1994

Modelling And Experimental Vibration Control Of A Two-link Three-dimensional Manipulator With Flexible Links

Ahmed Sa Zaki

Follow this and additional works at: https://ir.lib.uwo.ca/digitizedtheses

Recommended Citation

Zaki, Ahmed Sa, "Modelling And Experimental Vibration Control Of A Two-link Three-dimensional Manipulator With Flexible Links" (1994). *Digitized Theses*. 2384.
https://ir.lib.uwo.ca/digitizedtheses/2384

MODELLING AND EXPERIMENTAL VIBRATION CONTROL OF A TWO-LINK 3D MANIPULATOR WITH FLEXIBLE LINKS

by

Ahmed S. A. Zaki

Department of Mechanical Engineering

Faculty of Engineering Science

Submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Faculty of Graduate Studies
The University of Western Ontario
London, Ontario
March 1994

Canada

ABSTRACT

Current industrial and space manipulators are required to achieve higher speeds in a lighter structure without sacrificing payload capabilities. Consequently, undesirable vibration occurs during the motion. By suitable modelling of the manipulator flexibility, advanced control strategies can be formulated to improve the joint tracking performance and reduce the residual vibration of the end-point in the presence of payload uncertainties.

Toward this goal, an experimental two-link, 3D, anthropomorphic manipulator with flexible links was designed and built to be used as a test bed for the verification and refinement of the proposed modelling and control strategies.

The nonlinear equations of motion for the robot were derived using Lagrangian dynamics. The model was verified using experimental modal analysis techniques. Based on experimental results, a simplified nonlinear model, that contains the relevant modes of the system, was derived and subsequently used in controller designs and state estimation.

A conventional Proportional-plus-Derivative (PD) controller that implements joint angles feedback was designed to be used as a baseline controller due to its wide applicability on industrial manipulators.

By measuring the links tip vibration using accelerometers, several adaptive controllers and state observers were designed and implemented successfully on the manipulator, namely, a gain-scheduling linear quadratic regulator, a model reference adaptive controller, an adaptive inverse dynamics controller, a least-squares nonlinear state estimator and a robust sliding observer. The controllers performance and robustness were tested and experimentally verified against the change of the payload.

The control strategies and identification techniques, developed in this thesis, are applicable to a wide range of robot manipulators including industrial manipulators.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

xiii

NOMENCLATURE

This section lists the symbols used in this dissertation. The subscript i denotes the link or joint number. The page in which the symbol is first used is indicated.

English

A	state matrix, pp. 109.
A_m	reference model state matrix, pp. 132.
b	joint damping, pp. 80.
B	input matrix, pp. 109.
B_m	reference model input matrix, pp. 132.
$C(q, \dot{q})$	Coriolis matrix, pp. 59.
C	output matrix, pp. 82.
$D(q)$	inertia matrix, pp. 59.
E_i	Young's modulus of elasticity of the link cross-section, pp. 58.
g	gravity loading vector, pp. 58.
$G(q)$	gravity vector, pp. 58.
G_i	modulus of rigidity, pp. 58.
H	Hamiltonian, pp. 157.
${}^{i}R_j$	rotation matrix from the ith frame to the jth frame, pp. 55.
${}^{i}T_j$	transformation matrix from frame of reference i^{th} frame to the jth, pp. 56.
I_{mi}	inertia of the joint i, pp. 58.
I_{yy}	second moment of area of the beam about the z direction, pp. 58.
I_{zz}	second moment of area of the beam about the y direction, pp. 58.
J	continuous quadratic performance index regulator, pp. 111.
J_i	polar moment of inertia of link cross-section, pp. 58.
k	regulator gain matrix, pp. 111.
K	stiffness matrix, pp. 59.
k_d	derivative gain of joint i, pp. 91.
k_r	feed-forward gain matrix, pp. 131.

k_{pi}	proportional gain of joint i, pp. 91.
k_{ti}	the torsional stiffness of joint i, pp. 61.
K_e	estimator gain matrix, pp. 125.
K_s	sliding observer gain matrix, pp. 128.
K_{OL}	open loop gain, pp. 92.
K_{Ti}	motor torque constant, pp. 48.
L	the Lagrangian, pp. 52.
L_i	length of link, pp. 54.
m_2	elbow mass plus half the second link mass, pp. 85.
m_3	payload mass plus half the third link mass, pp. 85.
m_e	elbow mass, pp. 58.
m_{li}	mass of the link, pp. 58.
m_p	payload mass, pp. 58.
O_i	origin of the coordinate frame i, pp. 52.
P	positive definite matrix, pp. 135.
$p_{ix}(t)$	generalized coordinates for the torsional modes, pp. 57.
$p_{iy}(t)$	generalized coordinates for the bending modes in the y direction, pp. 57.
$p_{iz}(t)$	generalized coordinates for the for bending modes in the z direction, pp. 57.
q	generalized coordinates, pp. 60.
Q	positive definite matrix, pp. 135.
Q_{xx}	estimator state weighting matrix, pp. 126.
Q_{yy}	estimator output weighting matrix, pp. 126.
r_{ix}	position vector of the x position on link i, pp. 57.
R_{xx}	regulator state weighting matrix, pp. 111.
R_{uu}	regulator input weighting matrix, pp. 111.
s	Laplace operator, pp. 92.
T_f	friction compensation torque, pp. 101.
u	control input, pp. 47.
v_i	velocity at the tip of link, pp. 113.
V	Lyapunov function, pp. 135.
$w_{iy}(x, t)$	elastic deflection of position x in the y direction, pp. 53.
$w_{iz}(x, t)$	elastic deflection of position x in the z direction, pp. 54.

x	state vector, pp. 107.
x_m	reference state. pp. 132.
$\hat{x}$	estimated states. pp. 125.
$\tilde{y}$	output error. pp. 128.
$\hat{y}$	estimated output. pp. 125.

Greek

θ	joint angle, pp. 48.
$\dot{\theta}$	joint velocities, pp. 48.
θ_m	motor angle, pp. 61.
Ω	cantilevered natural frequencies, pp. 92.
$\psi(x)$	bending shape function of the beam, pp. 53.
$\phi(x, t)$	torsional deflection of the beam, pp. 55.
τ	joint torque. pp. 59.
$\delta_i(x)$	elastic deflection of the x position.

Abbreviations

GS-LQR	gain-scheduling linear quadratic regulator, pp. 121.
INV-GS	adaptive inverse dynamics control with gain scheduling, pp. 174.
MFLOPS	million floating point operations per second, pp. 42.
LQG	linear quadratic gaussian control, pp. 110.
LQR	linear quadratic regulator control, pp. 115.
LMRC	linear model reference controller, pp. 131.
MRAC	model reference adaptive control, pp. 138.
PE	potential energy, pp. 58.
KE	kinetic energy, pp. 58.

- CHAPTER 1 -

GENERAL INTRODUCTION

1.1 Motivation

A manipulator is a mechanical device whose purpose is to enable its end effector to follow a commanded trajectory in its workspace and then to stay at commanded locations in order to perform a given task. Mechanical manipulators are widely used in industry because they can be programmed to execute tedious tasks easily. Robotic manipulators are also essential in hazardous environments in industry, under water and in space.

A manipulator is generally built as a chain of structurally rigid links articulated by rotary or sliding joints, each driven by its own actuator. The position accuracy of the robot end-effector is limited by the links and joints flexibility.

Intuitively, one would expect that moving the manipulator too quickly would excite its resonances and cause the tip to oscillate. This may limit the robot performance in handling large payloads and executing fast assembly operations. Consequently, robots with a large workspace (> 1 m^2) are built to be massive in structure and require powerful motors in order to move payloads that are typically around 5% of their total weight.

With the demand of efficient and large workspace manipulators capable of handling large payloads, robots will have significant mechanical flexibility. Consequently, it is required to design a control system to reduce the undesirable vibration. Since the design of a control system is based on the model of the actual manipulator, the achievable

performance may be limited by the modelling accuracy.

One example of an existing large flexible manipulator is the Shuttle Remote Manipulator System (SRMS) built for the National Aeronautics and Space Administration (NASA) by Spar Aerospace Limited of Canada. This manipulator is designed to be light-weight for easy transportation to space. The SRMS extends 15 meters. When all its links are extended and locked, its natural resonance frequency is 0.3 Hz without payload, and 0.03 Hz with its largest 16,000 lb payload. In order to minimize the excitation of the elastic modes, the SRMS is moved very slowly so that the end-point speed remains below 0.6 m/ s for the unloaded configuration. The speed and accuracy of the SRMS is limited by its control system as it uses a collocated control algorithm [71].

Another example of an existing industrial robot is the Cincinnati-Milicorn T3R3 robot. It weighs about 1800 kgs and its largest payload is 23 kgs. The robot is controlled by a Proportional-plus-Derivative (PD) control algorithm. For this robot, high speed assembly operations have to be delayed to allow the elastic vibration of the arm to settle down. The manipulator servo bandwidth is about one tenth the first cantilevered natural frequency (see Sunada and Dubowsky [85]).

One way to enhance the performance of the successors of both the SRMS and Cincinnati-Milicorn is to use non-collocated control algorithms and to implement end-point position or vibration feedback. However, this requires accurate modelling to design a robust control system.

1.2 Research Objectives

The objective of this research is to contribute to the understanding of the dynamics

and control of multi-link flexible manipulators. This research achieves the objective through the following approaches:

- To design and build a flexible 3D manipulator. While most researchers investigated planar robots, 3D robots are closer to real manipulators and suffer some problems that planar robots do not have.

- To develop modelling techniques that permit accurate representation of manipulator dynamics and to apply model reduction techniques to develop a low order model without sacrificing the accuracy of the modelling.

- To develop control and state estimation techniques that use non-collocated sensors and are applicable to any class of robots provided that a dynamic model is available.

- To experimentally verify the derived dynamic models and control algorithms, and to study their limitation and disturbance rejection performance.

This research combines both simulation and experimentation. While simulation helped in verifying the control algorithms, experimental work was more involved. The experimental work helped in identifying the controllers limitations and gave confidence in implementing the control algorithms on real manipulators.

1.3 Contributions

The control of the tip vibration of an experimental two-link, three-degree of freedom manipulator was achieved. The following fundamental contributions have been made to the fields of robotics, dynamic modelling, and control:

1. An experimental two-link, 3D manipulator with flexible links was designed and

built for the purpose of verifying the developed modelling and control algorithms. The robot is designed to be modular to allow the study of a variety of configurations.

2. A large order model for the robot was derived using Lagrange's formulation using a symbolic algebra software package. The assumed modes method was used to represent the links deflections. These equations were used to characterize the natural behavior of the system.

3. The linear behavior of the manipulator was characterized. Using experimental modal analysis, the system natural frequencies, mode shapes and damping at two nominal configurations were identified. The dominant modes of the robot were characterized and their variation with the angles was determined.

4. A reduced order model that includes the dominant modes of the robot was derived. The flexibility of the links was modelled using the Finite Element approach. The model locked natural frequencies were tuned to the experimental values using a technique based on gradient search.

5. A collocated controller for the robot was designed and experimentally verified. It includes a PD joint control, gravity compensation and model-based friction compensation. The limitations and performance of the controller are discussed.

6. For the purpose of controlling the end-point vibration, a non-collocated controller, which used the links end-point acceleration feedback, was designed based on linear quadratic regulator theory. This included a gain-scheduled optimal regulator and a nonlinear state estimator. The controller was implemented on the robot.

7. A general nonlinear and robust state estimator that can be implemented easily, in real-time, for flexible manipulators was designed and simulated on the robot.

8. In order to have a controller which is robust to the variation of the payload, an adaptive controller was designed using the model reference approach. The reference model was chosen to be the linearized system under LQR control. The adaptation gains were synthesized to guarantee the system stability using the Lyapunov theory. The controller was experimentally tested under various loading conditions.

9. Using the inverse dynamics approach, an adaptive controller was developed based on the passivity theory. The performance of the controller was enhanced by implementing a regulator in the outer loop. The composite controller was implemented in real time and experimentally verified.

10. A comparison between the different controllers was done, in a unified way, using some defined performance measures. These performance measures may be used to test any developed controller. They include trajectory tracking measures and disturbance rejection measures.

Finally, it should be noted that the innovative techniques developed in this thesis have many generic features which make them useful to many space and industrial robotic applications. This would permit the use of robots in many new tasks and produce a robotic manipulator capable of handling larger payloads in a larger workspace at higher speeds.

1.4 Thesis Organization

The dissertation is divided into ten chapters and three appendices. Included in Chapter 1 is the motivation for this research, a discussion of the research objectives, a list of contributions and the thesis organization.

Chapter 2 contains a brief literature review that includes a review of the existing experimental flexible manipulators, dynamic modelling strategies, flexible manipulator control techniques, the use of passive and active damping and coulomb friction effect.

Chapter 3 contains a description of the experimental manipulator system. Included are the design criteria for the robot, design considerations, design parameters, the mechanical design and the computer systems used for simulation and control.

In Chapter 4, a high order model was derived for the robot. Included are the modelling assumptions, the description of the assumed modes method, the development of the equations of motion, the inclusion of the joint dynamics, and the calculation of system natural frequencies.

In Chapter 5, the parameters of the robot are experimentally identified. Included are the modal parameters identification using experimental modal analysis, torque constant estimation, damping estimation, and the identification of the robot joints inertia.

Chapter 6 includes the development of a reduced order model for the robot, a technique for model tuning by matching the locked natural frequencies of the model to the experimental data is applied.

Chapter 7 includes a discussion of PD control and its limitations, and experimental results which shows the effect of increasing the controller gains.

Chapter 8 contains the development of a control strategy based on optimal regulator theory. Included are the controllability and observability of the robot, the regulator and estimator design, enhancement of the controller performance using gain scheduling, the experimental implementation of the control algorithm, and the design of a nonlinear state estimator for flexible manipulators.

In Chapter 9, an adaptive controller is designed using the model reference approach. Included are the controller structure, the derivation of the adaptation technique, the enhancement of the reference model and the simulation and experimental results.

Chapter 10 contains the design of an adaptive controller based on the inverse dynamics method. Included are the properties of the equations of motion, the passivity structure of the flexible manipulator, discussion of the joint based inverse dynamics method, the passive controller design, the effect of the flexible dynamics, the development of a composite controller, and the experimental implementation of the controller.

Chapter 11 contains a summary of the designed controllers and addresses the comparison of the developed controllers in a unified way.

Finally, in chapter 12, the conclusions and a summary of the thesis research are illustrated. Also, recommendations for future research areas are presented.

Three appendices are attached to the end of the thesis. Appendix A contains a detailed description of the manipulator construction and the specifications for the actuators, transmission, and brakes.

Appendix B contains the listing of some important software that includes MACSYMA macros to generate the manipulator dynamic model, and a MATRIX$_X$ user defined function to calculate the equivalent discrete gains and solve the Lyapunov equation.

Appendix C contains a description of the real-time controller boards, the operator control sequences to start and stop the robot, the graphical interface program and the inverse kinematic equations for the robot. Finally, a bibliography which lists all the references that were researched for this thesis is included in the end.

- CHAPTER 2 -

LITERATURE SURVEY

The fundamental issues in the study of flexible manipulators are the modelling and control. Several subjects that are applicable for rigid manipulators can be addressed once the modelling and control problems are solved, e.g. path planning, obstacle avoidance, force control, and teleoperation. The literature review for this thesis is divided into five sections, namely: existing flexible manipulators, dynamic modelling, flexible manipulator control, passive and active damping, and finally the effect of coulomb friction.

2.1 Existing Flexible Manipulators

Experimental rigs have been constructed in many universities and research centres to investigate the modelling and control of flexible structures. The logical and historical evolution for the discussion of flexible manipulators is shown in Figure (2.1). However, flexible joint research was done nearly parallel to flexible links research. Since the emphasis of this dissertation is on the vibration control of manipulators with flexible links, the literature review will focus on this issue.

2 1.1 Single Link Manipulators

It was logical to study the single link manipulator as a starting point for research on flexible manipulators. The primary advantages of the single link configuration are that the

8
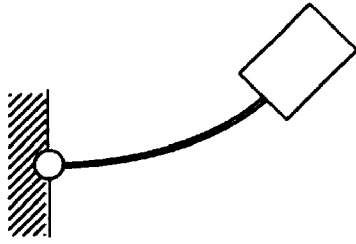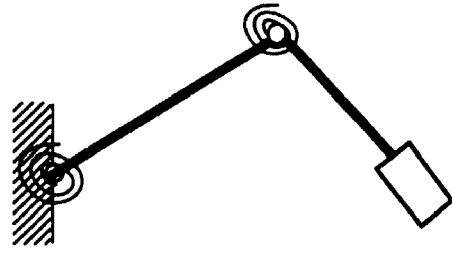
Single link

Two-link with flexible joints
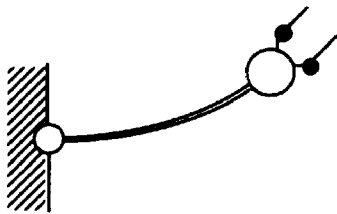
Single link with mini-manipulator
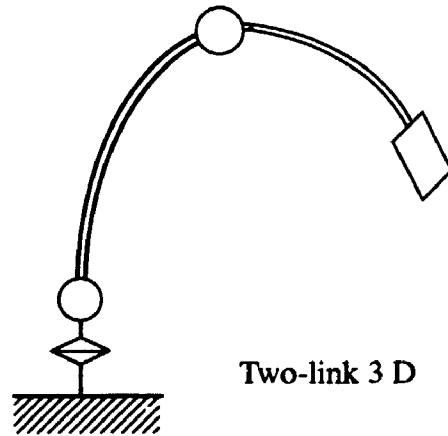
Two-link 3 D
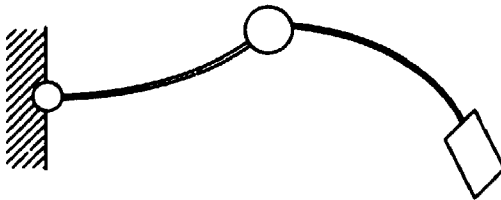
Two-link planar
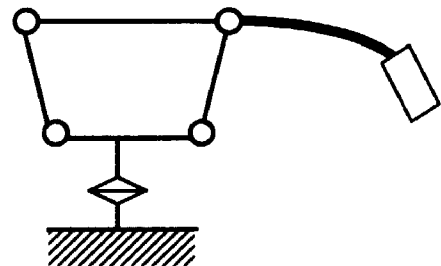
Parallel drive mechanism
with one flexible link

Figure (2.1): Flexible manipulators configurations

modelling can be done easily and the manipulator is simple to build. The main criterion for designing this manipulator is to have the lowest vertical bending and torsional modes occur at much higher frequencies than the lowest bending mode in the horizontal plane. The model developed for this system is linear and can be experimentally identified easily.

Perhaps the most relevant research was done by Cannon and Schmitz [8]. They identified the non-minimum phase behavior of the flexible link and designed an optimal controller that uses strain gauges at the hub and an end-point position sensor to control the tip of the link.

Hastings [29] used strain measurements to estimate the states of a single link and discussed the minimum number of modes and sensors needed to reconstruct the model. Rovner and Cannon [67] discussed the control of a single link with unknown payload.

A new concept was developed by adding a mini-manipulator at the end of the arm. This changes the nature of the arm by getting the large, light weight, link to carry out the gross motion and the small manipulator for the fine motion. The mini-manipulator can generate additional inertia forces that act to reduce the vibration and compensate for the relatively slow vibration of the large arm while keeping the end-point stationary. Some of these strategies as well as the modelling and control of the single link were discussed by Kraft [42], Book [6] and Wang and Vidyasagar [96].

