conducted on a three-link robot modeled with three lumped link compliances. In this case study, convergence of estimate is impeded due to the limited angular movement of the gripper. To verify this, the second and third simulations are performed on a six-link robot where wrist motion is implemented. The six-link robot is also modeled with three lumped link compliances. With the additional wrist motion, the system is well excited and payload estimates converge to exact values swiftly. The sec-

ond and third simulations have distinct initial estimates, whose purpose is to test adaptive capacity of the proposed controller. The relationship between persistently exciting condition and physical motion is observed from a comparison between both simulations. In both three-link and six-link examples, nominal motion tracking and vibration elimination are satisfactorily executed by the adaptive controller.

In physical implementation, real-time computation delays create a disturbance to the proposed control algorithm which is developed based on continuously updated information. In order to remedy the update delay disturbance, update delay effect on adaptive controller stability is analyzed. According to the analyzed results, a small ρ and termination of estimation once system ceases excitation are recommended to reduce the update delay impact. Case studies are conducted to verify the suggestion. Finally, comparison between adaptive and nonadaptive control laws are illustrated through an example simulation. Several criteria are proposed to compare between adaptive and nonadaptive controllers. It is suggested that different controllers should be tested experimentally over various tasks in order to construct a performance table that assists users to select the best controller for a specified task.

Chapter 7

Summary and Discussions

In this report, we have presented a systematic analysis of compliant manipulators which covers dynamic modeling, property investigation, and both regular and adaptive control algorithms. In the first chapter, we survey and categorize more than ninety recently published reports which are dedicated to building an efficient, lightweight manipulator that has a large payload-to-weight ratio and capable of undergoing high speed and precise operations. Due to nonlinear interactions between robot nominal motion and structural oscillations, the goal of building a general lightweight robot is a difficult and slowly developing task. At this stage, many researchers still conduct primitive experiments on simple compliant structures such as one-link arms in order to gain insight in dynamic modeling and control problems. Additionally, most of the studies are aimed for spatial applications such as the Spatial Shuttle Remote Manipulator System (RMS) which works in a zero-g environment where structural inertia is not the main concern. Therefore, most one-link models are so light that they can not even support their own weights, hence the experiments are limited to horizontal movements to avoid gravity effects. So far, a small number of studies, e.g., [Rivin, et al., 1987] and [Liao, Sung, and Thompson, 1987], are devoted to improve physically industrial manipulator structural design by using composite components or materials to reduce link weight while maintaining good rigidity. Other than

that studies of lightweight industrial robots are still in analytic and numerical simulation stages.

In order to create a valid description of flexibility dynamics, experimental works have been conducted to verify two mostly used compliant models: distributed and lumped parameter models. For example, [Tsujisawa and Book, 1989] have used a distributed parameter model on a two-link robot, and [Behi, 1985] has developed a lumped parameter model for a Cincinnati Milacron T3-776 robot. The former shows that the link lateral deflections could be depicted properly by the first two fundamental modes. However, the first mode amplitude is about ten times of the second mode. Similar results are also indicated by [Yang and Donath, 1988]. Both reports support the assumption used by lumped parameter model that the first mode dominates structural deformation and hence it is the only mode modeled to create simple but reasonably accurate dynamic equations. In order to retain physical reality, the spring stiffnesses used in lumped parameter models are actually evaluated from laboratory experiments as reported by [Sklar, 1988], [Elmaraghy and Johns, 1988], [Behi, 1985], and [Good, Sweet, and Strobel, 1984]. From these experimental results, it is discovered that even in lumped parameter models only finite salient modes could be triggered. Therefore, instead of using seven decoupled vibratory modes (i.e., one joint compliance, three link end-point deflections, and three link end-point twistings) to define lumped compliances of a flexible link, generally, only joint compliance and two principal plane deflections are considered in lumped parameter models. Link lateral deflections are also the major considerations in most of distributed parameter models. Therefore, in this report, we choose robot upper arm and forearm lateral deflections as the major link compliances in our case studies.

Based on availability of payload information, we propose two algorithms to control compliant manipulators: one is built on well-known system parameters, or regular controller, and the other is designed to adapt payload uncertainty, or adaptive controller. Both controllers are built on general system dynamic equations that are common for both distributed and lumped parameter models. Also, in order to maintain controller generality, neither nonlinear terms are neglected nor linearization is used to simplify the controller design. In the regular controller, we test the controller robustness by introducing uncertainty in payload description during the case studies. By choosing stable feedback gains and proper P matrix, the regular controller maintains system stability even under incorrect payload information and initial position errors. However, as payload uncertainty increases and becomes a dominant factor, the regular controller performance degrades. To remedy this problem, an adaptive controller is introduced to undertake payload uncertainty. By using the fact that system dynamics are a linear function of inertial parameters, the uncertain payload is estimated on-line with the assistance of the least square estimation method. One requirement of ensuring estimate convergence is the persistently exciting condition which, as we point out through case studies, is correlated with the system physical motion. That is, if an uncertain parameter is not activated appropriately during a controlled motion, the estimator can not collect enough information to reconstruct the true identity. This observation indicates that for off-line system parameter calibration, laboratory engineers need to test the trajectory carefully in order to excite all parameters which are to be calibrated.

One problem discussed in this report is the effect of update delay on the performance of adaptive controller. In real-time implementation, computation delays create a disturbance to the adaptive controller which is built

on continuously updated signals. Therefore, the effect of update delay on controlled system stability is analyzed, and suggestions are given to eliminate update delay disturbance on controlled system response. Another important issue, a comparison between adaptive and nonadaptive control methodologies, is also demonstrated through example simulations. Of course, a case study can not represent a general comparison between adaptive and nonadaptive controller performances, especially for a nonlinear system where controller parameters such as the PID feedback gains and the P matrix are coupled with system parameters in the final controlled system response. Hence, experiments need to be conducted to demonstrate controller capability under different tasks and uncertainties. We list several criteria to compare experimental results, which will help the user to choose the best controller suited for a given task.