2.1.2 Two-Link Manipulators

The two-link configuration provides a more complicated behavior than the single link since it involves the additional fundamental issues of high order control, a multi-actuator, multi-sensor plant and the nonlinear dynamics of the plant.

Oakley and Cannon [61,63] developed a low order model and implemented a regulator to control the end-point using a vision sensor. Carusone and d'Eleuterio [11] implemented strain gauges to estimate the flexible degrees of freedom and used a gain scheduling regulator to design a robust controller for the plant.

However, keeping the arm vibration in a plane (by suspending the arm on air bearing), eliminated other problems that flexible robots tend to have such as the coupling between the vibration in the horizontal and vertical planes, torsional flexibility and the gravity effect due to the payload change. An additional advantage of planar manipulators is that the end-point position can be tracked easily with an overhead camera.

Similar 2D setups are available in many universities including The Ohio State University, Rensselar Polytechnic Institute and Sandia Research Laboratories.

2.1.3 Two-link 3D Manipulators

Research in the design and control of non planar robots was done by fewer researchers due to its more complex nature. These manipulators are similar to real industrial robots and suffer from some problems that planar manipulators do not have, such as sagging, vibration in the vertical and horizontal direction, torsional vibration, and the complex change of the vibration frequencies with the change of the robot angles.

Pfeiffer and Gebler [66] designed a two-link 3D robot with flexible links. The links were rectangular in cross-section and stiff in the direction of gravity. They used strain gauges to measure the links deformation. Henerichfreise [31] designed a similar setup and used fast DSP hardware to implement the controller. Christian [16] designed a two-link 3D manipulator with flexible links and joints. The links had circular cross-sections to exhibit

symmetry in the vibration. He used an open-loop criterion to suppress the tip vibration.

2.1.4 Flexible Joints Manipulators

Flexible joints manipulators are attractive since 80% of the flexibility of industrial robots comes from the drive mechanism. Hollars [34] and Uhlik [93] designed a two-link SCARA type manipulator with flexible cables to simulate the joint flexibility. This design had encoders attached at the motors and at the links to calculate the end-point accurately without a need for an overhead camera. Similar configurations were discussed by Massoud and ElMaraghy [51] and Spong et al. [79].

2.1.5 Other Configurations

Besides the above configuration, several other configurations have been studied. A popular configuration is a parallel drive mechanism with a flexible forearm (see Sakawa et al.[70], Wang et al. [94]). This configuration has the advantage of keeping the actuators at the base and hence reducing the weight of the robot.

Yuan [100], and Tsujisawa and Book [91] have considered a very large two-link flexible manipulator with a parallel drive configuration. The robot moves in the vertical plane and both links are driven by hydraulic actuators.

2.2 Dynamic Modelling

Dynamic models are used for simulation, analysis and synthesis. In robotics,

models may be used either directly in a control algorithm like the computed torque method, or to derive the controller gains, e.g. in optimal control strategies. This section is divided into five subsections. The first is a summary of the commonly used methods to model the flexible behavior, the second discusses the nonlinear effects of large motion, the third discusses how to reduce the model order, the fourth addresses the equation development and the symbolic derivation, and the fifth section discusses the use of existing multibody codes.

2.2.1 Modelling the Flexible Behavior

Book [6] showed that it is possible to assess the modelling requirement of the system by examining the energy storage characteristics of its components.

Rigid arms store kinetic energy by virtue of their moving inertia and store potential energy by virtue of their position change in the gravitational field. Flexible arms also store potential energy by virtue of the elastic deflections of their links, joints or drive systems.

Many methods exist for modelling the flexibility of structures. A brief discussion of some of the common methods is presented.

Partial Differential Equations

The flexible link is a distributed parameter system. Theoretically, it is an infinite order system with an infinite number of modes and corresponding natural frequencies. Partial differential equations (PDE's) with time and independent spatial variables can be used to represent this problem. Slender link dynamics can be represented by the Bernoulli-Euler Equation, which ignores the shearing and rotary inertia of the beam section. For short

beams, the shearing effect can be included using the Timoshinko's beam model.

This was confirmed with Bayo's work [5] in which he showed that for short beams under high speed rotation, Timoshinko's beam model wi. give better results. However, such short "stubby" links are usually treated as rigid.

The PDE's representation of the problem has the advantage that no information due to truncation is lost, but the exact analytical solution is available only for constant cross-section links. In addition, the inclusion of body forces due to translation and rotation is very complex and the boundary conditions for multilinks are hard to derive. The solution is only available for simple cases (see Schmitz [71]).

In general PDE's can be discretized to a set of ordinary differential equations (ODE's) using approximate methods such as the finite element method, the modal expansion method and the lumped parameters technique.

The Finite Element Method

The finite element method describes the shape of the flexible link as a chain of small link segments. Each one satisfies the internal boundary conditions. Variable cross section and changing material properties can be easily modelled.

Naganthan and Soni [60] proposed a general finite element for modelling flexible arms and confirmed their results experimentally on a single link. Sunada and Dubowsky [85] presented a method to analyze complex flexible links of arbitrary shape using FEM implemented on NASTRAN. They applied their method to a real industrial robot (the Cincinnati-Milicron) and obtained good agreement with experimental results.

FEM is accurate for predicting the system modal parameters but the model order is

usually high and hence would impose limitations on designing a control system that uses this modelling strategy.

The Modal Expansion Method

The modal expansion method (called the assumed modes method) assumes that the links deformation can be represented by a finite summation of weighted functions or mode shapes that are a function of space multiplied by time dependent generalized coordinates. As a result, the order of the model is determined by the number of mode shapes used to describe the deformation of each link (component modes). Hughes [36] applied this method to analyze the SRMS.

There is no common rule followed to select the shape functions to obtain an accurate low order model. However, the best results come from the shapes that allow the natural deformation shape of the link to be accurately described in the total system.

To follow this idea, system modes representation can be used as opposed to component modes. Modi et al [56] compared both approaches for a two-beam example and found that component mode solutions are error prone, especially with equal rigidity beams. Oakley and Cannon [62] used this technique to obtain an accurate low order model for a two-link flexible manipulator. They showed that using shape functions taken from a cantilevered beam with end-point mass will give better results than using cantilevered mode shapes.

Hastings ,29] compared using clamped-mass mode shapes versus pinned-mass mode shapes for a single link and concluded that the clamped-mass mode shapes yield better results in estimating the single link natural frequencies for a large hub inertia.

Book [6] demonstrated that a clamped boundary condition leads to a physically

measurable joint variable, a simpler coefficient for the torques and better results for large

hub inertia, whereas pinned boundary conditions lead to ease in specifying the location of

the link tip. Schmitz [71] used the assumed modes to accurately fit the transfer function for

a single link (link tip to input torque at the joint) to experimental data and compared the

constrained and unconstrained modal expansion methods.

Lumped Parameter Approach

With the lumped parameter approach, the links are discretized as a series of rigid

masses to represent the beam inertia and springs to represent the beam flexibility.

The deflection shape function for the links does not have to be assumed a priori,

however, the values of the springs stiffness have to be appropriately selected to represent a

realistic model (see Zaki and ElMaraghy [102]). More accurate representation of the

flexibility can be obtained by increasing the number of beam segments.

2.2.2 Large Deflection Equations

The main assumption in modelling the flexible behavior of the links lies in the small

linear excursion from the nominal position. However, if the link length is very long and the

rotation speed is very high, this assumption may be violated assuming the material behavior

still remains within the elastic limit. Two main effects have to be considered in case of large

motion: The link foreshortening and centrifugal stiffening.

Kane et al. [39] showed that by assuming that the projected length of the

inextensible flexible beam on its rotational axis be equal to its length will lead to errors

under reasonable conditions of rotation. Kane et al. [39] showed also that if the high order

terms in the system equations of motion are discarded inconsistently (due to linearization), the simulation will predict unstable motion during high speed rotation.

Oakley and Cannon [62], in their modelling of a flexible two-link manipulator, took these effects into consideration and found that the effect of link foreshortening and centrifugal stiffening on the end-point position of the robot is very small and can be neglected for most cases. This is mainly because the rotation speed of most of present day flexible robots is not high enough to include these effects.

2.2.3 Model Order Reduction

Discretizing an infinite dimensional model and excluding higher modes may cause instability due to the spill-over effect (the effect of high frequency unmodelled modes). Deriving an appropriate reduced order model is very important for designing the controller. In theory, as indicated by Morris and Vidyasagar [58], a distributed system described by an undamped Euler-Bernoulli equation cannot be stabilized by any finite dimensional controller. In practice, the contribution of the high frequency modes to the system response is very small. In addition, all the system modes must have certain structural damping.

Several techniques exist to reduce the order of the system in state space form, e.g. the Hankle-Norm approach, the q-covariance approximation and the modal cost analysis. Tsujisawa and Book [91] applied the modal cost analysis for a large two-link manipulator and found that one or two modes per each flexible link were adequate for their case.

However, there is no guarantee that a model which includes the right degrees of freedom exists. In this dissertation, the order of the model was established by determining the number of dominant modes using experimental modal analysis.

2.2.4 Symbolic Development of the Equations of Motion

Deriving the equations of motion for flexible manipulators by hand is a very tedious task due to the exponential growth of the size of the equations with the number of modes included.

With the evolution of symbolic algebra systems, several authors addressed the systematic approach of deriving the equations of motion. They tailored the approaches to be used under symbolic algebra packages using either Lagrange's or Kane's formulation. Kane's formulation has some distinctions but Lagrange's formulation can be applied easily.

Book [7] developed a recursive formulation for the equations of motion for a class of flexible manipulators composed of a set flexible links connected by rotary joints. The kinematics of the joints and the deformed links were described by 4X4 transformation matrices using the assumed modes method.

Centinkunt and Book [14] extended this recursive formulation and developed an explicit non-recursive symbolic formulation for the same class of flexible manipulators. The final form of the equations is organized in a form similar to rigid manipulator equations which makes it easier to identify the difference between rigid and flexible manipulators. Centinkunt and Ittoop [12] showed the implementation of this algorithm on the symbolic algebra package (REDUCE).

King [41] developed a fast and accurate algorithm for general flexible manipulators using the assumed modes method. Toogood et al. [89] developed a computer package that assists in the development of the equations symbolically. Oakley and Cannon [62] presented the development of the equations of a two-link planar manipulator using Kane's method in a very concise and systematic approach.

2.2.5 Existing Multibody Codes

General purpose multibody codes are well known in simulation of multibody systems. Some packages are available to generate codes for multi-body system dynamics (MBSD) using Kane's approach e.g. (SDExact and SDFast [87]). Flexibility modelling can be accomplished, to date, using only the lumped mass stiffness approach.

Other packages such as (DADS [17]) are capable of handling large multi-body systems with flexible components. They generally accept the flexible component assumed shape data of the directly from finite element modal analysis results. They usually have interfaces to most FE packages e.g. (ANSYS [86]). They are also capable of generating a simplified linear model for control design. However, they are generally slow and do not have a direct link to the control systems packages[1].

Another elegant approach that is used for modelling general dynamic systems in a unified way is the Bondgraph approach. Packages that use this technique usually act as preprocessors to generate the code for simulation packages like ACSL [55]. However, it is difficult to model flexible members with this approach and it is still under development (see He [30], Zaki and ElMaraghy [109]).

2.3 Control of Flexible Manipulators

The control problem for flexible manipulators can be broken into trajectory planning and trajectory tracking. Trajectory planning for the joints and the end-point of a flexible arm are not equivalent problems as they are for rigid arms. The joints for the

---

1. Establishing a link between DADS & MATRIX$_x$ is under development.

flexible arm have to be controlled to account for the rigid link motion, the static deflection and the end-point residual vibration. Several authors such as Bayo and Moulin [4], Kwon and Book [44] and Tsujio [90] addressed this problem and the inverse dynamics problem.

This dissertation focuses on trajectory tracking i.e. given the joint trajectories, the robot joints are required to follow this trajectory with minimum tip vibration especially at the end of the motion.

This section is divided into four subsections: The first discusses the difference between collocated and non-collocated sensing, the second addresses some control algorithms used for robots with flexible links, the third and fourth subsections presents some important physical properties that affect the performance of the robot mainly, passive damping and coulomb friction.

2.3.1 Collocated and Non-collocated Sensing

Sensor selection is very important for the design of control systems since it places limitations on the system performance.

Collocated Sensing

Collocated sensors such as encoders, tachometers for revolute joints and linear variable differential transformers (LVDT's) for prismatic joints are located directly at the actuators.

Although collocated sensors are sufficient for the control of rigid robots, they limit the bandwidth to half the first cantilevered natural frequency for flexible manipulators due to excessive vibration at the tip [13].

Non-Collocated Sensing

In order to increase the robot bandwidth beyond its first cantilevered natural frequency, non-collocated sensors have to be used. Examples of non-collocated sensors are end-point position sensors, strain gauges, accelerometers and piezoelectric materials.

Schmitz [71] was able to increase the bandwidth of a flexible link by 30% over its first cantilevered natural frequency using direct end-point feedback and joint rate control.

However, measurements involving link flexure introduce non-minimum phase behavior, as in measuring the tip position of a single flexible link (see Schmitz [71]). The tip initially moves in the opposite direction of the applied torque which limits the controller gains due to instabilities.

2.3.2 Feedback Control Design

A typical modern control system for flexible robots may be represented by Figure (2.2). The observer is used to estimate the robot states. The feedback and feedforward controllers bring the robot to its desired trajectory.

Although several control algorithms exist for rigid robots, their application on multi-link flexible manipulators is questionable and must be modified for this purpose.

In this section, a discussion is presented on some control and state estimation algorithms that can be applied to flexible robots.

PD Control

The proportional-plus-derivative control strategy is very common for industrial
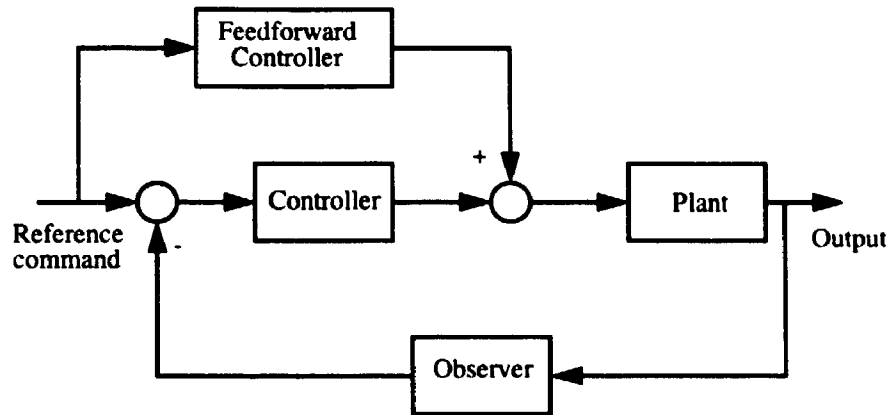
Figure (2.2): The control system structure.

manipulators control due to its simplicity and guaranteed stability (see Slotine and Li [77]).

It is achieved by comparing the desired joint angles and rates to the actual values measured

by the collocated sensors around each joint. The control action can be viewed as adding a

spring and dashpot at each joint to guide it to its desired location.

The bandwidth of this controller is limited to half the first cantilevered (joint

locked) natural frequency of the robot. This limitation is posed to avoid the interaction with

the plant lowest poles and to achieve good damping of the rigid body motion (see

Centinkunt and Book [13], Luh [48]). In other words, the robot has to move slowly to avoid

excessive vibration at the tip.

Centinkunt and Book [13] showed that if the coriolis and centrifugal forces have

comparable magnitude with the gravitational and inertia forces the bandwidth is further

restricted to 1/4 of the cantilevered natural frequency.

Optimal Control

Optimal control techniques are widely used for the control of flexible manipulators.

An optimal controller consists of a full state regulator and a state estimator. The optimal

regulator and estimator gains are based on the minimization of a quadratic performance

index. Optimal control has the advantage of the integration of multi-sensor and multi-actuator, especially for flexible manipulators, where the states cannot be measured directly.

Schmitz [71] demonstrated the use of Linear Quadratic Gaussian (LQG) controller for a one link flexible manipulator. He reconstructed the states using an end-point position sensor as well as strain measurements. The bandwidth of the system was increased up to twice the first cantilevered natural frequency.

Hollars [34] used an LQG controller along with an extended Kalman filter for the case of a two-link robot with flexible joints. The equations of motion are integrated on-line using Runge-Kutta fourth order algorithm to construct the flexible states. The controller was successfully implemented and the bandwidth was increased up to twice the cantilevered natural frequency.

Oakley and Cannon [61] implemented an optimal controller and a mostly linear estimator for a two-link flexible manipulator using end-point position sensing. Similar implementation was done by Carusone and d'Eleuterio [11] where they used strain feedback to estimate the flexible parameters of a similar system.

Yoshikawa et al. [99] implemented an optimal controller for the vibration control of a two-link 3D manipulator with flexible links. Simulation results showed significant improvement over the PD controller.

State Estimation

In general, observers are required to estimate the states which are not directly measured [24]. Designing a state observer for a single link is simple since the system is linear [71]. For nonlinear systems, which is the case with multi-link manipulators, the estimator has to be nonlinear. A comprehensive review of nonlinear observers was

presented by Misawa and Hedrick [54]. Designing observers for manipulators with flexible links was attempted by few authors due to the complex nature of the problem and its inherent nonlinear character.

Grossman et al. [27] designed a reduced order observer with a linearizable error dynamics. The observer form was obtained after some coordinate transformations and output injection. However, finding the transformation matrices poses a problem in implementing this estimator for the general multi-link flexible manipulators case.

An extended Kalman filter [34] is very difficult to implement in real-time, for the flexible links case, due to the heavy computations involved in integrating the equations of motion as well as inverting the inertia matrix.

Panzieri and Ulivi [65] designed a nonlinear observer for a two-link flexible manipulator by extracting the equations subset that describes the flexible dynamics, and compensating for the nonlinear interaction between the rigid and flexible variables. This procedure finally yielded an observer with linear dynamics which is easy to design.

The notion of sliding surfaces [77] has been implemented recently for estimating the states of nonlinear systems [76,54]. For rigid manipulators, sliding observers were used to estimate the joints velocity from position measurements in [10]. In this thesis, a general observer, based on the sliding modes approach, is derived and used to estimate the flexible states and joint velocities for flexible manipulators.

Adaptive Control

Since the characteristics of the multi-link flexible manipulators change significantly with the configuration and payload, adaptive control may have to be implemented to maintain the desired performance of the manipulator.

Authors who implemented adaptive control for flexible manipulators addressed gain scheduling, end-point mass estimation and model reference adaptive control.

Zaki and ElMaraghy [102] demonstrated the significant improvement in the performance by implementing the gain scheduling regulator over the constant gain regulator for a two-link flexible manipulator. Uhlik [93] showed that by implementing gain scheduling, the two-link flexible joint manipulator was able to preserve the same closed loop damping characteristic over its operation range.

End-point mass capture was discussed by several authors but the implementation was limited to flexible single link manipulators. The basic idea is that after identifying the payload mass, the controller gains are adjusted to compensate for the mass value. Early successful experiments were done by Rovner and Cannon [67] where the transfer function parameters for a single link was identified on line using recursive least squares technique. The identified parameters were used for on-line control design.

A comparison between different payload estimation techniques is presented by Chen and Menq [15]. They investigated the effects of high frequency unmodelled dynamics and noise on the estimator performance. They concluded that gradient search method performs the best in the presence of both effects.

Experimental and analytical evaluation of end-point mass estimation techniques for a two-link manipulator with flexible joints was done by Uhlik [93]. He demonstrated an improvement of the tracking errors using the adaptive controller over the fixed design. However, the approach of mass capture has some limitations e.g. the trajectory has to be persistently exciting, the signal to noise ratio should be high, and an accurate model for the system is needed for the identification.

MRAC was demonstrated for a single link manipulator by Sciliano et al. [74]. The robustness of the controller was tested by varying the payload. However, his algorithm was limited to the single link case.

Feedback Linearization

Feedback linearization or computed-torque control is very popular for the control of rigid robots [77]. In this method, the plant is linearized by feeding back the nonlinear dynamics to compose a linear, second-order, ordinary differential equation of the error. A tutorial in the different versions of this method including the passivity based approaches is presented in a unified approach by Ortega and Spong [64].

The control algorithms using this method are derived in the continuous-time domain. Consequently, they have to be implemented experimentally at a high sampling rate as demonstrated by Khosla and Kanade [40].

A problem that flexible manipulator dynamic models have, as opposed to rigid robot models, is that they do not have actuators for each degree of freedom. For a flexible joint two-link manipulator Uhlik [93] showed that the error equation will be of the fourth order and have to be implemented at a sampling rate which is high enough to avoid instability. He was unable to implement it experimentally due to hardware limitation.

Early application of the inverse dynamics method on a flexible link manipulator was discussed by De Luca and Siciliano [20]. They showed that this technique leads to a critically stable behavior for systems with distributed flexibility (a single link manipulator) due to the unobservability of the flexible modes. The system asymptotic stability is attributed only to the presence of structural damping in the links. Modi et al. [57] implemented this approach for the space station mobile manipulator. Other authors

discussed feedback linearization for a special class of robots with one flexible link e.g. Wang [94], Wang and Vidysagar [95], Yigit and Ulsoy [98].

The approach implemented in this dissertation is based on the passivity property for flexible manipulators which allowed the design and implementation of a robust controller for general manipulators with distributed flexibility.

Singular Perturbation Control

Since the elastic vibration is small compared to the gross rigid body motion, the complex nonlinear flexible problem can be separated into two simpler problems: the gross nonlinear (slow) rigid motion and the linear flexible (fast) motion.

By exploiting the time scale separation of both motions with singular perturbation analysis, a composite controller can be designed so that the rigid motion can be treated by any of the developed nonlinear feedback linearization algorithms (see Slotine [76]), whereas the flexible motion is handled by another loop that includes an optimal regulator designed by linearizing the system along the trajectory of the gross motion.

Siciliano and Book [73] implemented this technique on a flexible single link manipulator and demonstrated the superiority of this technique over the linear regulator approach. Ghorbel et al. [25] used the same approach for a link with a flexible joint.

If the first cantilevered natural frequency of the manipulator is close to zero, the first mode then has to be included in the slow system. This enables the dominant dynamics to remain in the slow system which makes the problem more difficult to control.

2.4 Passive and Active Damping

Passive damping is achieved by adding energy absorbing materials to the surface of the structure. While passive damping helps in reducing the vibration, it facilitates the implementation of the control algorithm as well as reducing the contribution of the spill-over of the high frequency unmodelled modes (see Alberts et al [1,2]).

Composite materials typically have more damping than metals. Alberts et al. [1,2] used constraint layer damping treatment for a single link and achieved an improvement in the damping by a factor of 10.

Manufacturing the arm from composite materials is very attractive because of their light weight, high strength and high damping. Liao et al. [47] presented schemes to optimize the geometry and material fabrication parameters for choosing the damping characteristics. Simulation results showed significant improvement of the arm dynamics. The work ongoing at the DAMRL includes designing of composite material links [53].

Active damping is widely used for flexible structural control. It consists of covering the beam with a layer of piezoelectric film. These materials respond by applying a force on the link once excited by external voltage. They also can measure the link strain. Hence, by closing the loop between the strain and voltage they can be used to suppress the link vibration (see Tzou et al. [92]). The disadvantage of using these materials is that they require high voltage to respond (in the range of 200-600 volts).

2.5 Coulomb Friction Effect

Coulomb friction constitutes the main source of nonlinearity which makes the time

response of the system extremely dependent on the reference amplitude level.

For flexible arms with high ratio speed reducers, the coulomb friction level is high enough to result in lack of back driving the actuators for relatively small command angles and small vibration. With greater stiction torque, larger amplitude vibration will exist with only structural damping reducing slowly their amplitude.

Several authors discussed friction modelling and compensation techniques. Armstrong [3] derived a model based on experimental analysis for a robot joint with a harmonic reducer and found out that in spite of the complex nature of friction, it is very repeatable. Kubo et al. [43] suggested an open loop scheme for friction compensation which is further improved by Gomes and Chretien [26].

Luh et al. [49] and Labinaz et al. [45] used strain feedback for the design of an inner torque control loop to compensate for the friction and eliminate the backlash for an industrial robot joint. This technique was implemented by Henerichfreise [31] for a flexible robot.

Adaptive friction compensation is very attractive due to the dependence of friction on many variables (e.g. temperature, load etc.). Canudas De Wit et al. [9] suggested and implemented several adaptive compensation techniques for friction compensation. The idea was to identify friction using for example recursive least squares and compensate for it through model based feedforward compensation. However, these techniques are difficult to implement for even rigid industrial manipulators due to the magnitude of the identification problem.

- CHAPTER 3 -

THE EXPERIMENTAL MANIPULATOR

The experimental manipulator serves as a test bed for verifying the modelling and control algorithms developed in this dissertation. Described in this chapter are the design criteria for the robot, design considerations, design parameters, actuator sizing, hardware design and performance, and the computer systems used. Further information about the specifications for the hardware can be found in Appendix A and Appendix C.

3.1 Design Criteria for the Robot

The idea was to design for one of the most difficult flexible manipulators problems, i.e. a two-link 3D manipulator with exaggerated flexibility at the links. If we are able to model and control it, we can apply the same modelling and control algorithms to solve industrial robots vibration control problems. The following is a summary of the design criteria for the robot and the reason for their choice:

- **Anthropomorphic, 3 degree of freedom, 2 link robot:** This configuration is chosen because it is geometrically similar to many industrial and space robots.

- **Lowest possible vibrational frequencies:** With flexible links only, it is desired to have the first cantilevered natural frequency below 3 Hz. However, if the joint design is modified to be flexible the vibration frequency will go lower.

- **Symmetry of the vibrational modes and frequencies:** This configuration has been

30

chosen to get similar response in both planes of motion.

- **Interchangeable links:** The robot is designed so that the links can be changed easily to study various link materials and geometries.

- **Modularity:** The robot has been designed so that it would be easy to use in either one or two axes.

- **Size:** Due to workspace limitations, the robot is designed to stretch up to 170 cm (full length). Since the links are interchangeable, it can be modified.

- **Speed and power:** The robot is designed to have 5 g nominal acceleration at the tip with nominal payload.

- **Static deflection:** The static deflection of the tip was restricted to 5 cm (max) when the arm is fully extended and horizontal.

3.2 Design Considerations

In order to meet the required design criteria for the robot, the following design considerations were taken into account:

3.2.1 Sagging and Static Deflection

Flexible earth based robots suffer from sagging under gravity loading. The standard way to get around sagging is to make the robot links asymmetric so that they would have higher stiffness vertically than horizontally. However, this will eliminate the symmetry in the vibration frequencies and increase the stiffness in the vertical direction.

The deflection at the end-point was limited to 5 cm when the robot stretches. This can be achieved by proper choice of material, links length and cross-section. In order to have an estimate for the static deflection at the tip and fundamental natural frequency, the system was represented by a two-cantilevered beams as shown in Figure (3.1).
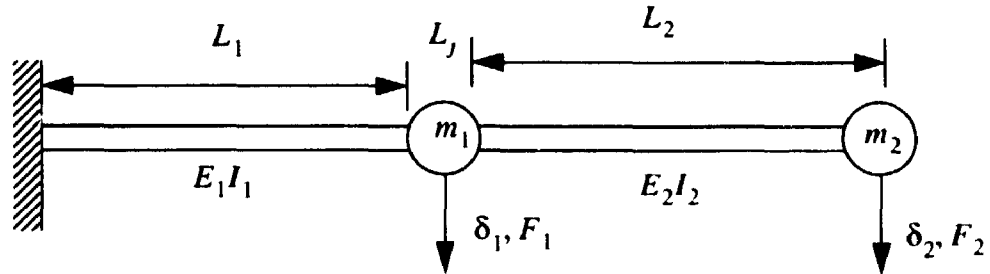
Figure (3.1): Representation of the robot as cantilevered beams.

The masses of the beams are lumped at the both ends. The beams behave in bending as Bernoulli- Euler beam. Hence, the flexibility matrix of the system is [28]

$$\begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} \tag{3.1}$$

where, δ_1 and δ_2 are the deflections at m_1 and m_2 from the forces F_1 and F_2. Under gravity loading, $F_1 = m_1 g$ and $F_2 = m_2 g$. a_{ij} can be found to be

$$a_{11} = \frac{1}{k_1}\left(1 + \frac{3L_j}{2L_1} + \frac{3L_j^2}{4L_1^2}\right) \tag{3.2}$$

$$a_{12} = a_{21} = \frac{1}{k_1}\left(1 + \frac{3L_j}{4L_1} + \frac{3}{2L_1^2}(L_1 + L_j)(L_2 + L_j)\right) \tag{3.3}$$

$$a_{22} = \frac{1}{k_2} + \frac{1}{k_1}\left(1 + \frac{3(L_2 + L_j)}{L_1} + \frac{3(L_2 + L_j)^2}{L_1^2}\right) \tag{3.4}$$

and $k_1 = \dfrac{3 . E_1 I_1}{L_1^3}$, $k_2 = \dfrac{3 . E_2 I_2}{L_2^3}$.

The deflection at the tip can be found to be

$$\delta_{tip} = \delta_2 = a_{21} m_1 g + a_{22} m_2 g \tag{3.5}$$

3.2.2 Vibration Frequency

The desired fundamental natural frequency was below 3 Hz when the robot is fully stretched and the payload is attached without including the joint flexibility. This choice of vibration frequency was based on having visible flexibility at the robot end (5 cm peak to peak). Lower vibrational frequency can be achieved by implementing springs at the joints. Using the model explained earlier, and using the standard assumption of harmonic motion, F_1 and F_2 can be replaced by the inertia forces

$$F_i = -m_i \ddot{\delta}_i = -\omega^2 m_i \delta_i \qquad (3.6)$$

The vibrational frequencies can be found by calculating the determinant and solving for ω from

$$det \begin{bmatrix} (a_{11}m_1 - \dfrac{1}{\omega^2}) & a_{12}m_2 \\ a_{21}m_1 & (a_{22}m_2 - \dfrac{1}{\omega^2}) \end{bmatrix} = 0.0 \qquad (3.7)$$

which can be solved explicitly for $f_i = \dfrac{\omega_i}{2\pi}$ (Hz) as shown

$$f_1 = \dfrac{\sqrt{2}}{2\pi} \left(a_{11}m_1 + a_{22}m_2 + ((a_{11}m_1 + a_{22}m_2)^2 - 4(a_{11}a_{12}m_1m_2 - a_{12}^2 m_1 m_2))^{\frac{1}{2}} \right)^{-\frac{1}{2}} \qquad (3.8)$$

3.2.3 Strength Considerations

Because of the large vibration amplitude at the end of the robot, the links have to be designed based on fatigue criteria. This can be achieved roughly by limiting the stress at the root of each link to half the yield strength of the material. Some other limitations were imposed by assuming the links to be round and hollow tubes with equal length and assuming equal stresses at the root of both links.

Using the available tables of pipe standards [72], the bending stress equation is

$$\sigma = \frac{My}{I}$$ (3.9)

where, $y = \frac{d_0}{2}$, and d_0 is the outer diameter. M, in general, is given by

$$M = m\,(a+g)\,L$$ (3.10)

where a is the acceleration during the robot motion, m is the moving mass and L is the length assuming cantilevered model for each link. Hence,

$$\sigma_{max} = \frac{d_0 m g l}{2I}\,(1 + \frac{a}{g}) \le \frac{1}{2}\sigma_y$$ (3.11)

The first link diameter was calculated based on a nominal acceleration of 5g at the tip, and the second link diameter on a tip motion of 5g relative to the elbow joint.

3.2.4 Joint Flexibility

The joint flexibility appears to the system as springs in series with the links. The two-link model described earlier can be modified by adding a torsional spring of value k_1 at the shoulder joint, and k_2 at the elbow joint. The coefficients a_{ij} of the flexibility matrix can be modified to take into account the torsional springs as follows [28]:

$$a'_{11} = a_{11} + \frac{(L_1 + 0.5L_j)}{k_1}$$ (3.12)

$$a'_{12} = a'_{21} = a_{12} + \frac{(L_1 + 0.5L_j)}{k_1}$$ (3.13)

$$a'_{22} = a_{22} + \frac{(L_1 + L_2 + L_j)}{k_1} + \frac{(L_2 + 0.5L_j)}{k_2}$$ (3.14)

The mechanical design for the joints can be easily modified to implement the springs, consequently, lower vibration frequencies can be achieved.

3.2.5 Size and Weight

The robot was limited by the laboratory space to be 180 cm max. The payload was limited to 1.5kg. The robot has the motors located at the joints to maximize its reach. The elbow joint mass was limited to 6 kg (max.).

3.3 Design Parameters

Several iterations were done to pick proper values for the robot parameters that will meet the design considerations stated earlier. The parameters and design constraints are shown in Table (3.1) and Table (3.2) respectively.

Table 3.1: Robot design parameters.

Element	Length (m)	Material	Specifications	Weight (kg)
First Link	0.75	Al 6061-T6	Pipe std. 0.75" Dia Sch. 40	0.46
Second link	0.75	Al 6061-T6	Pipe std 0.25" Dia Sch. 80	0.27
Elbow joint	0.1	Al 6061-T6	Al. plates 0.5" thick	5.5
Payload	0.075	Steel	Solid steel cylinder	1.5

Table 3.2: Design constraint values.

Design Constraint	Values
The static deflection at the tip	3.5 cm
The fundamental natural frequency	2.66 Hz
Stress at the first link under gravity	57 N/mm^2
Stress at the second link under gravity	52 N/mm^2
Stress at the first link with 5 g motion of the tip	239 N/mm^2
Stress at the second link with 7.5 g motion of the tip	250 N/mm^2

3.4 Hardware Design and Performance

The hardware design includes the selection of the actuators and the design of the robot joints. The performance includes the robot parameters and workspace.

3.4.1 Actuators Sizing

A basic requirement in any practical control mechanism is that adequate torque be available to control the load in some well defined manner. Selecting the actuators was done to provide enough power for future use. The specifications for selecting the actuators are as shown in Table (3.3)

Table 3.3: Specifications for selecting the actuators.

Specifications	Values
Maximum payload, including the gripper and force sensor	2 kg
Static force at the tip	100 N
Maximum acceleration at the tip	5 g
Maximum velocity at the tip	5 m/s
Maximum radius of the gripper from the mounting surface	1.8 m

It was decided to use DC servo motors for driving the robot. DC servos are used in many robotic applications. The decision for using DC servos and gear heads instead of direct-drive motors was based on weight consideration for the elbow joint and cost.

For gear heads, it was decided to use Harmonic Drives. Harmonic Drives are attractive for robotic applications because they are compact, can provide high reduction ratio with no backlash and have an exceptional torque to weight ratio.

However, Harmonic Drives have several drawbacks: they tend to be more flexible than corresponding gear systems, the spring constant of the drive is not linear but decreases with the load which tends to complicate the performance of the control system [45]. Another major drawback is that the output motion is not perfectly smooth, rather it has a small ripple corresponding to the drive frequency.

The reduction ratio was chosen to be 1:100 for all the joints. The specification of the selected actuators and drives are shown in Appendix A.

3.4.2 Components Design

This design includes the base, shoulder and elbow joints. The elbow components are the same as the shoulder but smaller in size. SDRC I-DEAS [84] was used to perform the solid modelling and to develop the drawings for the robot.

An interesting point to note for the design of the elbow and shoulder is the use of a fail-safe brake attached to the back of each motor. Without the brakes, gravity will cause the robot to collapse when power is removed.

The construction drawings for the robot are shown in Figure (3.2). The drawings for the different joints and the transmission are shown in Appendix A.

3.4.3 Robot Parameters

The specifications for the robot are shown in Table (3.4). The specifications for the robot joints are shown in Table (3.5). Note that the joints inertia were estimated using I-

ISOMETRIC

TOP VIEW

SIDE VIEW

CONSTRUCTION
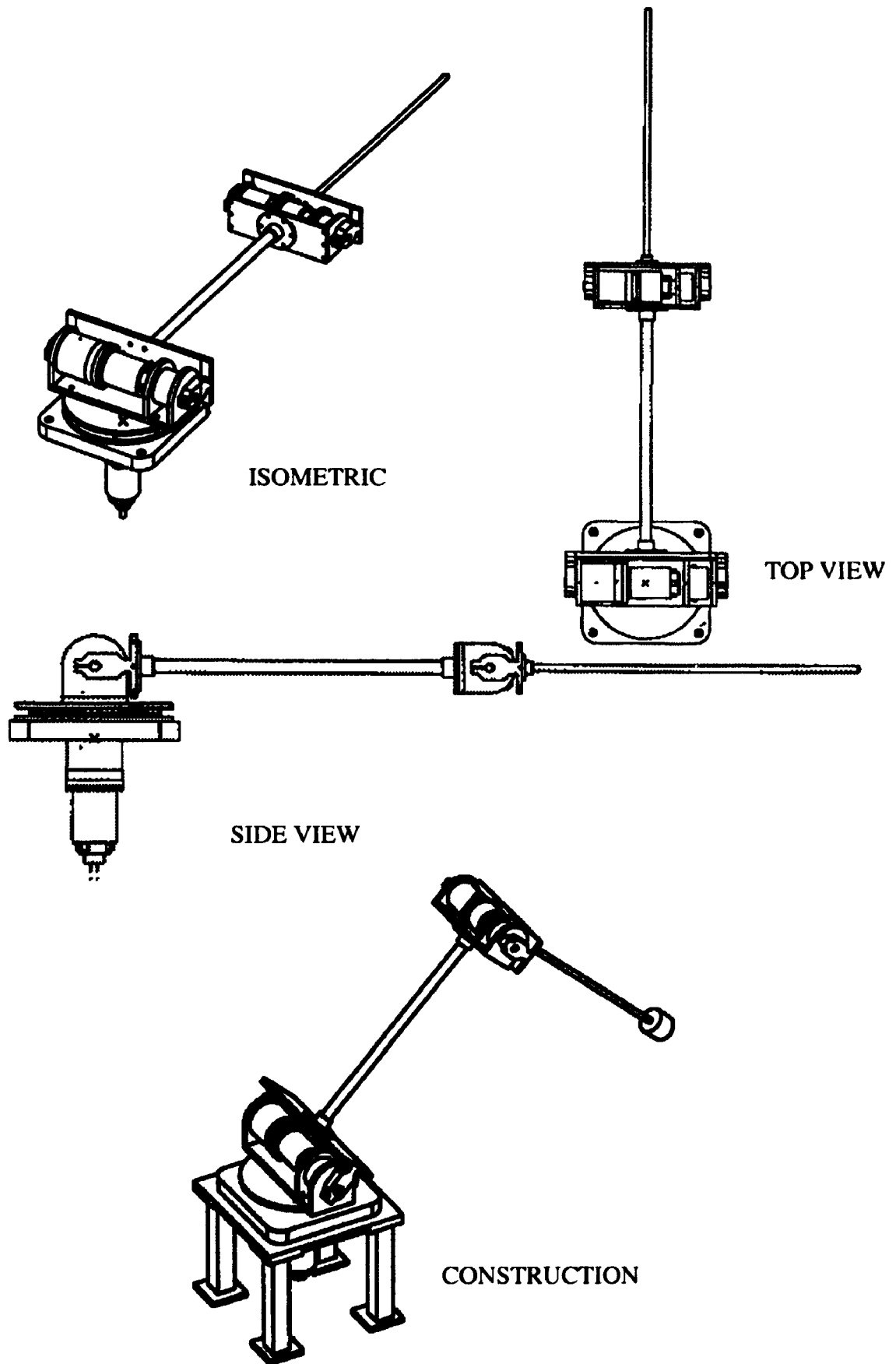
Figure (3.2): The experimental manipulator.