Another way of solving the update delay problem is to design the controller based on a discrete time dynamic model and then to synchronize both computation and control update rates. However, it is difficult to convert nonlinear robotic dynamics from continuous time to discrete time description. Although, Euler's method is often used to approximate the conversion, it is limited to slow varying systems. Hence, one solution is to write system dynamics in terms of an autoregressive and moving average (ARMA) model with uncertain coefficients. Then self-tuning regulator (STR) could be applied to stabilize the uncertain system either by pole placement technique or one-step ahead optimization. In STR, the uncertain ARMA model coefficients are estimated on-line and used in constructing control command. The drawback of STR method is that ARMA model coefficients must vary comparatively slower than the STR update rate, otherwise, STR can not produce accurate estimates to generate the desired control command. Also, STR es-

timates ARMA model coefficients at every update step, so unless a simple relation such as linear proportion exists between payload uncertainty and estimated coefficients, it is difficult to extract payload identity from estimated coefficients, which means that for nonlinear systems, STR could accomplish adaptive control assignment but might not be able to provide exact uncertainty information.

In both regular and adaptive control approaches, we encounter one problem that the Lyapunov stability is not ensured when all nominal error states become zero but vibratory error state remains active. This unique character of compliant manipulators is due to dimensional mismatch between the number of available actuators and that of modeled degrees of freedom. Since passive vibratory modes are added to compliant system dynamics, their elimination relies on the regulation of nominal motion, which means that controller has to transmit damping action through nominal joints, so nominal joints have the responsibility of eliminating structural vibration instead of just trajectory tracking. In the controller design, it is possible to disturb nominal tracking precision deliberately in order to obtain the Lyapunov stability. However, that is not an effective tradeoff, therefore, in both controllers, we propose methodologies to reduce the size of the spherical ball in error space where the Lyapunov stability is uncertain. In doing so, residual oscillation is left to be dampened passively by structural damping. In case residual oscillation scale becomes intolerable, then a second phase controller could be employed to reduce residual oscillations. Now, the regular or adaptive controller designed before acts as the first phase controller that concentrates on trajectory tracking and vibration elimination. Once the nominal joints get closer to the terminal point and the error states approach the spherical ball that the Lyapunov stability is uncertain, then the second phase is switched

on. The second phase controller could be built on different control criteria, for example, one such second phase controller could be the optimal controller presented below.

From our earlier case study experience, it is observed that in the vicinity of terminal steady state, the velocity states remain relatively small. Therefore, it is assumed that all coupling terms are negligible and gravity effect is compensated at terminal state, then, in the vicinity of the terminal point, the flexibility dynamics in Equation 4.9 could be approximated by the following linear time-invariant form

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} A_1 \\ C \end{bmatrix} u + \begin{bmatrix} C^T \\ A_2 \end{bmatrix} K\beta \tag{7.1}$$

where $A_1 = A_1(\theta_r) \in \mathcal{R}^{n_\theta \times n_\theta}$, $C = C(\theta_r) \in \mathcal{R}^{n_\beta \times n_\theta}$, $A_2 = A_2(\theta_r) \in \mathcal{R}^{n_\beta \times n_\beta}$, and $\theta_r = \theta(t_f)$ is the terminal state of nominal joint in which t_f is the final time. In the above equation, A_1, A_2, and C are constant submatrices evaluated at the terminal state with no vibrations. Defining

$$q = \begin{bmatrix} \theta \\ \dot{\theta} \\ \beta \\ \dot{\beta} \end{bmatrix} \in \mathcal{R}^{2(n_\theta + n_\beta)} \; ; \; q_r = q(t_f) = \begin{bmatrix} \theta_r \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{7.2}$$

Equation 7.1 could be expressed as

$$\begin{aligned} \dot{q} &= \begin{bmatrix} 0 & \mathcal{I}' & 0 & 0 \\ 0 & 0 & -C^T K & 0 \\ 0 & 0 & 0 & \mathcal{I}'' \\ 0 & 0 & -A_2 K & 0 \end{bmatrix} q + \begin{bmatrix} 0 \\ A_1 \\ 0 \\ C \end{bmatrix} u \\ &\stackrel{\text{def}}{=} Aq + Bu \end{aligned} \tag{7.3}$$

where A is a $2(n_\theta + n_\beta) \times 2(n_\theta + n_\beta)$ matrix, B is a $2(n_\theta + n_\beta) \times n_\theta$ matrix, $\mathcal{I}'$ is an $n_\theta \times n_\theta$ identity matrix, and $\mathcal{I}''$ is an $n_\beta \times n_\beta$ identity matrix. Now, the

goal is to control q to reach q_r at $t = t_f$ and minimize a cost function given by

$$J = \int_{t_0}^{t_f} \left\{ (q - q_r)^T S (q - q_r) + u^T W u \right\} dt \tag{7.4}$$

where t_0 is the time optimal control starts, and S and W are two positive definite, diagonal matrices with $S = S^T \in \mathcal{R}^{2(n_\theta + n_\beta) \times 2(n_\theta + n_\beta)}$ and $W = W^T \in \mathcal{R}^{n_\theta \times n_\theta}$. According to [Bryson and Ho, 1975], Equation 7.4 is an optimal control problem with fixed terminal time and specified state variables, and one optimal solution is given by

$$u = W^{-1}(B^T \lambda + B^T \Phi^T \nu) \tag{7.5}$$

where λ is the Lagrange multiplier defined by

$$\begin{aligned} \dot{\lambda} &= -A^T \lambda - 2Sq - 2Sq_r \\ \lambda(t_0) &= 0\,;\ \lambda \in \mathcal{R}^{2(n_\theta + n_\beta)} \end{aligned} \tag{7.6}$$