DEAS and the manufacturers' catalogues. They were verified experimentally.

The shoulder and base drive and motor were the same. The acceleration data for the base assumes that the arm is fully extended. It should be noted that the robot will never run at an acceleration higher than 5g as the links may yield.

The manipulator workspace is defined to be the set of locations achievable by the manipulator end-point. This space is limited by the combination of actuators, and physical space. The joint limits are shown in Table (3.5). The home position for the encoders resetting is when the robot is fully horizontal and extended.

The robot behaves as designed. The fundamental vibration frequency when the links are fully extended is 2.56 Hz. The static deflection is 3.5 cm.

Table 3.4: Robot parameters.

length	base to elbow	0.85 m
	elbow to tip	0.75 m
	overall	1.60 m
	first link	0.635 m
	second link	0.635 m
	elbow joint	0.14 m
	base to ground	0.47 m
weight	elbow	5.35 kg
	elbow joint[1]	6.775 kg
	payload	1.54 kg
	first link	0.46 kg
	second link	0.27 kg
Inertia	base joint	1.836 kg.m^2
	shoulder joint	1.638 kg.m^2
	elbow joint	1.01 kg.m^2

Table 3.4: Robot parameters.

Bending stiffness	first link	$1108.31 \ N.m^2$
	second link	$202.35 \ N.m^2$
Torsional stiffness	first link	$874.16 \ N.m^2$
	second link	$159.5 \ N.m^2$
Torsional flexibility of the drive.	Base drive	5.82E4 N.m/rad
	shoulder drive	5.82E4 N.m/rad
	elbow drive	2.64E4 N.m/rad

1. The weights of the attachment plates are added to the elbow joint

Table 3.5: Robot joints parameters.

Elbow joint	peak torque	210. N.m
	continuous torque	25.0 N.m
	maximum speed	45.0 N.m
	encoder resolution	3.6E-4 degrees
Elbow motion	max. tip acceleration	15 g
	max tip speed	3.5 m/s
	Elbow joint limit	± 90 degrees
Base joint	peak torque	542. N.m
	continuous torque	62. N.m
	max. speed	45. N.m
	encoder resolution	3.6E-4 degrees
Base motion	max. tip acceleration	11.5 g
	max. tip speed	7.5 m/s
	Base joint limit	± 135 degrees
Shoulder motion [1]	shoulder joint limit	+90 deg. -30 deg

1. The shoulder joint parameters is the same as the base

3.5 Computer Systems

Described in this section are the analysis software and the real-time hardware.

3.5.1 Analysis Software

The software tools used for the analysis and simulation are based on a UNIX environment on a network of SPARC stations available at the DAMRL. The following packages were used during the course of work: ISI MATRIX$_x$ [37], a matrix manipulation and dynamic simulation package, was used to carry out the control design and simulation. MACSYMA [88], a symbolic algebra package was used to carry out the derivation of the equations of motion.

Since flexible systems are typically large order systems, they can be easily modelled with this particular combination of software. Further information about the symbolic generation of the code will be presented in Section 4.4.5.

3.5.2 Real-Time Hardware

One of the major problems that faced this project is selecting computer hardware that is fast enough to run the high order nonlinear controllers. DSP technology seemed to be very attractive for control purposes due to its powerful computational speed.

Henrichfreise [31] used a TMS32010 processor to control a two-link 3DOF robot. However, using fixed DSP boards posed problems in the divisions and in calculating trigonometric functions. This led us to use floating point DSP hardware.

A system that was very attractive to use was developed by dSPACE Inc., called the DSP-CIT. As illustrated in Figure (3.3), the system has a TMS320C30 floating point digital signal processor, with a computation speed of 33 MFLOPS (Million Floating-point Operation per Second). Interfacing between the DSP and the robot is achieved by the dSPACE DS2002 Analog to Digital Converter (A/D) board, DS3001 Incremental Encoder board and DS2101 Digital to Analog (D/A) converter board. The system boards communicate with each other and with the host PC through a proprietary 32 bit wide peripheral high speed bus (PHS-bus).
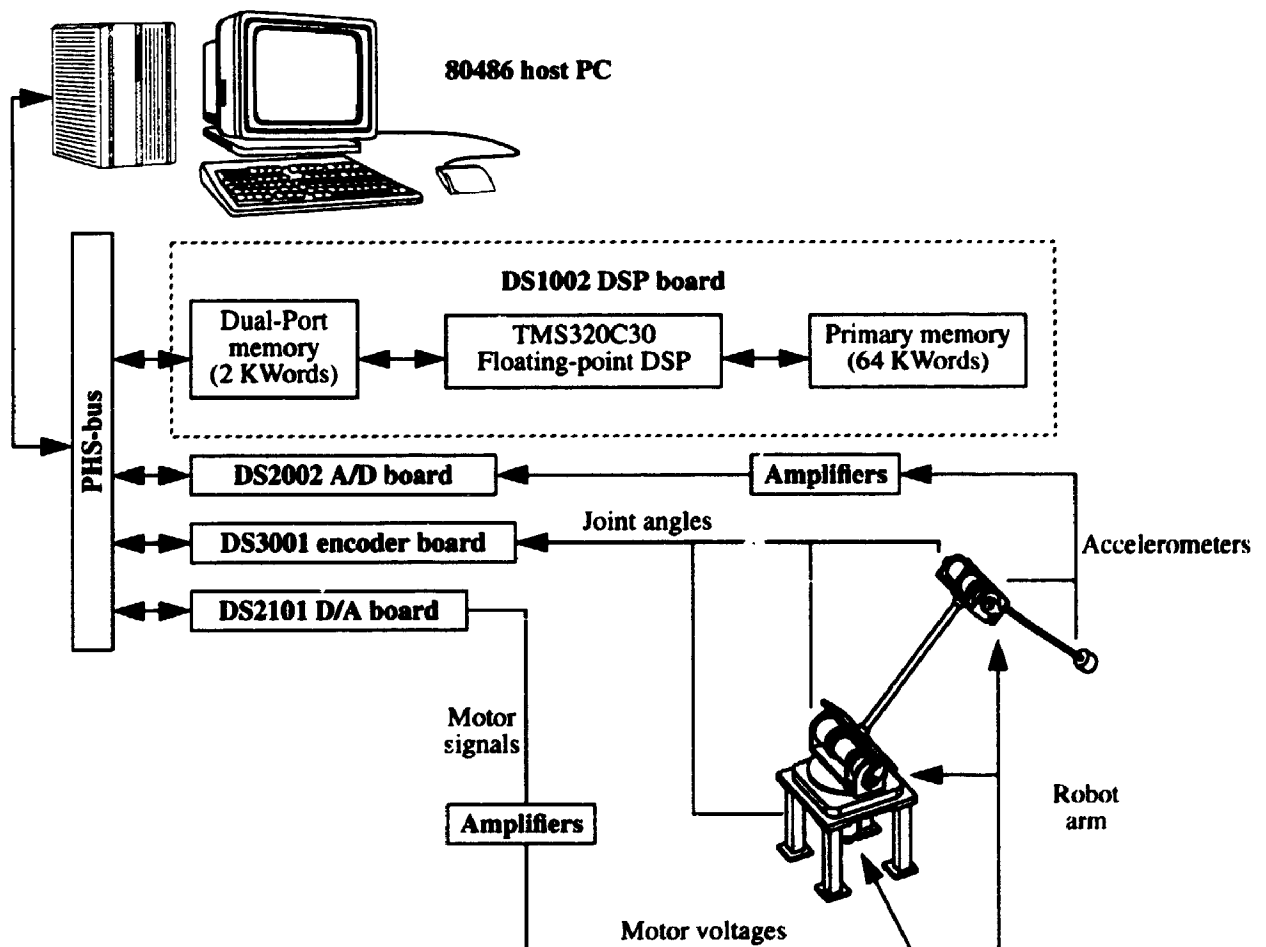
Figure (3.3): Real-time hardware configuration.

The control programs were either written in standard C language or generated using MATRIX$_X$ SystemBuild C code generator [37]. The source is combined with hardware-specific interface routines and compiled on the PC. The compiled code is then downloaded to the DSP to run in real-time.

While the control algorithm is downloaded and run on the DSP, its inputs, measured and calculated output data are linked to the PC interface through the dual-port memory. The dual-port memory is a two-KWords of memory that can be shared between the PC and the DSP chip hence achieving data transfer while the DSP is running (called on the fly). Note that the PC does not have an access to the DSP while it is running except through the dual-port memory. Consequently, an interface program was written, on the PC side, to handle the robot commanded-movement in a user-friendly environment.

Further description of the boards of the dSPACE system and the robot interface program is presented in Appendix C.

In order to drive the robot, special custom-made power circuits are manufactured to do the following: provide the operator with an interface to safely start and stop the robot, interface the motors drive signals to the D/A output, prevent the robot from collapsing using a brake-release circuit (synchronized with the motors). In addition, a tele-operation circuit was designed to drive the robot independent of the controller, and a special watch-dog circuit was built to disable individual joints from running if a computer system failure is detected.

The information about the wiring diagrams for the power electronics are available in [106]. In Appendix C, further description of the operator interface program and hardware specifications are presented.

- CHAPTER 4 -

DYNAMIC MODELLING

The design of a control system for flexible manipulators depends on how accurately the actual system is modelled. As a result, the achievable performance of the controller is intimately related to the modelling accuracy.

In this chapter, the equations of motion are derived using Lagrange's formulation. The assumed modes method is used to represent the links flexibility. The joint flexibility is introduced to the dynamic model afterward. This relatively high order model was used to predict the dominant flexible behavior of the manipulator as a first step toward deriving a reduced order model.

This chapter is divided into five sections. The robot is described in section one. Section two addresses the assumptions used in the derivation. Section three demonstrates the assumed modes method. In section four, the derivation for the equations of motion and the introduction of the joint flexibility are accomplished. Finally, in section five, the natural frequencies for the robot are calculated for the locked and unlocked configurations.

4.1 Description of the Plant

Figure (4.1) contains a schematic diagram of the two-link, three-degree of freedom flexible manipulator. The links undergo horizontal and vertical deflection as well as torsional motion. Ideal pin joints are located at the base, shoulder, and elbow joints. The

elbow motor and payload are assumed to be rigid masses with center of gravity acts at the end of the links.
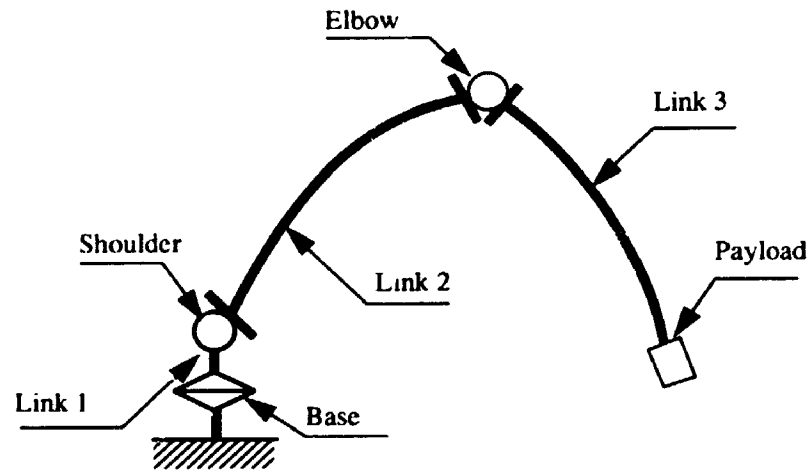
Figure (4.1): Schematic diagram for the robot.

The robot parameters, taken from Chapter 3, are calculated based on the manufacturer's catalogue and the solid modelling package I-DEAS. Note that, the links effective flexible lengths are calculated so that the elbow and shov'der joints offset are subtracted from the center to center distance.

The techniques used for developing the equations of motion can be applied to manipulator systems with different configurations. For this reason the notational conventions were devised so that the links are numbered sequentially as shown in Figure (4.1). Note that the first link has a length of zero.

4.2 Modelling Assumptions

Deriving a detailed model for the robot is a very difficult task. See Zaki et al. [108] for a detailed model using DADS [17] and ANSYS [85]. However, since the objective of

this model is to identify the flexible behavior of the manipulator, the following assumptions were made based on the physical observation of the plant.

4.2.1 Bernoulli-Euler Beam Model

The flexible dynamics of the link itself can be represented by either the Bernoulli-Euler (B-E) model or Timoshinko model. The Bernoulli-Euler beam equation is [52]

$$EI\frac{\partial^4 y}{\partial x^4} + \mu\frac{\partial^2 y}{\partial t^2} = f(x, t) \qquad (4.1)$$

where y is the deflection, μ is the mass per unit length, and EI is the flexural rigidity of the beam. In this model, the shear and rotary inertia are neglected, whereas in Timoshinko's model, both effects are included which starts to have an influence on the response if the beam length is short relative to the diameter.

Since flexible manipulators usually have slender links, the B-E model will be used. The eigen-solution for the PDE (equation (4.1)) will give a shape function which depends on the boundary conditions of the beam. This shape function can be used in the derivation of the equations of motion for the robot using the assumed modes method.

4.2.2 Small Elastic Deflection

The flexible links are assumed to be nominally straight and of a uniform cross-section, hence the flexural rigidity is assumed to be uniform. The links are assumed to be inextinsible and the deflection is assumed to be due to the bending in the horizontal and vertical planes as well as the torsional deformation.

The links axial deformation due to bending is neglected. This assumption overlooks the centrifugal stiffening effects of the links. The centrifugal stiffening effect can be explained as follows: a straight beam of length L projects a distance L on its neutral axis. When the beam undergoes bending, the projected distance must be less than L assuming no axial deformation. This effect is usually neglected, however, it usually starts to have an influence if the links are very flexible, long, and the rotation speed is high.

Consequently, for many robotic manipulator configurations and operating speeds, the effect of centrifugal stiffening is insignificant [6].

4.2.3 Simplified Actuator Dynamics

The experimental manipulator is driven by a DC servo motor. The motor model is shown schematically in Figure (4.2). Neglecting coulomb friction and joint flexibility for now, the equations of motion for the motor are as follows
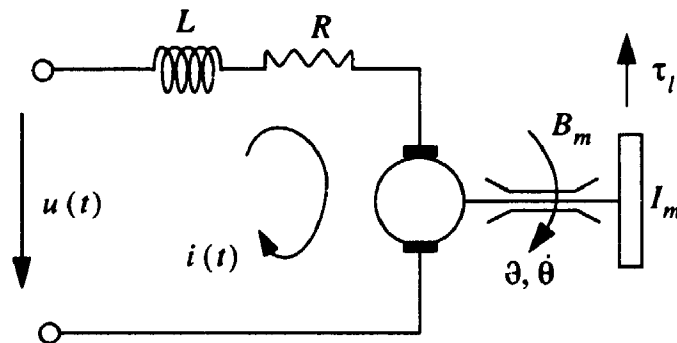
Figure (4.2): Schematic of the motor dynamics.

$$Ri + L\frac{di}{dt} = u(t) - E_b \tag{4.2}$$

$$E_b = K_b\frac{d\theta}{dt} \tag{4.3}$$

$$\tau_m = K_t i \tag{4.4}$$

$$I_m\ddot{\theta} + B_m\dot{\theta} = \tau_m - \tau_l \qquad (4.5)$$

where R and L are the armature resistance and inductance respectively, E_b and K_b are the back E.M.F and its coefficient, K_t is the torque sensitivity, I_m is the motor inertia, B_m is the motor viscous friction, u is the applied voltage, i is the current, τ_l is the load torque and τ_m is the motor torque.

Substituting from equation (4.3) to equation (4.2) and dividing by R

$$\frac{L}{R}\frac{di}{dt} + i = \frac{1}{R}(u - K_b\frac{d\theta}{dt}) \qquad (4.6)$$

The electrical time constant $\frac{L}{R}$ can be neglected since it is very small (around 2 ms) compared to the time constant of the system. Hence

$$i = \frac{1}{R}(u - K_b\frac{d\theta}{dt}) \qquad (4.7)$$

Substituting from equation (4.7) to equations (4.4), the torque equation becomes

$$\tau_m = K_t i = \frac{K_t}{R}(u - K_b\frac{d\theta}{dt}) \qquad (4.8)$$

Equating equation (4.8) and equation (4.5)

$$u = \frac{R}{K_t}(I_m\ddot{\theta} + (B_m + \frac{K_b K_t}{R})\dot{\theta} + \tau_l) \qquad (4.9)$$

By combining $I_m\ddot{\theta}$ and $(B_m + \frac{K_b K_t}{R})\dot{\theta}$ with the load torque (i.e transfer them to the robot side), equation (4.9) can be reduced to

$$\tau_m = K_T u \qquad (4.10)$$

where u is the input voltage and τ_m is the output torque at the robot joint taking the motor mechanical dynamics to be on the robot side. Consequently, the motors dynamics can be represented by gains or torque sources at the joints.

4.2.4 Simplified Joint Dynamics

The robot has DC motor-driven revolute joints. The elastic mechanical coupling between the joint and link is modelled as a torsional spring. Due to the drive high gear reduction ratio (1:100), the rotational energy of each joint about its own center of mass is only due to its own rotation. The contribution to rotational energy from the rotation of the previous joints and links is neglected.

The rotor/gear assembly inertia is assumed to be symmetric about the rotor axis of rotation such that the velocity of the joint center of mass is independent of the rotor position. This assumption is generally satisfied in most industrial robots applications.

4.2.5 Linear Damping Models

The joints coulomb friction is neglected in the modelling since it is being compensated for as will shown in Section 7.5.2. The joint damping is assumed to be linear and proportional to the joint velocity. The coefficients of the joints damping are determined experimentally.

The links structural damping is very small since they are made of Aluminum tubes. It is assumed that the links structural damping is, for now, proportional to the stiffness of the links as follows

$$C = \gamma K \tag{4.11}$$

where γ is assumed to be 2%. An experimental estimation of the damping matrix is presented in Section 5.3.1 where the damping matrix is assumed to be linear [52].

4.3 The Assumed Modes Method

The assumed modes method was used to superimpose the linear elastic deflection over the rigid body motion [52]. This section describes how this method is implemented.

4.3.1 Definition and Limitations

The motion of the manipulator is described by the rigid body rotation angles of the base, shoulder and elbow ($\theta_1, \theta_2, \theta_3$ respectively), the links deflections in the horizontal and vertical directions (w_{2y}, w_{2z} for the second link and w_{3y}, w_{3z} for the third link), and the torsional deflection of the links (ϕ_{2x} for the second link and ϕ_{3x} for the third link). The rigid body rotations are functions of time whereas the links deflections are functions of time and the distance along the undeformed axis of the link.

The Assumed Modes method can be considered as a subset for Rayleigh-Ritz approximation for the solution of partial differential equations [52]. To solve for the deflections, the system can be approximated by a set of ordinary differential equations. This can be achieved by approximating the links deflection as a finite sum of spatially dependent mode shapes multiplied by time-dependent generalized coordinates as follows:

$$w(x,t) = \sum_{i=1}^{n} \psi_i(x) p_i(t) \qquad (4.12)$$

where ψ_i (called the shape function) is the function expressing the spatial displacement of mode i, p_i is the time-varying amplitude of mode i, and n is the number of modes used to describe the deflection of the link.