Because q is not specified at $t = t_0$, $\lambda(t_0) = 0$ is chosen in the above equation. Also, the Φ in Equation 7.5 is given by

$$\begin{aligned} \dot{\Phi}(t, t_0) &= -\Phi(t, t_0) A \\ \Phi(t_0, t_0) &= \mathcal{I} \in \mathcal{R}^{2(n_\theta + n_\beta) \times 2(n_\theta + n_\beta)} \end{aligned} \tag{7.7}$$

which is the transition matrix used in formulating the solution of q given by

$$q(t) = \Phi(t, t_0) q(t_0) + \int_{t_0}^{t} \Phi(t, s) B u(s) ds\ ;\ t_0 \le t \le t_f \tag{7.8}$$

Finally, the ν in Equation 7.5 is defined as

$$\begin{aligned} \nu &= -W_{ct}^{-1} \varphi \in \mathcal{R}^{2(n_\theta + n_\beta)} \\ W_{ct} &= \int_{t_0}^{t_f} \Phi B B^T \Phi^T dt \in \mathcal{R}^{2(n_\theta + n_\beta) \times 2(n_\theta + n_\beta)} \\ \varphi &= \int_{t_0}^{t_f} \Phi B (W u + B^T \lambda) dt \in \mathcal{R}^{2(n_\theta + n_\beta)} \end{aligned} \tag{7.9}$$

where W_{ct} is called controllability grammian and is nonsingular for a controllable system in Equation 7.3. Since Equations 7.5 and 7.9 need to be solved iteratively, an off-line evaluated ν value could be used in computing on-line the u given by Equation 7.5. Actually, t_f could be free in order to obtain a minimum settling time control, yet, that will introduce another control parameter needed to be decided iteratively. Therefore, fixed time optimal control is chosen here.

Additionally, in this report we introduce a special characteristic of compliant manipulators that has seldom been noticed. This is the inaccessibility problem of vibratory modes. According to the analysis in Chapter 4, manipulator configuration affects actively damping of structural oscillations. It is suggested that the inaccessible positions of a compliant manipulator should be identified before trajectory planning so that undesired working positions can be avoided. Also, since the first mode dominates structural deflections, lumped parameter model will be an efficient and effective approach to study the inaccessibility problem.

Due to the complexity of compliant manipulators, this report only investigates some of the dynamic and control problems. There are some issues of compliant manipulators that need further study, which are listed below:

- development of lightweight robot designs for industrial applications
- comparison between lumped and distributed parameter models
- kinematic interpretation of vibratory mode inaccessibility
- interconnection between the inaccessibility of the first mode and that of higher order modes

- compensation of robot structural deflections by adjusting joint motions
- experimental verification of regular and adaptive control algorithms
- improvement of regular and adaptive controller performance by introducing the second phase controller
- criteria for the selection of regular and adaptive controllers
- implementation of micromanipulators to perform on-line compensation of structural deflection
- incorporation of adaptive control with learning control to build an accurate system model for systems containing uncertain parameters

These research topics require considerable effort, but they will certainly make lightweight manipulators become much more precise and efficient machines.

BIBLIOGRAPHY

[1] Anderson, G.L., "Stability of a Manipulator with Resilient Joints," *Journal of Sound and Vibration,* Vol. 101, No. 4, 1985, pp. 463–480.

[2] Asada, H., Ma, Z.D., and Tokumaru, H., "Inverse Dynamics of Flexible Robot Arms for Trajectory Control," *Modeling and Control of Robotic Manipulators and Manufacturing Processes,* ASME Winter Annual Meeting, Boston, Massachusetts, December 13–18, 1987, pp. 329–336.

[3] Astrom, K.J. and Wittenmark, B., *Adaptive Control,* Addison-Wesley Publishing Company, 1989.

[4] Bayo, E., "Computed Torque for the Position Control of Open-Chain Flexible Robots," *1988 IEEE International Conference on Robotics and Automation,* April 24–29, Philadelphia, Pennsylvania, pp. 316–321.

[5] Bayo, E. and Moulin, H., "An Efficient Computation of the Inverse Dynamics of Flexible Manipulators in the Time Domain," *1989 IEEE International Conference on Robotics and Automation,* May 14–19, Scottsdale, Arizona, pp. 710–715.

[6] Bellezza, F., Lanari, L., and Ulivi, G., "Exact Modeling of the Flexible Slewing Link," *1990 IEEE International Conference on Robotics and Automation,* May 13–18, Cincinnati, Ohio, pp. 734–739.

[7] Benati, M. and Morro, A., "Dynamics of Chain of Flexible Links," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 110, No. 4, December 1988, pp. 410–415.

[8] Biswas, S.K. and Klafter, R.D., "Dynamic Modeling and Optimal Control of Flexible Robotic Manipulators," *1988 IEEE International Conference on Robotics and Automation,* April 24–29, Philadelphia, Pennsylvania, pp. 15–20.

[9] Book, W.J., "Recursive Lagrangian Dynamics of Flexible Manipulator Arms," *The International Journal of Robotics Research,* Vol. 3, No. 3, Fall 1984, pp. 87–101.

[10] Book, W.J., "Analysis of Massless Elastic Chain with Servo Controlled Joint," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 101, No. 3, September 1979, pp 187–192.

[11] Book, W.J. and Cetinkunt, S., and Woodruff, G.W., "Near Optimum Control of Flexible Robot Arms on Fixed Path," *Proc. of 24th IEEE Conference on Decision and Control,* Fort Lauderdale, Florida, December 1985, pp. 1522–1528.

[12] Bryson, A.E. and Ho, Y.-C., *Applied Optimal Control: Optimization, Estimation, and Control,* Hemisphere Publishing Corperation, Washington, 1975.

[13] Cannon, R.H. and Schmitz, E., "Initial Experiments on the End-Point Control of a Flexible One-Link Robot," *The International Journal of Robotics Research,* Vol. 3, No. 3, Fall 1984, pp. 62–75.

[14] Cetinkunt, S. and Wu, S., "Tip Position Control of a Flexible One Arm Robot with Predictive Adaptive Output Feedback Implemented with Lattice Filter Parameter Identification," *1990 IEEE International Conference on Robotics and Automation,* May 13–18, Cincinnati, Ohio, pp. 1620–1625.