The mode shapes will satisfy the geometric boundary conditions. However, using this method, the solution is limited to a finite set of modes and the infinite modes solution

that results from the PDEs is lost. Consequently, n should be at least equal to the number of dominant modes for each link otherwise we get spill-over from the unmodelled modes. The spill-over effect may result in unstable system if the controller is designed neglecting some of the dominant modes.

However, if the unmodelled modes occur at relatively high frequency and have minor contribution to the response, the spill-over effect will be minimal and the system will be stable.

4.3.2 Choice of the Shape Functions

As explained previously, the shape functions have to satisfy the geometric boundary conditions. There is no rule to select the shape functions so that a combination of model simplicity and accuracy is achieved. Popular choices for the shape functions are: simple polynomials, splines, shape functions from the eigen-solution of a beam, eigen-vector solution from finite elements problem, and modal test results performed on the actual system or its components individually [6].

If the slope of the shape functions at the link root is zero, it will lead to a physically measured joint variable and simpler coefficients for the joint torques. This choice of coordinates to represent the flexibility is called the constrained modal coordinate. An example of shape functions for this case is the cantilevered beam mode shapes.

Alternatively, if the slope of the shape functions at the root is not equal to zero, which is the unconstrained modal coordinate case, the total rotation of the link is the rigid rotation plus the slope at the root. A popular choice of the shape functions for this case is the mode shapes of a pinned-free beam.

For real manipulators, the actual deformation of the links is in between the clamped-free case and the pinned-free case. However, if the hub inertia is large, and if the link is firmly clamped at the hub, the choice of the shape functions to be the clamped-free or the clamped with mass gives a better approximation for the links deflection than the pinned-free mode shapes.

4.4 Development of the Equations of Motion

In this section, the equations of motion are developed for the flexible manipulator using Lagrange's formulation. The joints flexibility wil' be introduced to the model by introducing their torsional effect afterward.

4.4.1 Coordinate Systems

The coordinate frames are shown in Figure (4.3). O_i is the origin of the base frame $(XYZ)_i$, where i is the frame number. The inertial frame of the system O_0 is at the int:rsection of the base and the shoulder joints. $(XYZ)_1$ is defined by rotating $(XYZ)_0$ around Z_0 an angle θ_1 [14,99].

The system $(XYZ)_2$ is taken by defining X_2 along the axis of the second link, Z_2 is the shoulder joint axis. The transformation between the coordinate system $(XYZ)_2$ and $(XYZ)_1$ is achieved by ro.ating about X_2 $90°$ and by rotating around Z_2 an angle of $_2$.

The frame $(XYZ)_2$ is fixed to the third link where X_3 is along the undeformed axis of the third link and O_3 is at the point where the second link meets the elbow joint axis when the second link is undeformed.

The flexibility of the links is defined by defining the coordinate frames $(XYZ)_{ei}$ which represents the infinitesimal deformation of link i in such a way that the directions of their axes coincide with those of $(XYZ)_i$ when no link deformation exists. The deformation of the tip of each link is adequately described by this coordinate systems.
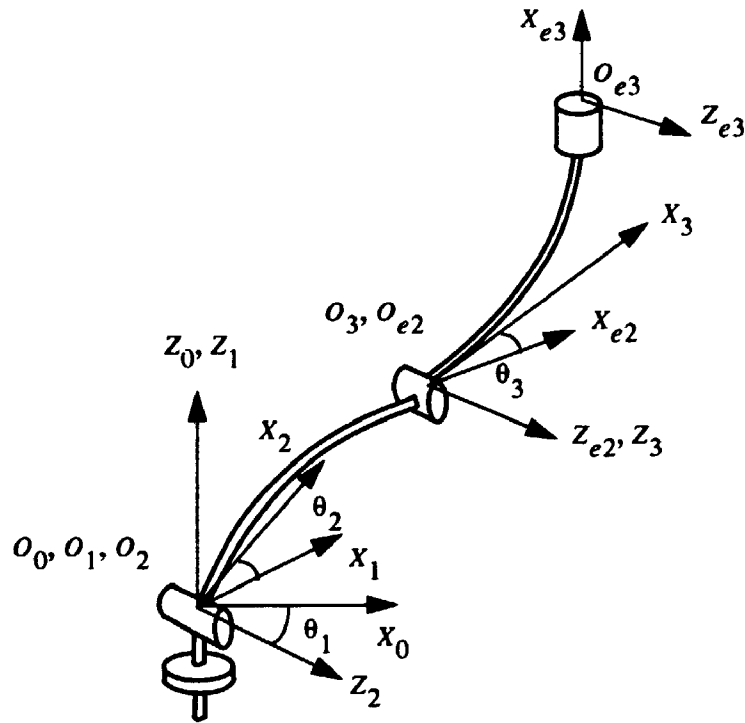
Figure (4.3): Robot coordinate frames.

4.4.2 Deformation of Each Link

The deformation of each link is described with respect to its local frame at its root. Since usually the X_i axis is taken to be along the undeformed axis of link i, the deflection in the Y_i and Z_i direction of link i can be described as

$$w_{zi}(x, t) = \sum_{j=1}^{n} \psi_{jz}(x) p_{jz}(t) \quad (i=2,3) \tag{4.13}$$

$$w_{yi}(x, t) = \sum_{j=1}^{n} \psi_{jy}(x) p_{jy}(t) \qquad (i=2,3) \qquad (4.14)$$

The bending deflection of each link is modelled by only the first cantilevered mode

shape. The expression for the ith cantilevered mode shapes is

$$\psi_i(x) = L\left(\cosh(\beta_i x) - \cos(\beta_i x) - \left(\frac{\cosh\lambda_i + \cos\lambda_i}{\sinh\lambda_i + \sin\lambda_i}\right)(\sinh(\beta_i x - \sin\beta_i x))\right) (4.15)$$

where $\lambda_i = \beta_i L$ is the ith root of: $1 + \cos\lambda_i \cosh\lambda_i = 0$ and the cantilevered natural

frequencies Ω_i are related to λ_i by

$$\Omega^2_i = \frac{EI}{\rho L^4}\lambda^4_i \qquad (4.16)$$

where ρ is the mass of the beam per unit length. The numerical values for the first three

roots are 1.8751, 4.694 and 7.855 respectively [52,71].

Since the symbolic algebra system (MACSYMA) was used to carry out the

derivation, the expression (equation (4.15)) is very complicated to implement. If equation

(4.15) was implemented, it would result in a huge expression for the Lagrangian and

eventually computer system memory overrun. Using a third order polynomial fit, shown in

Figure (4.4), the expression for the deflection will be the same as the deflection due to static

force at the link tip given by

$$\psi(x) = \frac{3}{2}\left(\frac{x}{L}\right)^2 - \frac{1}{2}\left(\frac{x}{L}\right)^3 \qquad (4.17)$$
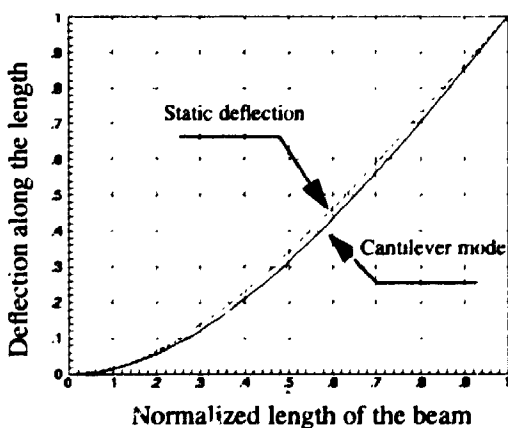
Figure (4.4): A comparison between the fitted and actual shape function.

The torsional deflection of each link can be approximated to be linear with x [28]; the expression for the torsional deflection is

$$\phi_{ix}(x, t) = (\frac{x}{L})p_{ix}(t) \quad (i=2,3)$$ (4.18)

where $p_{ix}(t)$ is the time dependent coordinate.

4.4.3 Kinematics

The rotation matrices and translation vectors are defined in this section. The position vector for any arbitrary point on the robot will be defined consequently.

Rotation Matrices

Refer to Figure (4.3), the rotation matrices are defined with respect to the transformed frames. The rotation matrices from the ith to the jth frame is $^{i}R_{j}$. Hence

$$^{0}R_{1} = rot(z_0, \theta_1) = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$ (4.19)

$$^{1}R_{2} = rot(x_1, 90)rot(z_2, \theta_2) = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ 0 & 0 & -1 \\ \sin\theta_2 & \cos\theta_2 & 0 \end{bmatrix}$$ (4.20)

$$^{e_2}R_{3} = rot(z_{e2}, \theta_3) = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 \\ \sin\theta_3 & \cos\theta_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$ (4.21)

The rotation matrices due to the links elastic deflection can be derived by using Taylor's expansion for the sine and cosine functions around zero and neglecting high-order terms (due to the very small values of elastic deflections). Hence, the rotation matrix from $(XYZ)_{e2x}$ (the subscript x_2 denotes the x_2 position of the second link) to the $(XYZ)_2$ is

$$^2R_{e_{2x}} = rot(z_{e2}, \phi_{2z})\, rot(y_{e2}, \phi_{2y})\, rot(x_{e2}, \phi_{2x}) = \begin{bmatrix} 1 & -\phi^x_{2z} & \phi^x_{2y} \\ \phi^x_{2z} & 1 & -\phi^x_{2x} \\ -\phi^x_{2y} & \phi^x_{2x} & 1 \end{bmatrix} \qquad (4.22)$$

where

$$\phi_{2x} = \left(\frac{x_2}{L_2}\right) p_{2x}(t) \qquad (4.23)$$

$$\phi_{2y} = \psi'_{2z} p_{2z}(t) \qquad (4.24)$$

$$\phi_{2z} = \psi'_{2y} p_{2y}(t) \qquad (4.25)$$

The rotation at the end of the second link with respect to $(XYZ)_2$ can be obtained by setting x_2 equal to L_2 in equation (4.22). Similarly, the rotation matrix from $(XYZ)_{e3x}$ to $(XYZ)_3$ is

$$^3R_{e_{3x}} = rot(z_{e3}, \phi_{3z})\, rot(y_{e3}, \phi_{3y})\, rot(x_{e3}, \phi_{3x}) = \begin{bmatrix} 1 & -\phi^x_{3z} & \phi^x_{3y} \\ \phi^x_{3z} & 1 & -\phi^x_{3x} \\ -\phi^x_{3y} & \phi^x_{3x} & 1 \end{bmatrix} \qquad (4.26)$$

Defining ^0T_i to be the rotation matrix from the inertial frame of reference $(XYZ)_0$ to $(XYZ)_i$, hence

$$^0T_2 = {}^0R_1\, {}^1R_2 \qquad (4.27)$$

$$^0T_{e_{2x}} = {}^0R_1\, {}^1R_2\, {}^2R_{e_{2x}} \qquad (4.28)$$

$$^0T_{e_{3x}} = {}^0R_1\, {}^1R_2\, {}^2R_{e_2}\, {}^{e_2}R_3\, {}^3R_{e_{3x}} \qquad (4.29)$$

where $^0T_{e_{2x}}$, $^0T_{e_{3x}}$ are the rotation of any point on the second and third link respectively.

Translation Vectors

Since the links are assumed to be inextensible, the translation vector for an arbitrary point on any link with respect to its corresponding local frame at its root is

$$d_{ix} = \begin{bmatrix} x_i \\ \delta^x_{iy} \\ \delta^x_{iz} \end{bmatrix} \quad (i=2,3)$$

(4.30)

where δ^x_{iy}, δ^x_{iz} are the deflections at the position x of the ith link given by

$$\delta^x_{iy}(t) = \psi_{iy}(x)p_{iy}(t)$$

(4.31)

$$\delta^x_{iz} = \psi_{iz}(x)p_{iz}(t)$$

(4.32)

The elastic deflection at the end of the link can be obtained by evaluating equations

(4.31,4.32) at $x = L_i$.

Position Vectors

The position vector for any arbitrary point along the second link with respect to the

inertial frame of reference is

$$r_{2x} = {}^0T_{e_{2x}}d_{2x}$$

(4.33)

Similarly, the position vector for any point on the third link with respect to the

inertal frame $(XYZ)_0$ is

$$r_{3x} = {}^0T_{e_2}d_2 + {}^0T_{e_{3x}}d_{3x}$$

(4.34)

where ${}^0T_{e_2}$, d_2 are calculated at the tip of the second link.

4.4.4 Dynamics

Lagrange's formulation was implemented in deriving the robot equations of motion.

To formulate the Lagrangian, the expressions for the potential and kinetic energies are

needed.

The Kinetic Energy

The kinetic energy for the manipulator can be divided into three parts, namely

- The kinetic energy of the rotating components at the joints (note that the joints inertias are calculated on the robot side)

$$KE_1 = \frac{1}{2} I_{m1} \dot{\theta}^2_1 + \frac{1}{2} I_{m2} \dot{\theta}^2_2 + \frac{1}{2} I_{m3} \dot{\theta}^2_3 \tag{4.35}$$

- The kinetic energy of the payload and the elbow joint

$$KE_2 = \frac{1}{2} m_p \dot{r}^T_3 \dot{r}_3 + \frac{1}{2} m_e \dot{r}^T_2 \dot{r}_2 \tag{4.36}$$

- The kinetic energy of the links which includes the kinetic energy of vibration

$$KE_3 = \frac{1}{2} \frac{m_{l2}}{L_2} \int_0^{L_2} \dot{r}^T_{2x} \dot{r}_{2x} dx_2 + \frac{1}{2} \frac{m_{l3}}{L_3} \int_0^{L_3} \dot{r}^T_{3x} \dot{r}_{3x} dx_3 \tag{4.37}$$

The total kinetic energy is

$$KE = KE_1 + KE_2 + KE_3 \tag{4.38}$$

The Potential Energy

The potential energy can be divided into two parts, namely

- The gravitational potential energy

$$PE_1 = m_e g^T r_2 + m_p g^T r_3 + \frac{m_{l2}}{L_2} \int_0^{L_2} g^T r_{2x} dx_2 + \frac{m_{l3}}{L_3} \int_0^{L_3} g^T r_{3x} dx_3 \tag{4.39}$$

where $g^T = [0, 0, -g_0]$, and g_0 is the acceleration of gravity.

- The strain energy due to the bending and torsional flexibility

$$PE_2 = \frac{1}{2} \int_0^{L_2} E_2 I_{2zz} w''^2_{2y} dx_2 + \frac{1}{2} \int_0^{L_2} E_2 I_{2yy} w''^2_{2z} dx_2 + \frac{1}{2} \int_0^{L_2} G_2 J_2 \Phi'^2_{2z} dx_2 +$$

$$\frac{1}{2} \int_0^{L_3} E_3 I_{3zz} w''^2_{3y} dx_3 + \frac{1}{2} \int_0^{L_3} E_3 I_{3yy} w''^2_{3z} dx_3 + \frac{1}{2} \int_0^{L_3} G_3 J_3 \Phi'^2_{3z} dx_3 \tag{4.40}$$

The total potential energy is

$$PE = PE_1 + PE_2 \tag{4.41}$$

The Equations of motion

Defining the Lagrangian L to be [7,52]

$$L = KE - PE \tag{4.42}$$

Using the joint coordinates θ_i $(i=1,2,3)$ as generalized coordinates, the equations of motion become

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{\theta}_i}) - \frac{\partial L}{\partial \theta_i} = \tau_i \ (i=1,2,3) \tag{4.43}$$

where τ_i is the ith motor torque applied at the ith joint.

Similarly, the equations of motion for the flexible generalized coordinates are

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{p}_i}) - \frac{\partial L}{\partial p_i} = 0 \tag{4.44}$$

where p_i is $(p_{2x}, p_{2y}, p_{2z}, p_{3x}, p_{3y}, p_{3z})$ the flexible generalized coordinates.

The final form of the equations of motion is very long and complicated. The derivation was carried out under a symbolic algebra system as explained in the following section. The final form of the equations of motion is

$$\begin{bmatrix} D_\theta & D_{\theta p} \\ D_{p\theta} & D_p \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{p} \end{bmatrix} + \begin{bmatrix} C_\theta & C_{\theta p} \\ C_{P\theta} & C_p \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{p} \end{bmatrix} + \begin{bmatrix} G_\theta \\ G_p \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} \theta \\ p \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix} \tag{4.45}$$

4.4.5 Frame of Work

This section demonstrates how the derivation was carried out under MACSYMA and how it is put in a simplified form for simulation he steps for the derivation are as follows

- The kinetic, potential energy and the Lagrangian are derived by the formula given in equations (4.41, 4.38). Equation simplification routines were applied to the Lagrangian expression so that the time independent terms are extracted and replaced by a constant to be evaluated once and off-line.

- The equations of motion are derived (equations (4.43,4.44)) and simplified so that the constant terms as well as the terms that are functions of q, $\dot{q}$ are extracted, where $q^T = \begin{bmatrix} \theta^T & p^T \end{bmatrix}$. Note that the dimension of q is 9 (3 rigid DOF + 6 flexible DOF).

- The coefficients of $\ddot{q}^T = \begin{bmatrix} \ddot{\theta}^T & \ddot{p}^T \end{bmatrix}$ are extracted (which is the inertia matrix) to put the final equations in a state space form as follows:

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ -D^{-1}\left(\tau - F(q, \dot{q})\right) \end{bmatrix} \tag{4.46}$$

where $F(q, \dot{q})$ is the nonlinear terms vector.

- The FORTRAN code for the equations of motion was generated and linked with the control system package MATRIX$_x$ in the form of "user code block" to carry out the simulation and control design. This frame of work helped in speeding up the simulation and control system design. Available in Appendix B. is the derivation of the equations of motion up to the final form (equation (4.46)) on MACSYMA.

4.4.6 Inclusion of the Joint Dynamics

Inclusion of the joint dynamics in the model involves modifying equation (4.45) by redefining the links inertia matrix and augmenting the model by a set of second order equations which represent the joint inertia and flexibility.

The flexible joint-link assembly is shown Figure (4.5). Centinkunt and Book [14]

concluded that the equations of motion can be modified to bt

$$\begin{bmatrix} D'_\theta & D_{\theta P} \\ D_{p\theta} & D_p \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{p} \end{bmatrix} + \begin{bmatrix} C_\theta & C_{\theta P} \\ C_{P\theta} & C_p \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{p} \end{bmatrix} + \begin{bmatrix} G_\theta \\ G_p \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} \theta \\ p \end{bmatrix} = \begin{bmatrix} -k_t & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta - \theta_m \\ p \end{bmatrix} \tag{4.47}$$

$$\begin{bmatrix} I_m \end{bmatrix} \begin{bmatrix} \ddot{\theta}_m \end{bmatrix} + \begin{bmatrix} k_t \end{bmatrix} \begin{bmatrix} \theta_m - \theta \end{bmatrix} = \begin{bmatrix} \tau \end{bmatrix} \tag{4.48}$$

where θ_{mi} is the ith motor displacement, θ is the ith link displacement, k_{ti} is the joint stiffness, and I_m is the motor's inertia. Note that D'_θ is a modified version of D_θ where the I_{mi}'s are set to zero.
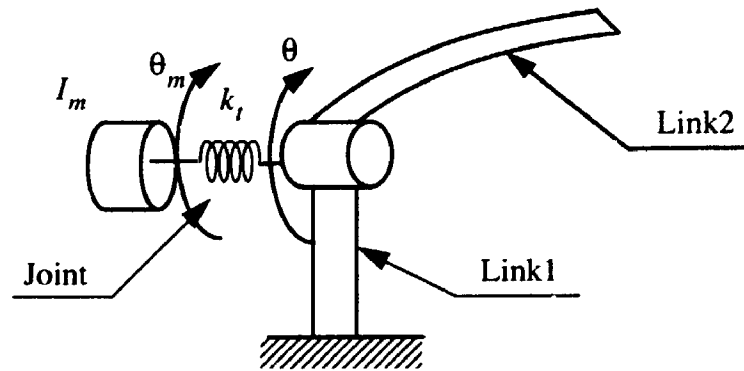
Figure (4.5): The Flexible joint - link assembly.

4.5 The System Frequency Response Functions

After generating the model, the system frequency response functions have to be calculated to give a first approximation to the robot natural frequencies. The state-space form of the equations of motion is linearized around zero-angles configuration (i.e the arm is fully extended and horizontal) using the perturbation techniques [37]. Using the estimated robot parameters shown in Table (3.4), the frequency response functions of the joint angle to the corresponding joint torque are shown in Figures (4.6,4.7,4.8). The system poles and zeros are shown in Table (4.1).
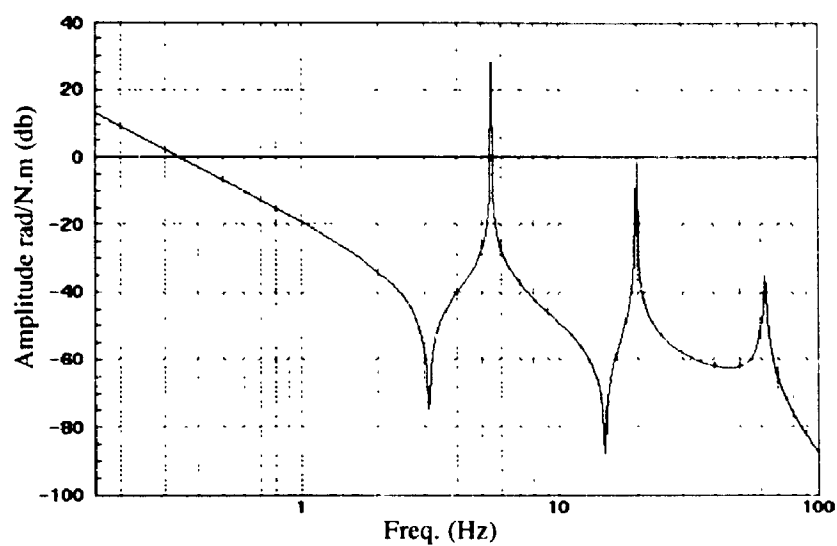
Figure (4.6): Frequency response of the base angle to the base torque.
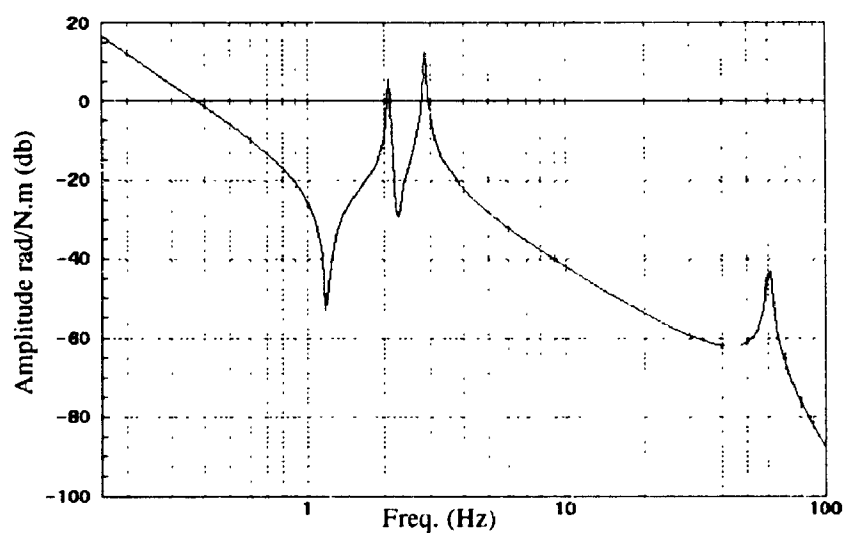
Figure (4.7): Frequency response of the shoulder angle to the shoulder torque.
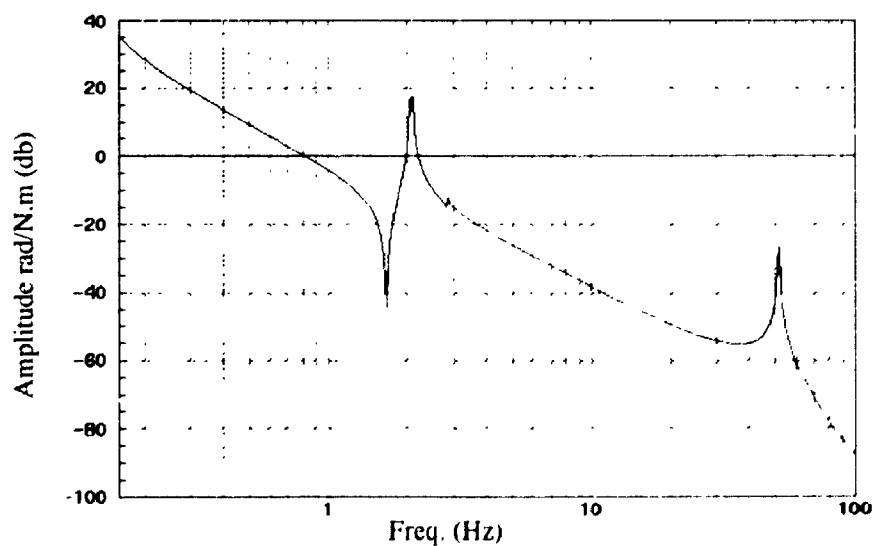
Figure (4.8): Frequency response of the elbow angle to the elbow torque.

Table 4.1: Robot poles and zeros.

Mode No.	Poles (Hz)	Explanation	Zeros (Hz)
1	0	Rigid body (base joint)	1.21
2	0	Rigid body (shoulder joint)	1.97
3	0	Rigid Body (elbow joint)	2.88
4	2.06	first bending (vertical)	10.8
5	2.86	first bending (horizontal)	18.48
6	5.47	second bending (vertical)	21.06
7	19.90	second bending (horizontal)	25.95
8	51.53	Elbow joint flexibility	58.44
9	60.72	shoulder joint flexibility	63.04
10	62.33	base joint flexibility	64.31
11	120.2	torsion mode (link 3)	120.1
12	179.5	torsional mode (link2)	179.4

The system poles are the eigenvalues of the state matrix while the system zeros are the eigenvalues of the constrained (joint locked) state matrix. The constrained system can be obtained simply by closing each torque / corresponding joint angle loop with a very high-gain proportional feedback.

It is shown that for all the joint angles, the joint flexibility starts to have an influence at frequencies higher than 50 Hz. This was attributed to the high stiffness of the drive compared to the links. It is also shown that the links torsional modes occur at relatively high frequency compared to the bending frequencies due to the modelling of the elbow and payload as point masses.

It should be noted that there is some pole-zeros cancelation at the system (e.g. the elbow joint) due to the relatively close values of some poles and zeros.

A final conclusion for this chapter is that the joint flexibility can be neglected in the modelling. Moreover, the analysis bandwidth can be limited to 40 Hz. This will be confirmed experimentally with modal analysis.

- CHAPTER 5 -

SYSTEM IDENTIFICATION

In this chapter, the system parameters are identified experimentally. This includes the modal parameters, structural and joint damping, torque constant and the inertia of the joints. Based on the identification results of this chapter, a reduced order model will be derived in Chapter 6.

The identification techniques, used here, are applicable to a wide range of robotic manipulators including industrial robots (see Zaki et al. [103]).

5.1 Experimental Modal Analysis

In this section, an experimental determination of the natural frequencies, modal damping and mode shapes of the flexible manipulator is accomplished. Note that the modal parameters are dependent on the robot configuration.

By analyzing the results of this test, the following aspects can be accomplished: identifying the resonant behavior of the robot, determining the relevant modes that dominate the robot response, identifying the relative importance of the flexibility of the drive system versus the links, determining the sensitivity of the robot to varying postures and verifying the analytically derived models.

Important assumptions have to be made while carrying out a modal test: the structure is assumed to be linear, homogenous, stable and time invariant [23]. Although

robots may violate these assumptions especially during motion, the vibration characteristics of most robots tend to meet the linear behavior when robots are at a stop condition.

5.1.1 System Setup

To carry out a modal test, care has to be taken in preparing the structure as well as setting up the equipment. The time responses were captured using a spectrum analyzer (HP35655) and the post-processing of the data was carried on SDRC-TDAS modules [84].

Manipulator Setup

The first step for any modal test is a decision on how the structure is going to be supported for testing. In general, it is preferred that the structural is tested in the free condition. White et al. [97] carried out , odal tests for industrial robots (General Electric model GE A4, SCARA type robot). They indicated that there is a little variation in the modal frequencies with servo on/off. However, this conclusion depends on the joint inertia and is valid if high gear reductions are used.

For the experimental manipulator, it was found that implementing different PD gains does not alter the natural frequencies significantly. The results shown in Figure (5.1) were taken from a quick impact test for the robot in the vertical direction. It is shown that due to high inertia at the joints, there is no difference between the FRF wiv, 'ow-gains PD and the brakes locked which represent infinitely high-gains PD. Consequently, a grounded (brakes locked) condition was selected as a test configuration.

The next step for any modal analysis test is to lay out the test points on the structure.

It is known that too few test points will give an inaccurate display of the mode shapes, whereas too many points will make the test very time consuming. The modal test points were chosen to give an accurate display of the mode shapes of the structure as shown in the stick diagram (Figure (5.2)). The total number of test points was 72.

Equipment Setup

A major point in modal testing is how to excite the structure. Initially, modal tests were carried out by exciting each motor at a time with a random or sine sweep excitation. Although it seemed logical to use this method, it gave very noisy frequency response functions due to the chatter of the harm.onic drives and the stiction.

The excitation was given from an electrodynamic shaker which gave a relatively clean spectrum. The driving signal for the excitation was a continuously random or "white noise" signal. The main advantage of using white noise is that the excitation power is evenly distributed over all the frequencies. The excitation signal was generated using the source module of the spectrum analyzer.

The equipment setup is as shown in Figure (5.3) and the list is in Table (5.1)

Table 5.1: List of equipment used for modal testing.

Instrument	Maker	Serial No.
Force Transducer	B & K	8200
Accelerometeis	B & K	4374 #2
Charge amplifiers	B & K	2651
Electromagnetic shaker	B & K	4810
Impact hammer	B & K	8202
Calibration exciter	B & K	4294
Spectrum analyzer	HP	35655
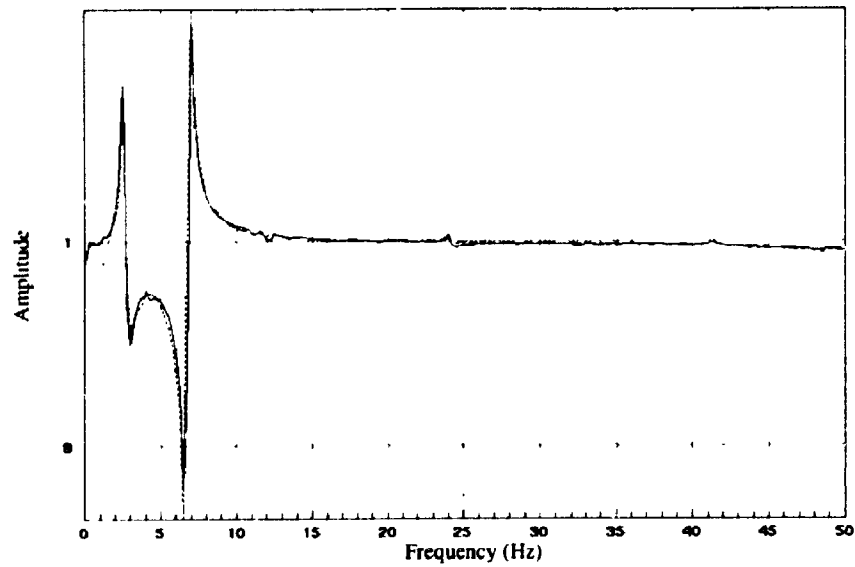
Figure (5.1): Variation of the natural frequencies with the PD gain.
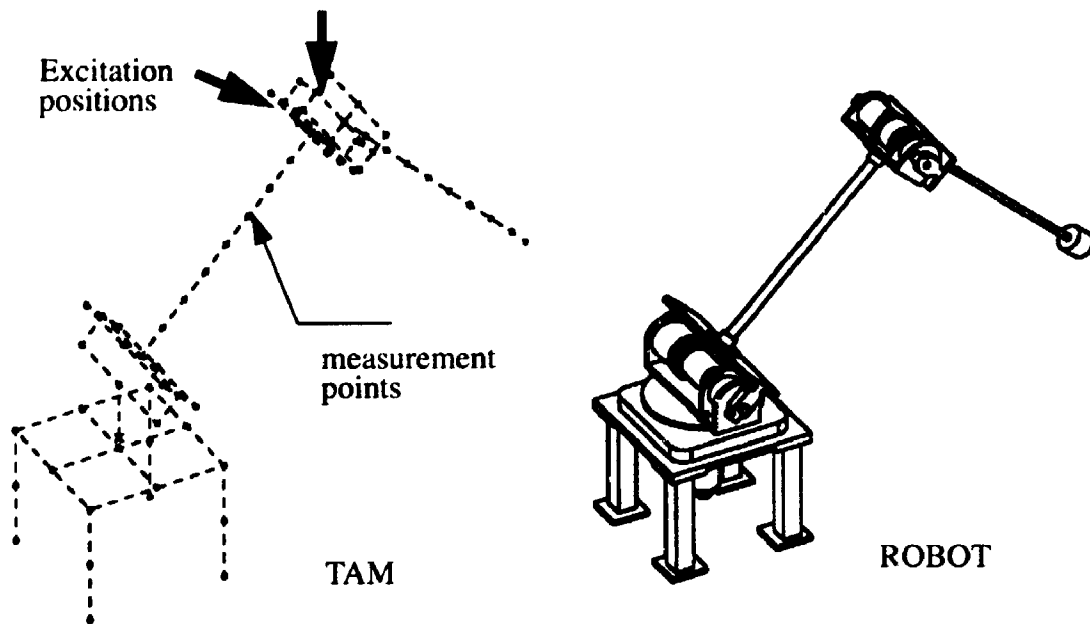
Figure (5.2): A comparison between the test-analysis-model (TAM) and the robot.
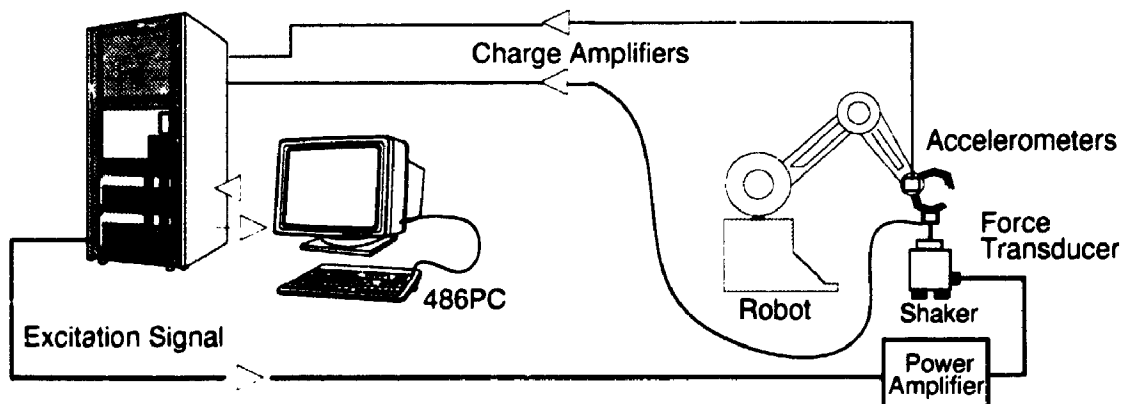
Figure (5.3): The modal test setup (simplified).

5.1.2 Modal Test Results

The modal test was carried out by exciting the robot in the horizontal and vertical directions. Two configurations for the robot were tested: a straight horizontal configuration that represent maximum inertia and a typical operating condition (45 degrees for the shoulder and 90 degrees for the elbow angle (elbow up)). Also, another series of tests were carried out on the robot while the payload was detached.

Frequency Response Functions (FRFs)

The frequency response function for all the test points with respect to the excitation point has to be acquired and analyzed. Depending on the test location, various numbers of averages were taken in order to suppress the noise. At locations near the areas of maximum mobility, such as the end-effector, as few as ten averages were needed. However, as many as 32 averages have to be taken to get a clean FRF and good coherence while gathering data from the supporting base. An example for a typical coherence function is shown in Figure (5.4). The analyzer was set for 400 lines over a range of 0-100 Hz. This gave a resolution of 0.25 Hz over a time record of 4 sec.
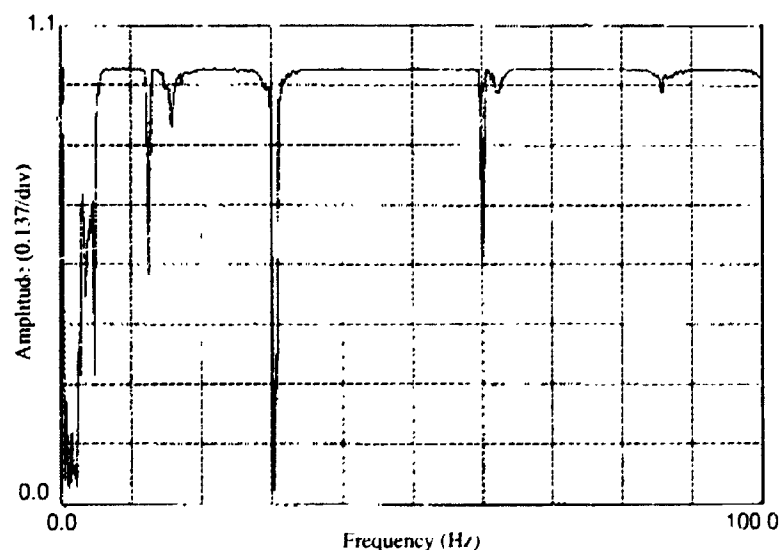
Figure (5.4): The coherence function for a typical measurement.

The FRFs for the end-point with respect to the excitation location for a straight configuration in the horizontal and vertical directions are shown in Figures (5.5). Similarly, for a bent configuration, the FRFs are shown in Figures (5.6). From these plots, it can be shown that the robot has well defined and lightly damped modes.

System Modal Parameters

The structure has lightly coupled modes and small damping factors for which single degree of freedom (SDOF) characteristics can be assumed around the modal frequencies.

Consequently, a SDOF peak-amplitude method [23] was used for estimating the modal parameter. To get accurate results for the damping, a circle-fit technique [23] was used which accounts for slight interference of the other modes. An example of curve fitting is shown in Figure (5.7). It is shown that high correlation is achieved.

The modal parameters and damping for the straight and bent arm configurations in the horizontal and vertical directions are shown in Table (5.2, 5.3, 5.4, and 5.5). The analysis bandwidth was limited to 50 Hz since the contribution of the higher frequencies to the response will be very small. The results of the modal test with no mass were presented for comparison only. Notice the shift in the natural frequency with the arm configuration and with the mass change.

It is shown that the first two modes in the horizontal and vertical directions for the straight arm configuration are close in value due to the arm symmetry. They tend to separate as the arm moves due to the lack of symmetry.

The payload removal seemed to affect the first and second natural frequencies by almost a factor of two, although the mode shapes remained the same. The third mode remained the same since it is due to the torsion of the elbow joint around the first link axis.