[15] Chalhoub, N.G. and Ulsoy, A.G., "Control of a Flexible Robot Arm: Experimental and Theoretical Results," *Modeling and Control of Robotic Manipulators and Manufacturing Processes,* ASME Winter Annual Meeting, Boston, Massachusetts, December 13–18, 1987, pp. 287–295.

[16] Chassiakos, A.G. and Bekey, G.A., "On the Modeling and Control of a Flexible Manipulator Arm by Point Actuators," *Proc. of 25th IEEE Conference on Decision and Control,* Athens, Greece, December 1986, pp. 1145–1150.

[17] Chen, J.-S. and Menq, C.-H., "Experiments on the Payload-Adaptation of a Flexible One-link Manipulator with Unknown Payload," *1990 IEEE International Conference on Robotics and Automation,* May 13–18, Cincinnati, Ohio, pp. 1614–1619.

[18] Chiou, B.C. and Shahinpoor, M., "Stability Considerations for a Two-Link Force-Controlled Flexible Manipulator," *1990 IEEE International Conference on Robotics and Automation,* May 13–18, Cincinnati, Ohio, pp. 728–733.

[19] Choi, S.B., Thompson, B.S., and Gandhi, M.V., "Modeling and Control of a Single-Link Flexible Manipulator Featuring a Graphite-Epoxy Composite Arm," *1990 IEEE International Conference on Robotics and Automation,* May 13–18, Cincinnati, Ohio, pp. 1450–1455.

[20] Cleghorn, W.L., Tabarrok, B., and Fenton, R.G., "Critical Running Speeds and Stability of High-Speed Flexible Mechanisms," *Mechanism and Machine Theory,* Vol. 19, No. 3, 1984, pp. 307–317.

[21] Dado, M. and Soni, A.H., "A Generalized Approach for Forward and Inverse Dynamics of Elastic Manipulators," *1986 IEEE International Conference on Robotics and Automation,* April 7–10, San Francisco, California, pp. 359–364.

[22] De Luca, A., Isidori, A., and Nicolo, F., "Control of Robot Arm with Elastic Joints Via Nonlinear Dynamic Feedback," *Proc. of 24th IEEE Conference on Decision and Control,* Fort Lauderdale, Florida, December 1985, pp. 1671–1679.

[23] De Luca, A., Lucibello, P., and Ulivi, G., "Inversion Techniques for Trajectory Control of Flexible Robot Arms," *Journal of Robotic System,* Vol. 6, No. 4, 1989, pp. 325–344.

[24] De Maria, G. and Siciliano, B., "A Multilayer Approach to Control of a Flexible Arm," *1987 IEEE International Conference on Robotics and Automation,* March 31–April 3, Raleigh, North Carolina, pp. 774–778.

[25] Driels, M.R. and Pathre, U.S., "Generalized Joint Model for Robot Manipulator Kinematic Calibration and Compensation," *Journal of Robotic Systems,* Vol. 4, No. 1, 1987, pp. 77–114.

[26] Elmaraghy, H.A. and Johns, B., "An Investigation Into the Compliance of SCARA Robots. Part I: Analytical Model," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 110, No. 1, March 1988, pp. 18–22.

[27] Elmaraghy, H.A. and Johns, B., "An Investigation Into the Compliance of SCARA Robots. Part II: Experimental and Numerical Validation," *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 110, No. 1, March 1988, pp. 23–30.

[28] Feliu, V., Rattan, K.S., and Brown, H.B., "Adaptive Control of a Single-Link Flexible Manipulator in the Presence of Joint Friction and Load Changes," *1989 IEEE International Conference on Robotics and Automation*, May 14–19, Scottsdale, Arizona, pp. 1036–1041.

[29] Fresonke, D.A., Hernandez, E., and Tesar, D., "Deflection Prediction for Serial Manipulators," *1988 IEEE International Conference on Robotics and Automation*, April 24–29, Philadelphia, Pennsylvania, pp. 482–487.

[30] Good, M.C., Sweet, L.M., and Strobel, K.L., "Dynamic Models for Control System Design of Integrated Robots and Drive Systems," *Sensors and Controls for Automated Manufacturing and Robotics*, ASME Winter Annual Meeting, New Orleans, Louisiana, December 9–14, 1984, pp. 253–269.

[31] Graves, P.L., *The Effect of Inertial Coupling in the Dynamics and Control of Flexible Robotic Manipulators*, Master Thesis, University of Texas at Austin, August 1988.

[32] Gupta, S., "Effect of Link Compliance on Positioning Accuracy and Kinematic Geometry in Robots," *Modeling and Control of Robotic Manipulators and Manufacturing Processes*, ASME Winter Annual Meeting, Boston, Massachusetts, December 13–18, 1987, pp. 315–322.

[33] Han, C.S., Traver, A.E., and Tesar, D., "Using CAD/CAM in the Design of a Stewart Platform Type of Micromanipulator," *Proc. ASME 1989*

International Conference on Computers in Engineering, Anaheim, CA, pp. 251–256, July 1989.

[34] Hastings, G.G. and Book, W.J., "Reconstruction and Robust Reduced-Order Observation of Flexible Variables," *Robotics: Theory and Applications,* ASME Winter Annual Meeting, December 1986, pp. 11-16.

[35] Hastings, G.G. and Book, W.J., "Verification of a Linear Dynamic Model for Flexible Robotic Manipulators," *1986 IEEE International Conference on Robotics and Automation,* April 7–10, San Francisco, California, pp. 1024–1029.

[36] Hastings, G.G., Dorsey, J.F., and Book, W.J., "Application of Balanced Realizations to Estimate Model Order Requirements for Flexible Manipulators," *1989 IEEE International Conference on Robotics and Automation,* May 14–19, Scottsdale, Arizona, pp.