(a) Horizontal FRF

(b) Vertical FRF
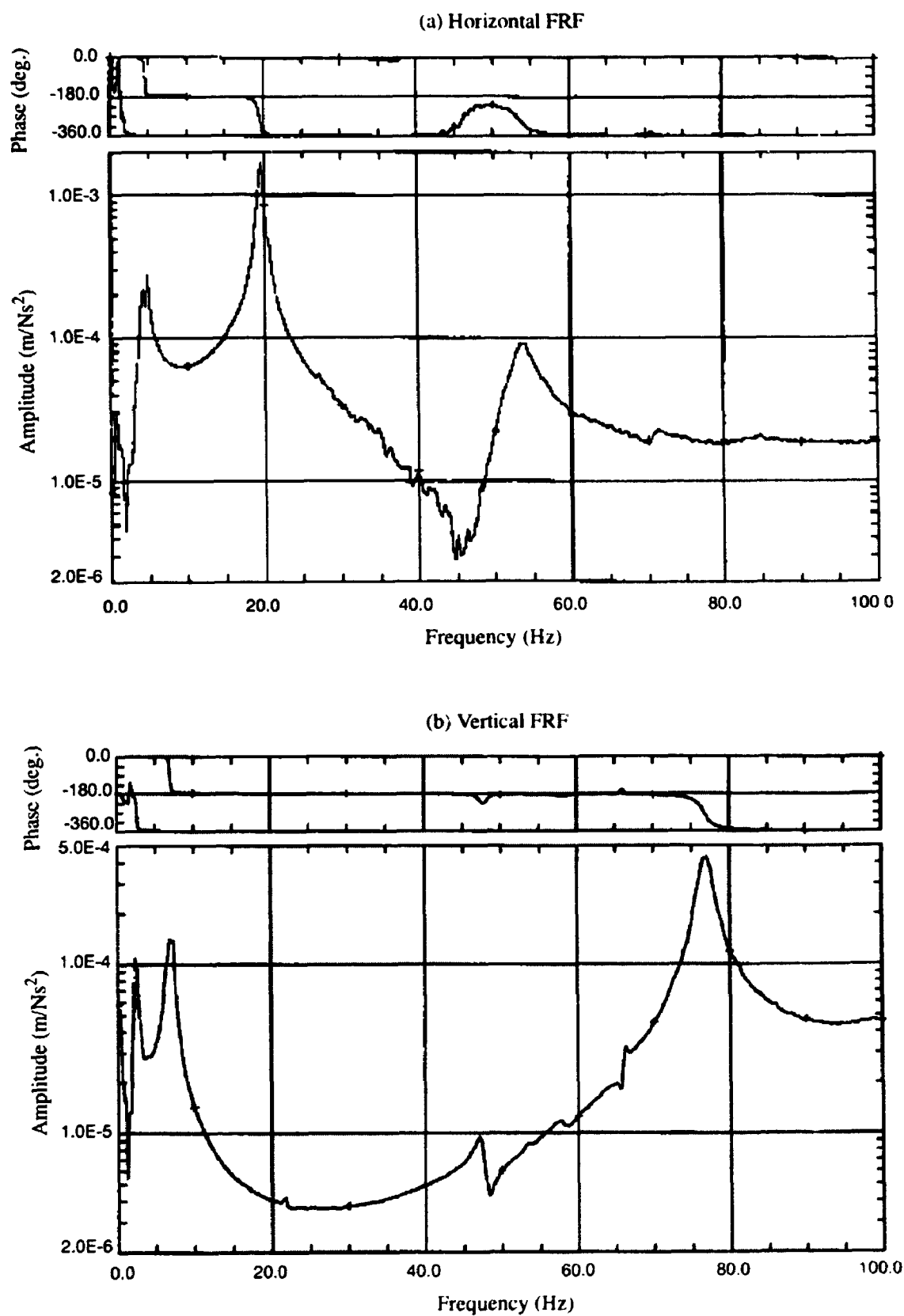
Figure (5.5): End-point frequency response function to excitation points for straight horizontal configuration.

(a) Horizontal FRF

(b) Vertical FRF
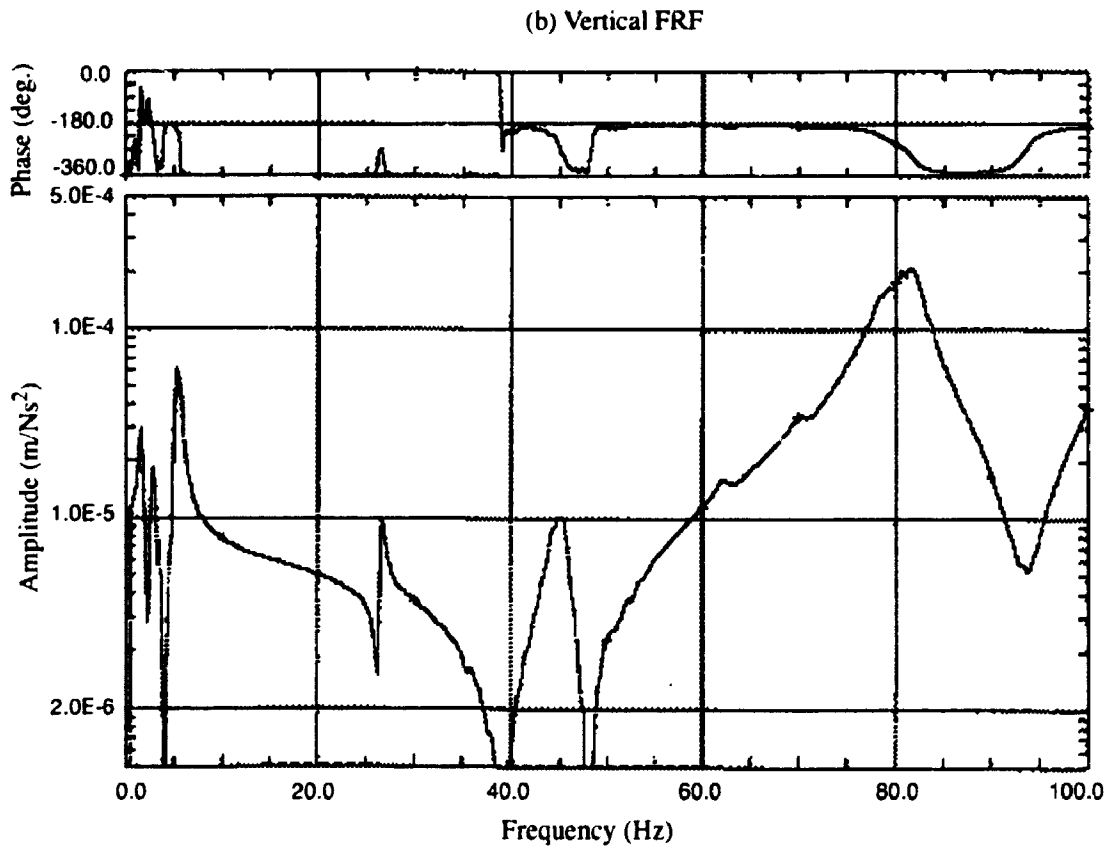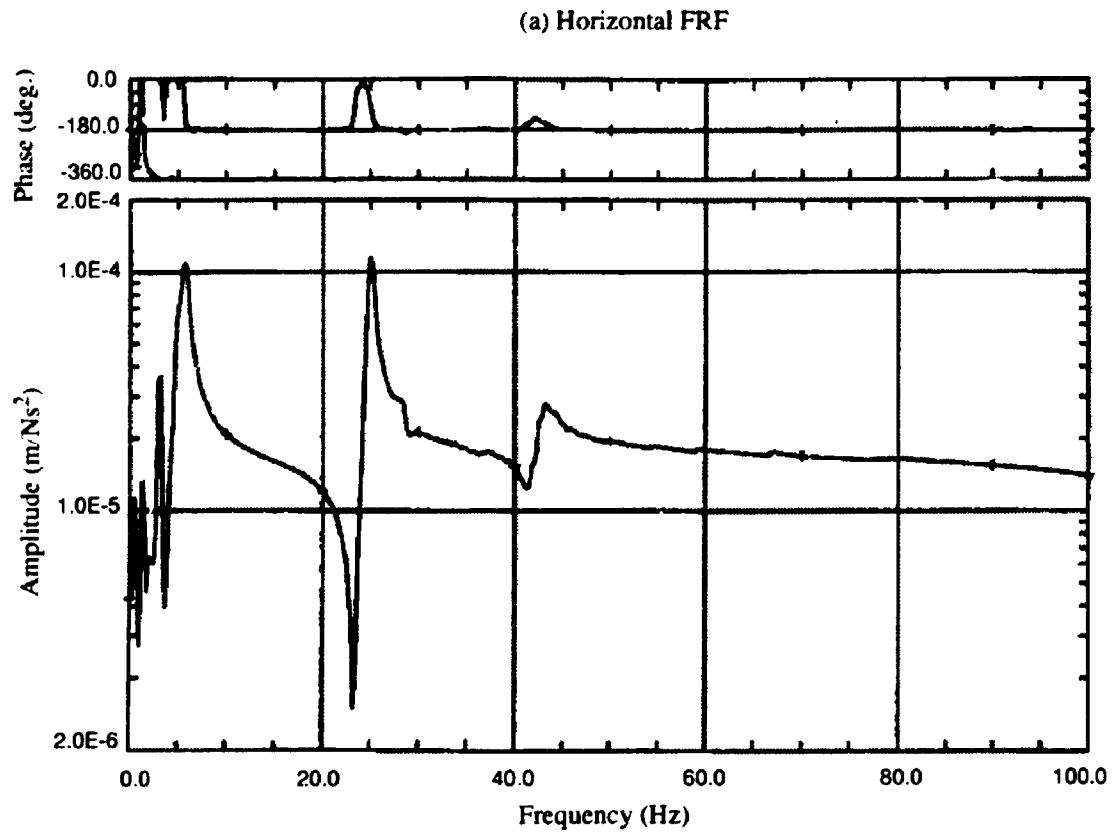
Figure (5.6): End-point frequency response function to excitation points for bent-arm configuration.
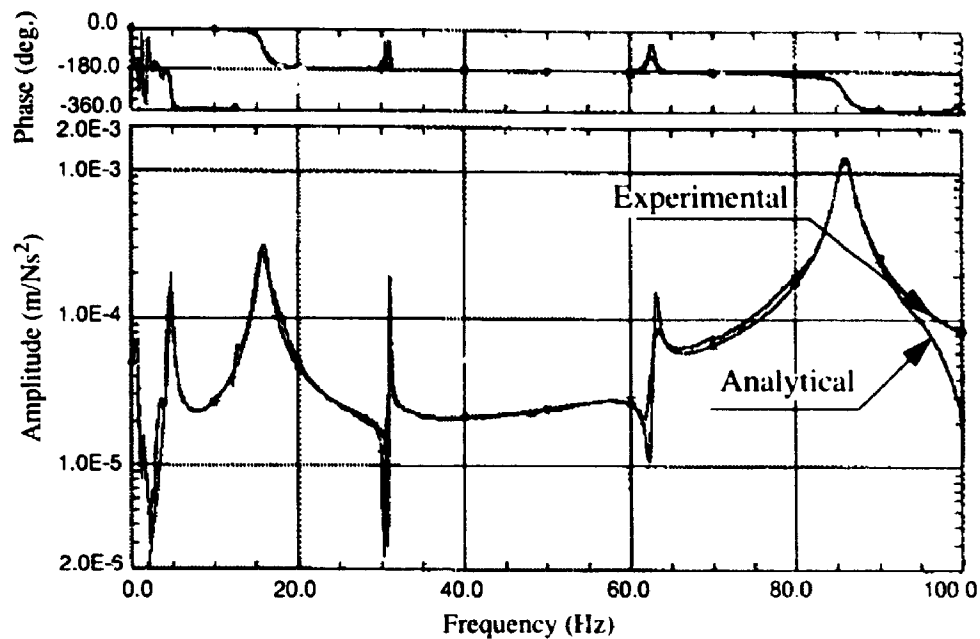
Figure (5.7): An example for curve fitting using SDOF methods.

Table 5.2: Modal par neters-straight configuration - horizontal direction.

Mode No.	Frequency (Hz)	Damping (%)	Explanation	No Mass Nat. Freq (Hz)
1	2.59	0.18	first bending	4.68
2	7.68	1.27	second bending	19.45
3	20.12	0.14	elbow torsion	20.02

Table 5.3: Modal parameters-straight configuration- vertical direction.

Mode No.	Frequency (Hz)	Damping (%)	Explanation	No Mass Nat. Freq (Hz)
1	2.32	0.68	first bending	4.27
2	7.00	1.40	second bending	17.24
3	22.05	0.162	elbow torsion	22.25
4	47.51	1.14	elbow joint + bending	61.05

Table 5.4: Modal parameters-bent configuration - horizontal direction.

Mode No.	Frequency (Hz)	Damping (%)	Explanation	No Mass Nat. Freq (Hz)
1	3.14	0.51	first bending	4.63
2	5.66	0.87	second bending	13.22
3	25.06	0.84	elbow torsion	20.02
4	42.83	0.42	base joint + bending	53.03

Table 5.5: Modal parameters-bent configuration - vertical direction.

Mode No.	Frequency (Hz)	Damping (%)	Explanation	No Mass Nat. Freq (Hz)
1	2.93	0.87	first bending	4.74
2	5.52	1.31	second bending	15.89
3	26.74	0.36	elbow torsion	20.02
4	45.06	1.56	shoulder joint + bending	63.00

5.1.3 Dominant Vibration Modes

As expected, the lowest frequency of vibration occurs when all of the masses are moving in the same direction at the same time. The second frequency of vibration can be monitored as the elbow joint and payload move in the opposite directions. This is shown in Figures (5.8) for straight configuration and Figure (5.9) for bent configuration, where the undeformed configuration is represented by dashed lines.

Animation of the mode shapes was very useful in analyzing the behavior of the robot at the third mode. It was found out that this mode corresponds to the twist of the elbow joint around the first link vertical axis. Since the nature of this mode is torsion, it will be

f = 2.59 Hz

First mode (horizontal direction)

f = 7.68 Hz

Second mode (horizontal direction)

f = 2.32 Hz

First mode (vertical direction)

f = 7.00 Hz

Second mode (vertical direction)
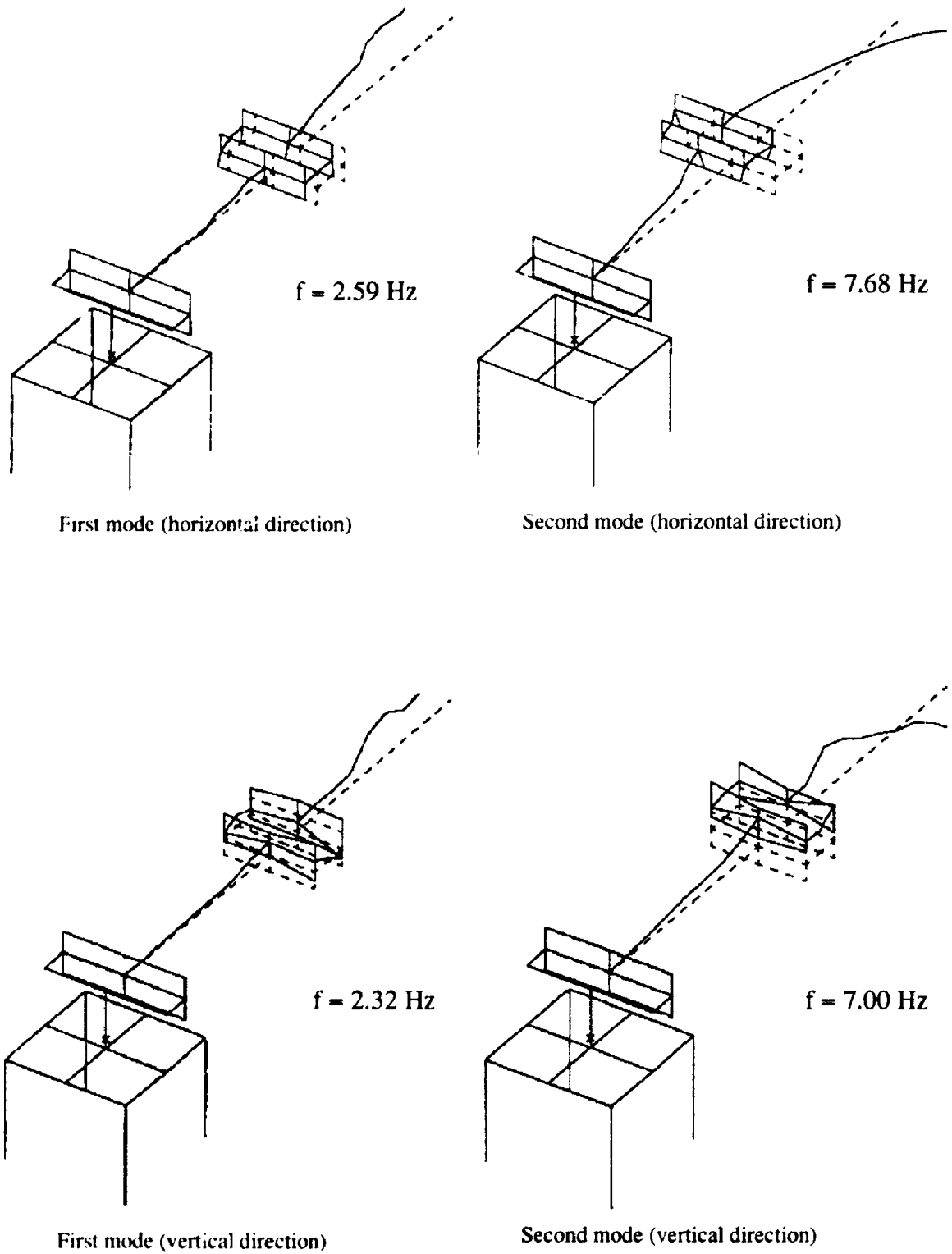
Figure (5.8): Mode shapes display for straight configuration.