[37] Hastings, G.G. and Ravishankar, B.N., "An Experimental System for Investigation of Flexible Links Experiencing General Motion," *Proc. of 1989 American Control Conference*, Pittsburgh, PA, June 1989, pp. 1003–1008.

[38] Henrichfreise, H., "The Control of an Elastic Manipulation Device Using DSP," *Proc. of 1988 American Control Conference*, Atlanta, GA, June 1988.

[39] Hernandez, E., *Efficient Algorithms and Real-Time Software for Quasi-Static Manipulator Deflection Modeling,* Master Thesis, University of Texas at Austin, May 1989.

[40] Huang, Y. and Lee, C.S.G., "Generalization of Newton-Euler Formulation of Dynamic Equations to Nonrigid Manipulators," *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 110, No. 3, 1988, pp. 308–315.

[41] Hudgens, J. and Tesar, D., "A Fully Parallel Six Degree-of-Freedom Micromanipulator: Kinematic Analysis and Dynamic Model," *Proc. 20th ASME Mechanisms Conference,* DE Vol. 15–3, September 1988, pp. 29–37.

[42] Hudgins, K.P., *Model Driven Flexible System Adaptive Control of a Swing-Arm Manipulator,* Master Thesis, University of Texas at Austin, May 1982.

[43] Hughes, P.C., "Dynamics of a Chain of Flexible Bodies," *The Journal of the Astronautical Sciences,* Vol. XXVII, No. 4, Oct.–Dec., 1979, 359–380.

[44] Huston, R.L., "Flexibility Effects in Multibody System Dynamics," *Mechanical Research Communications,* Vol. 7, No. 4, 1980, pp. 261–268.

[45] Kane, T.R. and Levinson, D.A., *Dynamics: Theory and Applications,* McGraw-Hill Book Company, New York.

[46] Kane, T.R., Ryan, R.R., and Banerjee, A.K., "Dynamics of a Cantilever Beam Attached to a Moving Base," *AIAA Journal of Guidance,* Vol. 10, No. 2, March–April 1987, pp. 139–151.

[47] Kelly, F.A. and Huston, R.L., "Modeling of Flexibility Effects in Robot Arm," *IEEE Joint Automatic Control Conference,* June 17-19, 1981, Charlottesville, VA, pp. WP-2C.

[48] Khorasani, K. and Kokotovic, P.V., "Feedback Linearization of a Flexible Manipulator Near Its Rigid Body Manifold," *Systems and Control Letters,* Vol. 6, 1985, pp. 187–192.

[49] Khorasani, K. and Spong, M.W. "Invariant Manifolds and Their Application to Robot Manipulators with Flexible Joints," *1985 IEEE International Conference on Robotics and Automation,* March 25–28, St. Louis, Missouri, pp. 978–983.

[50] Khorrami, F. and Ozguner, U., "Perturbation Methods in Control of Flexible Link Manipulators," *1988 IEEE International Conference on Robotics and Automation,* April 24–29, Philadelphia, Pennsylvania, pp. 310–315.

[51] Koivo, A.J. and Lee, K.S., "Self-Tuning Control of Planar Two-Link Manipulator with Non-Rigid Arm," *1989 IEEE International Conference on Robotics and Automation,* May 14–19, Scottsdale, Arizona, pp. 1030–1035.

[52] Korolov, V.V. and Chen, Y.H., "Controller Design Robust to Frequency Variation in a One-Link FLexible Robot Arm," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 111, No. 1, March 1989, pp. 9–14.

[53] Kotnik, P.T., Yurkovich, S., and Ozguner, U. "Acceleration Feedback for Control of a Flexible Manipulator Arm," *Journal of Robotic Systems,* Vol. 5, No. 3, pp. 181 – 196, 1988.

[54] Krishnan, H. and Vidyasagar, M., "Control of a Single-Link Flexible Beam Using a Hankel-Norm-Based Reduced Order Model," *1988*

IEEE International Conference on Robotics and Automation, April 24–29, Philadelphia, Pennsylvania, pp. 9–14.

[55] Landau, Y.D., "Adaptive Control: The Model Reference Approach," Marcel Dekker, Inc., New York, 1979.

[56] Lee, H. and Castelazo, I.A., "Nonlinear Feedback Control of a Flexible Robot Arm," *Modeling and Control of Robotic Manipulators and Manufacturing Processes,* ASME Winter Annual Meeting, Boston, Massachusetts, December 13–18, 1987, pp. 307–314.

[57] Lee, J.W., Huggins, J.D., and Book, W.J., "Experimental Verification of a Large Flexible Manipulator," *Proc. of 1988 American Control Conference*, Atlanta, GA, June 1988, pp. 1021–1028.

[58] Li, W. and Slotine, J.-J.E., "Parameter Estimation Strategies for Robotic Applications," *Modeling and Control of Robotic Manipulators and Manufacturing Processes,* ASME Winter Annual Meeting, Boston, Massachusetts, December 13–18, 1987, pp. 213–218.

[59] Liao, D.X., Sung, C.K., and Thompson, B.S., "The Design of Flexible Robotic Manipulators with Optimal Arm Geometries Fabricated from Composite Laminates with Optimal Material Properties," *The International Journal of Robotics Research,* Vol. 6, No. 3, Fall 1987, pp. 116–130.

[60] Lin, S.H., Tosunoglu, S., and Tesar, D., a, "A Controller Design for Compliant Manipulators Modeled with Elastic Links and Joints," accepted for presentation, *29'th IEEE Conf. on Decision and Control,* Honolulu, Hawaii, December 5–7, 1990.

[61] Lin, S.H., Tosunoglu, S., and Tesar, D., b, "Control of a Six-Degree-of-Freedom Industrial Manipulator Modeled with Seven Link and Joint Flexibility Components," *1990 American Control Conference,* San Diego, California, May 23–25, 1990, Vol. 1, pp. 712–717.

[62] Lin, S.H., Tosunoglu, S., and Tesar, D., "Control of Multi-Link Robotic Manipulators with Compliant Joints," *Robotics Research,* 1989 ASME Winter Annual Meeting, San Francisco, CA, DSC-Vol. 14, pp. 299–307.

[63] Low, K.H., "A Systematic Formulation of Dynamic Equations for Robot Manipulators with Elastic Links," *Journal of Robotic System,* Vol. 4, No. 3, 1987, pp. 435–456.

[64] Low, K.H., "Solution Schemes for the System Equations of Flexible Robots," *Journal of Robotic System,* Vol. 6, No. 4, 1989, pp. 383–405.

[65] Low, K.H. and Vidyasagar, M., "A Lagrangian Formulation of the Dynamic Model for Flexible Manipulator Systems," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 110, No. 2, June 1988, pp. 175–181.

[66] Lowen, G.G. and Chassapis, C., "The Elastic Behavior of Linkage: An Update," *Mechanism and Machine Theory,* Vol. 21, No. 1, 1986, pp. 33–42.

[67] Lowen, G.G. and Jandrasits, W.G., "Survey of Investigations into the Dynamic Behavior of Mechanisms Containing Links with Distributed Mass and Elasticity," *Mechanism and Machine Theory,* Vol. 7, 1972, pp. 3–17.

[68] Marino, R. and Nicosia, S. "On the Control of Robots with Elastic Joints," *Proc. of 1985 American Control Conference*, Boston, MA, June 1985, pp. 69–70.

[69] Marino, R. and Spong, M.W., "Nonlinear Control Techniques for Flexible Joint Manipulators: A Single Link Case Study" *1986 IEEE International Conference on Robotics and Automation,* April 7–10, San Francisco, California, pp. 1030–1036.

[70] Markus, L. and Lee, E. B., "On the Existence of Optimal Control," Trans. of the ASME Journal of Basic Engineering, Vol. 84, Series D, No. 1, March 1962, pp. 13 – 22.

[71] Martin, H.C., *Introduction to Matrix Methods of Structural Analysis,* McGraw-Hill Book Company, New York.

[72] Matsuno, F., Fukushima, S., Kiyohara, M., and Sakawa, Y., "Feedback Control of a Flexible Manipulator with a Parallel Drive Mechanism," *The International Journal of Robotic Research,* Vol. 6, No. 4, Winter 1987, pp. 76–84.

[73] Meckl, P.H. and Seering, W.P., "Reducing Residual Vibration in Systems With Time-Varying Resonances," *Proc. International Conference on Robotics and Automation*, Raleigh, NC, March 1987, pp. 1690 –1695.

[74] Meckl, P.H. and Seering, W.P. "Feedforward Control Techniques to Achieve Fast Settling Time in Robots," *Proc. of 1986 American Control Conference*, Seattle, WA, pp. 1913 – 1918, June 1986.

[75] Meckl, P. and Seering, W. "Active Damping in a Three-Axes Robotic Manipulator," *Journal of Vibration, Acoustics, Stress, and Reliability in*

Design, Vol. 107, pp. 38 – 46, January 1985.

[76] Menq, C.-H. and Chen, J.-S., "Dynamic Modeling and Payload-Adaptive Control of a Flexible Manipulator," *1988 IEEE International Conference on Robotics and Automation,* April 24–29, Philadelphia, Pennsylvania, pp. 488–493.

[77] Meirovitch, L., *Computational Methods in Structural Dynamics,* Sijthoff and Noordhoff, Maryland, 1980.

[78] Meirovitch, L., *Analytical Methods in Vibrations,* Collier-Macmillan Limited, London, 1967.

[79] Naganathan, G. and Soni, A.H., "Non-Linear Flexibility Studies for Spatial Manipulators," *1986 IEEE International Conference on Robotics and Automation,* April 7–10, San Francisco, California, pp. 373–378.

[80] Naganathan, G. and Soni, A.H., "Coupling Effects of Kinematics and Flexibility in Manipulators," *The International Journal of Robotics Research,* Vol. 6, No. 1, Spring 1987, pp. 75–84.

[81] Nathan, P.J. and Singh, S.N., "Variable Structure Control of a Robotic Arm with Flexible Links," *1989 IEEE International Conference on Robotics and Automation,* May 14–19, Scottsdale, Arizona, pp. 882–887.

[82] Nelson, W. and Mitra, D., "Load Estimation and Load-Adaptive Optimal Control of a Flexible Robot Arm," *1986 IEEE International Conference on Robotics and Automation,* April 7–10, San Francisco, California, pp. 206–211.

[83] Nemir, D., Koivo, A.J., and Kashyap, R.L., "Control of Gripper Position of a Compliant Link Using Strain Gauge Measurement," *Proc. of 25th*

IEEE Conference on Decision and Control, Athens, Greece, December 1986, pp. 1140-1144.

[84] Nicosia, S., Tomei, P., and Tornambe, A., "Hamiltonian Description and Dynamic Control of Flexible Robots," *Journal of Robotic Systems*, Vol. 6, No. 4, 1989, pp. 345–361.

[85] Nicosia, S., Tomei, P., and Tornambe, A., "Dynamic Modeling of Flexible Robotic Manipulators," *1986 IEEE International Conference on Robotics and Automation*, April 7–10, San Francisco, California, pp. 365–372.

[86] Oliver, J.H., Wysocki, D.A., and Thompson, B.S., "The Synthesis of Flexible Linkages by Balancing the Tracer Point Quasi-Static Deflections Using Microprocessor and Advanced Materials Technologies," *Mechanism and Machine Theory*, Vol. 20, No. 2, 1985, pp. 103–114.

[87] Oosting, K. and Dickerson, S.L., "Simulation of a High-Speed Lightweight Arm," *1988 IEEE International Conference on Robotics and Automation*, April 24–29, Philadelphia, Pennsylvania, pp. 494–496.

[88] Pal, S., Stephanou, H.E., and Cook, G., "Optimal Control of a Single Link Flexible Manipulator," *1988 IEEE International Conference on Robotics and Automation*, April 24–29, Philadelphia, Pennsylvania, pp. 171-175.

[89] Pfeiffer, F., "A Feedforward Decoupling Concept for the Control of Elastic Robots," *Journal of Robotic Systems*, Vol. 6, No. 4, 1989, pp. 407–416.

[90] Pfeiffer, F. and Gebler, B., "A Multistage-Approach to the Dynamics and Control of Elastic Robots," *1988 IEEE International Conference on*

Robotics and Automation, April 24–29, Philadelphia, Pennsylvania, pp. 2–8.

[91] Rakhsha, F. and Goldenberg, A.A., "Dynamics Modeling of a Single-Link Flexible Robot," *1985 IEEE International Conference on Robotics and Automation,* March 25–28, St. Louis, Missouri, pp. 984–989.

[92] Rivin, E.I., "Effective Rigidity of Robot Structure: Analysis and Enhancement," *Proc. of 1985 American Control Conference*, Boston, MA, June 1985, pp. 381–382.

[93] Rivin, E.I., Holbrook, R., Bhatt, S., and Bhattacharyya, A., "A High Stiffness/Low Inertia Revolute Link for Robotic Manipulators," *Modeling and Control of Robotic Manipulators and Manufacturing Process,* ASME Winter Annual Meeting, December 13–18, 1987, pp. 253–259.

[94] Rivin, E.I., Zeid, A., and Rastgu-Ghamsari, A., "Dynamic Effects Associated with Joint Compliance in Cartesian-Frame Manipulators," *Symposium on Robot Control '85,* Pre prints of the 1st IFAC Symposium, Barcelona, Spain, pp. 125–130.

[95] Sakawa, Y. and Matsuno, F., "Modeling and Control of a Flexible Manipulator Manipulator with a Parallel Drive Mechanism," *International Journal of Control,* Vol. 44, No. 2, 1986, pp. 299–313.

[96] Sakawa, Y., Matsuno, F., and Fukushima, S., "Modeling and Feedback Control of a Flexible Arm," *Journal of Robotic System,* Vol. 2, No. 4, 1985, pp. 453–472.

[97] Schmitz, E., "Modeling and Control of a Planar Manipulator with an Elastic Forearm," *1989 IEEE International Conference on Robotics and*

Automation, May 14–19, Scottsdale, Arizona, pp. 894–899.

[98] Schmitz, E., "Dynamics and Control of a Planar Manipulator with Elastic Links," *Proc. of 25th IEEE Conference on Decision and Control,* Athens, Greece, December 1986, pp. 1135–1139.

[99] Sellhorst, G.R., *A Three-Dimensional Analysis of Space Shuttle Orbiter-Remote Manipulator System-Payload Dynamics,* Master Thesis, The University of Texas at Austin, December, 1982.

[100] Serna, M.A. and Bayo, E., "A Simple and Efficient Computational Approach for the Forward Dynamics of Elastic Robots," *Journal of Robotic Systems,* Vol. 6, No. 4, 1989, pp. 363–382.

[101] Shahinpoor, M. and Meghdari, A., "Combined Flexural-Joint Stiffness Matrix and The Elastic Deformation of a Servo-Controlled Two-Link Robot Manipulator," Robotica, Vol. 4, 1988, pp. 237–242.

[102] Siciliano, B. and Book, W.J., "A Singular Perturbation Approach to Control of Lightweight Flexible Manipulators," *The International Journal of Robotic Research,* Vol. 7, No. 4, August 1988, pp. 79–90.

[103] Siciliano, B., Yuan, B.-S., and Book, W.J., "Model Reference Adaptive Control of a One Link Flexible Arm," *Proc. of 25th IEEE Conference on Decision and Control,* Athens, Greece, December 1986, pp. 91–95.

[104] Singer, N.C. and Seering, W.P., "Design and Comparison of Command Shaping Methods for Controlling Residual Vibration," *Proc. IEEE International Conference on Robotics and Automation,* Scottsdale, Arizona, May 1989, pp. 888 –893.

[105] Singer, N.C. and Seering, W.P., "Using Acausal Shaping Techniques to Reduce Robot Vibration," *Proc. IEEE International Conference on Robotics and Automation*, philadelphia, PA, April 1988, pp. 1434 -1439.

[106] Skelton, R.E. and Hughes, P.C., "Modal Cost Analysis for Linear Matrix-Second-Order Systems," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 102, No. 3, September 1980, pp. 151–158.

[107] Sklar, M.E., *Metrology and Calibration Techniques for the Performance Enhancement of Industrial Robots,* Ph.D. Dissertation, The University of Texas at Austin, July 1988.

[108] Sklar, M.E., Hudgens, J.C., and Tesar, D., "Dynamic Model Calibration for Rigid-Link Serial Manipulators," accepted for presentation, *ASME 1990 Computers in Engineering Conference,* Boston, MA, August 1990.

[109] Slotine, J.-J.E. and Li, W., "Composite Adaptive Control of Robot Manipulators," submitted to *Automatica*, February, 1988.

[110] Slotine, J.-J.E. and Hong, S., "Two-Time Scale Sliding Control of Manipulators with Flexible Joints," *Proc. of 1986 American Control Conference*, Seattle, WA, June 1986, pp. 805–810.

[111] Spong, M.W., "Modeling and Control of Elastic Joint Robots," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 109, No. 4, December 1987, 310–319.

[112] Spong, M.W., Khorasani, K., and Kokotovic, P.V., "An Integral Manifold Approach to the Feedback Control of Flexible Joint Robots," *IEEE*

Journal of Robotics and Automation, Vol. RA-3, No. 4, August 1987, pp. 291–300

[113] Streit, D.A., Krousgrill, C.M., and Bajaj, A.K., "A Preliminary Investigation of the Dynamic Stability of Flexible Manipulators Performing Repetitive Tasks," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 108, No. 3, September 1986, pp. 206–214.

[114] Su, R., "On the Linear Equivalents of Nonlinear System," *Systems and Control Letters,* Vol. 2, No. 1, July 1982, pp. 48–52.

[115] Sunada, W.H. and Dubowsky, S., "The Application of Finite Element Methods to the Dynamic Analysis of Flexible Spatial and Co-Planar Linkage Systems," *ASME Journal of Mechanical Design,* Vol. 103, July 1981, pp. 643–651.

[116] Taylor, H.J. "A Large-Scale Manipulator For Space Shuttle Payload Handling-The Shuttle Remote Manipulator System," *Teleoperated Robotics in Hostile Environments*, edited by Martin, H.L. and Kuban, D.P., published by Robotic International of SME, 1985, pp. 142–150.

[117] Tesar, D., "A Review of Modeling, Control, and Design of Manipulators in Terms of the Planar 3 DOF System," University of Florida, Internal Report, Gainesville, Florida, January, 1978.

[118] Tlusty, J. and Wegerif, D., "Compensating for Deflections of a Robot in Light Machining Operations," *Robotics: Theory and Applications*, ASME Winter Annual Meeting, Dec. 1986, pp. 91–100.

[119] Tosunoglu, S., Lin, S.H., and Tesar, D., a, "Complete Accessibility of Oscillations in Robotic Systems by Orthogonal Projections," *ASME*

Journal of Dynamic Systems, Measurement, and Control, Vol. 112, No. 2, June 1990, pp. 194–202.

[120] Tosunoglu, S., Lin, S.H., and Tesar, D., b, "Accessibility and Controllability of Flexible Robotic Manipulators," *1990 American Control Conference,* San Diego, California, May 23–25, 1990, Vol. 1, pp. 704–711.

[121] Tosunoglu, S., Lin, S.H., and Tesar, D., "Identification of Inaccessible Oscillations in N-Link Flexible Robotic Systems," *28'th IEEE Conf. on Decision and Control,* Tampa, FL, 1989, Vol. 3, pp. 2512–2518.

[122] Tosunoglu, S. and Tesar, D., "State of the Art in Adaptive Control of Robotic Systems," *IEEE Transactions on Aerospace and Electronic Systems,* Vol. 24, 1988, pp. 552–561.

[123] Tsujisawa, T. and Book, W.J., "A Reduced Order Model Derivation for Lightweight Arms with a Parallel Mechanism," *1989 IEEE International Conference on Robotics and Automation,* May 14–19, Scottsdale, Arizona, pp.

[124] Usoro, P.B., Nadira, R., and Mahil, S.S., "A Finite Element/Lagrange Approach to Modeling Lightweight Flexible Manipulators," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 108, No. 3, September 1986, pp. 198–205.

[125] Ussher, T.H. and Doetsch, K.H., "An Overview of the Shuttle Remote Manipulator System," Space Shuttle Technical Conference, NASA Lyndon B. Johnson Space Center, Houston, Texas, June 28–30, 1983, pp. 892–904.

[126] Wander, J.P. and Tesar, D., "Pipelined Computation of Manipulator Modeling Matrices," *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 6, December 1987, pp. 556–566.

[127] Wang, W.J., Lu, S.S., and Hsu, C.F., "Experiments on the Position Control of a One-Link Flexible Robot Arm," *IEEE Transactions on Robotics and Automation,* Vol. 5, No. 3, June 1989, pp. 373–377.

[128] Wang, D. and Vidyasagar, M., "Transfer Functions for a Single Flexible Link," *1989 IEEE International Conference on Robotics and Automation,* May 14–19, Scottsdale, Arizona, pp. 1042–1046.

[129] Wang, D. and Vidyasagar, M., "Modeling of a 5-Bar-Linkage Manipulator with One Flexible Link," *1988 IEEE International Conference on Robotics and Automation,* April 24–29, Philadelphia, Pennsylvania, pp. 21–26.

[130] Yang, G.-B. and Donath, M., "Dynamic Model of a One-Link Robot Manipulator with Both Structural and Joint Flexibility," *1988 IEEE International Conference on Robotics and Automation,* April 24–29, Philadelphia, Pennsylvania, pp. 476–481.

[131] Yang, Y.P. and Gibson, J.S., "Adaptive Control of a Manipulator with Flexible Link," *Journal of Robotic Systems,* Vol. 6, No. 3, 1989, pp. 217–232.

[132] Yuan, B.-S., Book, W.J., and Siciliano, B., "Direct Adaptive Control of a One-Link Flexible Arm with Tracking," *Journal of Robotic Systems,* Vol. 6, No. 6, December 1989, pp. 663-680.

[133] Yuh, J., "Application of Discrete-Time Model Reference Adaptive Control to a Flexible Single-Link Robot," *Journal of Robotic Systems,* Vol. 4, No. 5, 1987, pp. 621–630.

[134] Yuh, J. and Tissue, D.K., "Adaptive Control for Mechanical Manipulators Having a Joint Compliance," *1990 IEEE International Conference on Robotics and Automation,* May 13–18, Cincinnati, Ohio, pp. 1632–1637.

[135] Yuh, J., Young, T., and Baek, Y.S., "Modeling of a Flexible Link Having a Prismatic Joint in Robot Mechanism– Experimental Verification," *1989 IEEE International Conference on Robotics and Automation,* May 14–19, Scottsdale, Arizona, pp. 722–727.

[136] Yurkovich, S., Pacheco, F.E., and Tzes, A.P., " On-lin Frequency Domain Information For Control of a Flexible-Link Robot With Varying Payload," *Proc. IEEE International Conference on Robotics and Automation*, Scottsdale, Arizona, May 1989, pp. 876 –881.

[137] Yurkovich, S., Tzes, A.P., Lee, I., and Hillsley, K.L., "Control and System Identification of a Two-Link Flexible Manipulator," *1990 IEEE International Conference on Robotics and Automation,* May 13–18, Cincinnati, Ohio, pp. 1626–1631.

[138] Zalucky, A. and Hardt, D.E., "Active Control of Robot Structure Deflections," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 106, No. 1, March 1984, pp. 63–69.