First mode (horizontal direction)   f = 3.14 Hz

Second mode (horizontal direction)   f = 5.66 Hz

First mode (vertical direction)   f = 2.93 Hz

Second mode (vertical direction)   f = 5.52 Hz
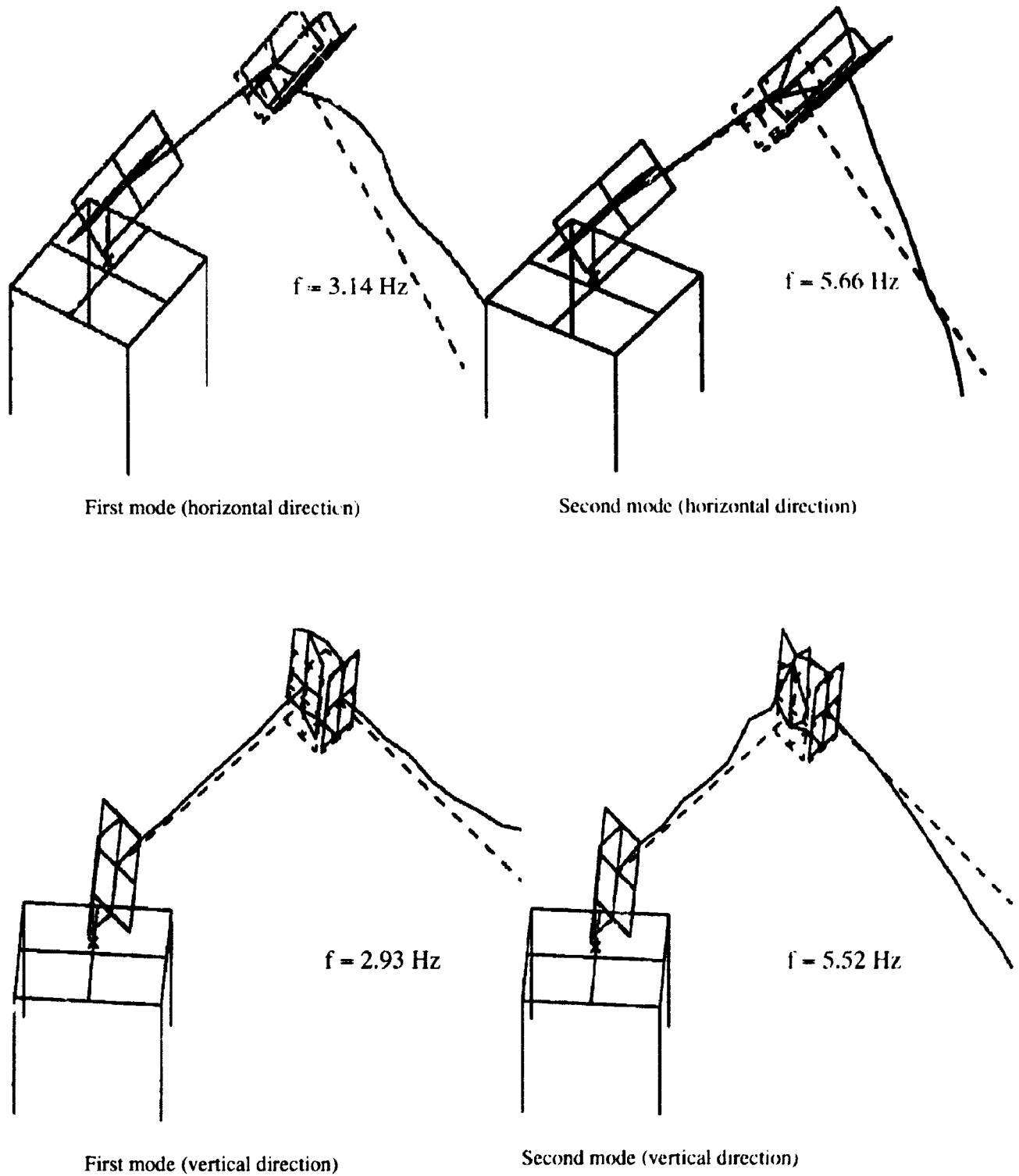
Figure (5.9): Mode shapes display for bent configuration.

excited in both directions. However, this mode has very little influence on the tip response of the third link and it does not even appear in the FRF of the end-point.

Animation of the mode shapes served to identify whether the joint flexibility contributed to the first two modes in both directions or not. It was discovered that for this particular design and set of links, the link flexibility is the primary cause of the end-point vibration. The modes of vibration due to the joint flexibility did not manifest themselves until approximately 40 Hz. This agrees with the analytical results presented in Chapter 4.

Consequently, it can be seen that the first two modes in the horizontal and vertical directions dominate the robot vibration characteristics, and hence controlling these four modes will significantly reduce the tip vibration as will be shown in the following chapters.

5.1.4 Modal Parameters Variation in Space

The variation of the first two modes in the horizontal and vertical directions (the dominant modes) in the robot workspace is studied. This was accomplished by positioning the robot throughout the workspace by 10 degrees increments and analyzing the FRFs of the end-point acceleration to a soft impact hammer hit at the elbow.

The frequencies of the planar vibration depend on the elbow joint variable, whereas the frequencies of the out of plane vibration are functions of the elbow and shoulder joint. Note that the position of the base joint has no influence on the vibration frequencies.

The change of the natural frequencies in the vertical directions with the elbow angle is shown in Figure (5.10). The lateral vibrations are shown in Figures (5.11,5.12). Note that positive elbow angle means elbow-up. These plots give a good idea of how the frequencies change as the arm moves, which indicates how complex the system is.
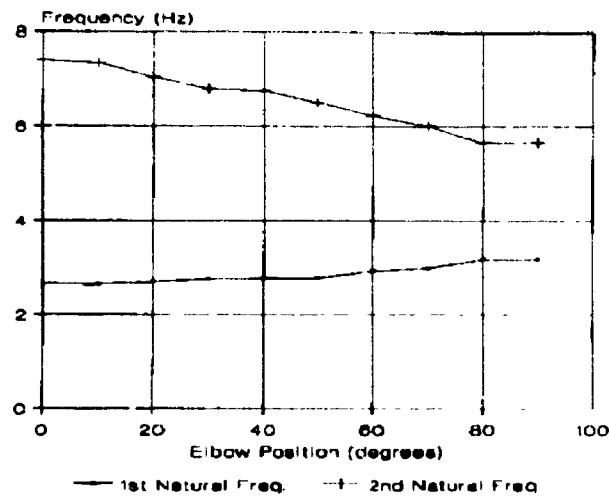
Figure (5.10): Vertical modes variation with the elbow angle.
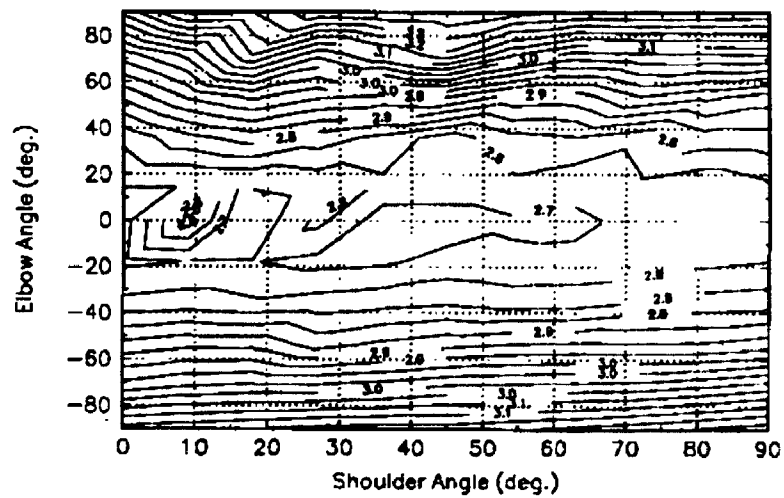
Figure (5.1₁): First Horizontal mode variation with the elbow
and shoulder angles.
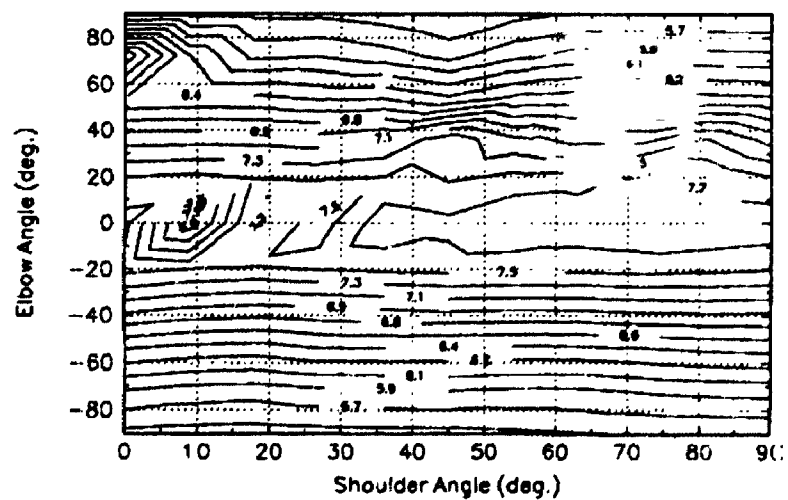
Figure (5.12): Second horizontal mode variation with the elbow
and shoulder angles.

5.2 Torque Constant Estimation

The motor equation, as shown in section 4.2.3 is given by

$$\tau = k_T u \tag{5.1}$$

where k_T is the torque constant from the input voltage u to the output torque τ. Note that there is a bias input volta⌐⌐ to compensate for the friction. The torque constants were obtained by hanging known weights at a known arm distance from the motor and increasing the voltage till the arm starts to move. Then, k_T is obtained by fitting equation (5.1) using least squares. The torque constant for the robot joints are as shown in Table (5.6).

Table 5.6: Robot Torque constants.

Robot joint	Gain
k_T base (N.m/v)	46.75
k_T shoulder (N.m/v)	46.75
k_T elbow (N.m/v)	45.75

5.3 Damping Estimation

Three types of damping were identified in the experimental manipulator: joint coulomb friction, joint viscous damping and links structural damping. The purpose of identifying the damping is to add it to the model. The joint coulomb friction has an adverse effect on the manipulator. The compensation of the friction will be presented in chapter 7.

5.3.1 Structural Damping Estimation

It is known from simulation [2,107] that the structural damping tends to have an

influence on the robot response if the modal damping ratios for the dominant modes is higher than 10%. The structural damping mechanism of the robot is very complex as it changes with the configuration. However, since the robot has very little structural damping, the damping matrix can be assumed to be linear and proportional to the stiffness and mass matrices [23].

The damping matrix can be obtained according to the procedures outlined in [52]. The linearized constrained[1] robot equations can be written as

$$M\ddot{p} + C\dot{p} + Kp = 0 \tag{5.2}$$

where M is the mass matrix, C is the damping matrix, and K is the stiffness matrix. These equations can be diagonalized by performing eigenvalue analysis and substituting for p by

$$p = \psi\eta \tag{5.3}$$

where ψ is the transformation matrix to the modal coordinates. Hence, equation (5.2) becomes [52]

$$M\psi\ddot{\eta} + C\psi\dot{\eta} + K\psi\eta = 0 \tag{5.4}$$

pre-multiplying by ψ^T

$$\psi^T M\psi\ddot{\eta} + \psi^T C\psi\dot{\eta} + \psi^T K\psi\eta = 0 \tag{5.5}$$

$\psi^T M\psi$, $\psi^T C\psi$ and $\psi^T K\psi$ are diagonal matrices.

Pre-multiplying by $(\psi^T M\psi)^{-1}$ we get [52,23]

$$\ddot{\eta} + \hat{C}\dot{\eta} + \hat{K}\eta = 0 \tag{5.6}$$

where $\hat{C} = (\psi^T M\psi)^{-1} (\psi^T C\psi)$ is the modal damping matrix, where

---

1. Constrained equations mean that they do not include the rigid body angles θ.

$$\hat{C} = diag\,(2\zeta_1\omega_1,\, 2\zeta_2\omega_2,\, ...\,) \tag{5.7}$$

Since ζ, ω can be obtained from the modal testing results, the damping matrix is

$$C = (\psi^{-1})^{-1}\,(\psi^T M\psi)\,\hat{C}\,(\psi)^{-1} \tag{5.8}$$

5.3.2 Joint Viscous Damping and Coulomb Friction

Considering the motor model in section 4.2.3. By including the coulomb friction as disturbance, the motor torque τ contains

$$\tau = \tau_d + b\dot{\theta} + \tau_f \tag{5.9}$$

where $b\dot{\theta}$ is the joint effective[2] damping, τ_f is the joint coulomb friction and τ_d is the desired torque to control the robot.

The values of b for the joints can be obtained by running the motor at constant speeds using a DC power supply and measuring the corresponding torque (by measuring the current). Consequently, b can be estimated using least squares fitting of the linear relationship between the voltage and the joint speed. The values for the effective damping b for the robot joints are given in Table (5.7). Note that these values are calculated for the robot side (i.e. multiplied by the gear ratio squared).

Harmonic drives were used as gear reducers for the robot. The mechanism of motion for the harmonic drives is very complex and they tend to have relatively high coulomb friction [3]. The coulomb friction is assumed to be invariant with the angles (it may vary with the joint angle due to the irregularity of the drive). For each robot joint, the coulomb friction can be identified by averaging the measured voltage that corresponds to

---

2. called effective because it is a combination of the joint damping and back e.m.f

the start of the motor motion. The coulomb friction for each joint is given in Table (5.7).

Table 5.7: Joints damping and coulomb friction (robot side).

Robot Joint	Viscous Damping (N.m/rad.s^{-1})	Coulomb Friction (N.m)
Base	13.05	9.35
Shoulder	9.55	7.05
Elbow	2.65	7.78

5.4 Joint Inertia Estimation

The inertia of the motor and gearbox can be approximately calculated using I-DEAS and the manufacturer catalogues. The experimental determination of the joint inertia is done by calculating the time it takes for the motor and gearbox to step to a different constant speed. Considering the joint equation of motion

$$\tau = I\ddot{\theta} + b\dot{\theta} \tag{5.10}$$

Hence,

$$\frac{\dot{\theta}(s)}{\tau(s)} = \frac{1}{Is + b} = \frac{1}{b(\tau's + 1)} \tag{5.11}$$

where τ' is the time taken to reach 0.633 of the steady state amplitude. The values for the identified inertias compared to the estimated using I-DEAS are shown in Table (5.9)

Table 5.8: Joints inertia (robot side).

Robot Joints	Estimated by I-DEAS	Identified Experimentally
Base joint (kg.m^2)	1.683	1.79
Shoulder joint (kg.m^2)	1.638	1.69
Elbow joint (kg.m^2)	1.004	1.1

- CHAPTER 6 -

A REDUCED ORDER MODEL

In this chapter, a reduced order model that includes the dominant vibration modes is derived. The equations of motion are derived using Lagrange's formulation where the links flexibility is represented using finite-element approximation. The developed model is highly nonlinear and will be used subsequently for simulation and controllers design.

A technique for model tuning by matching the locked natural frequencies of the model to the experimental data is applied. This technique results in verifying the reduced order model for the purpose of control.

6.1 Why a Reduced Order Model?

In Chapter 5, it was concluded that the robot dominant vibration modes are the first two modes in the horizontal and vertical directions. Therefore, by controlling these modes, the end-point vibration will be dramatically reduced.

It will be redundant to model vibration modes that do not contribute significantly to the robot response. Hence, a proper model that contains the relevant modes is needed to reduce the simulation time and simplify the controller design.

Finally, a reduced order model is also essential for the development of control algorithms which are based on the inverse dynamics technique. Controllers based on this technique are formulated in the continuous-time domain. Consequently, they have to be

implemented at a sampling frequency higher than (50 - 100) the bandwidth of the system to approximate the continuous-time domain assumption [40,93].

In general, the equations of motion for manipulators with flexible links have a large number of terms that have to be calculated in real-time. Hence, reducing the model order will significantly increase the sampling frequency.

6.2 Modelling Assumptions

In addition to the assumptions posed in Chapter 4, the following modelling assumptions were adopted based on experimental modal analysis results and the physical observation of the robot behavior. These assumption will simplify the development of the equations of motion as will be shown

- **Neglecting the joints flexibility:** It was found that the joint flexibility starts to have an influence on the modes which are higher than 40 Hz. Since it is required to control the modes that are up to 10 Hz, the effect of the joint flexibility on these two modes is very small and can be neglected.

- **Neglecting the links kinetic energy of vibration:** From the robot parameters, it can be seen that the first and second link masses are small in comparison with the elbow and tip mass. Hence, the distributed effect of the links masses can be neglected and both their masses and inertias can be lumped at the end of each link.

- **One mode per link:** Since the first two modes in the horizontal and vertical directions are desired to be controlled, one mode per link will be sufficient for the purpose of end-point vibration control.

• **The elbow and tip masses are point masses:** Since the purpose of the control system is to control the end-point vibration, the kinetic energy due to the rotational vibration of the tip and elbow masses (due to the flexible motion) can be neglected with respect to the kinetic energy due to the rigid body rotation.

Based on these assumptions, it will be shown that a relationship can be formulated between the flexible rotation and the deflections at the links ends. As a result, the flexible degrees of freedom will be significantly reduced.

6.3 Representation of the Links Flexibility

The coordinate systems for the robot are the same as in Chapter 4 (see Figure (4.3)). δ_i and ϕ_i are the deflections and angular deformations of each link with respect to its local frame at the root, where

$$\delta_i = \begin{bmatrix} \delta_{iy} & \delta_{iz} \end{bmatrix}^T \qquad (i = 2,3) \tag{6.1}$$

$$\phi_i = \begin{bmatrix} \phi_{ix} & \phi_{iy} & \phi_{iz} \end{bmatrix}^T \qquad (i = 2,3) \tag{6.2}$$

The elastic deformation of each link is represented by a simple clamped-free beam. The relationship between the force and deformation, taken from finite-element analysis, is

$$\begin{bmatrix} F_i \\ M_i \end{bmatrix} = \begin{bmatrix} K_{i1} & K_{i3} \\ K_{i3}^T & K_{i2} \end{bmatrix} \begin{bmatrix} \delta_i \\ \phi_i \end{bmatrix} \qquad (i = 2,3) \tag{6.3}$$

where,

$$K_{i1} = diag\,(\frac{12EI}{L^3}, \frac{12EI}{L^3}), \quad K_{i2} = diag\,(\frac{GJ}{L}, \frac{4EI}{L}, \frac{4EI}{L}) \quad \text{and} \quad K_{i3} = \begin{bmatrix} 0 & 0 & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & 0 \end{bmatrix}$$

J and I are the polar and area moment of inertia of link i.

6.4 Development of the Equations of Motion

Similar to Chapter 4, the equations of motion were derived using Lagrangian dynamics [101]. Youshikawa et al. [99] developed a similar model. Using the same conventions as Section 4.4.3, the rotation and transformation matrices from the coordinate system $(XYZ)_O$ to $(XYZ)_i$ are

$$^0T_2 = {}^0R_1 {}^1R_2 \tag{6.4}$$

$$^0T_{e_2} = {}^0R_1 {}^1R_2 {}^2R_{e_2} \tag{6.5}$$

$$^0T_3 = {}^0R_1 {}^1R_2 {}^2R_{e_2} {}^{e_2}R_3 \tag{6.6}$$

where,

$$^0R_1 = rot(z_0, \theta_1)$$

$$^1R_2 = rot(x_1, 90) rot(z_2, \theta_2)$$

$$^{e_2}R_3 = rot(z_{e2}, \theta_3)$$

$$^2R_{e_2} = rot(z_{e2}, \phi_{2z}) rot(y_{e2}, \phi_{2y}) rot(x_{e2}, \phi_{2x})$$

The translation vectors for the second and third links with respect to $(XYZ)_O$ are

$$d_2 = [l_2, \delta_{2y}, \delta_{2z}]^T \tag{6.7}$$

$$d_3 = [l_3, \delta_{3y}, \delta_{3z}]^T \tag{6.8}$$

The positions of the elbow mass and tip mass with respect to $(XYZ)_O$ are

$$r_2 = {}^0T_2 d_2 \tag{6.9}$$

$$r_3 = {}^0T_2 d_2 + {}^0T_3 d_3 \tag{6.10}$$

Notice the difference between these equations and equations (4.33, 4.34) as the

flexible rotation of m_2 due to the second link and m_3 due to the third link are not included in defining the position of each mass.

The Kinetic and Potential Energies

The kinetic and potential energies are as follows

$$2KE = I_{m1}\dot{\theta}^2_1 + I_{m2}\dot{\theta}^2_2 + I_{m3}\dot{\theta}^2_3 + m_2\dot{r}^T_2\dot{r}_2 + m_3\dot{r}^T_3\dot{r}_3 \tag{6.11}$$

$$PE = \frac{1}{2}\begin{bmatrix}\delta^T_2 & \phi^T_2\end{bmatrix}\left(\begin{bmatrix}K_{21} & K_{23}\\K^T_{23} & K_{22}\end{bmatrix}\begin{bmatrix}\delta_2\\\phi_2\end{bmatrix}\right) + \frac{1}{2}\begin{bmatrix}\delta^T_3 & \phi^T_3\end{bmatrix}\left(\begin{bmatrix}K_{31} & K_{33}\\K^T_{33} & K_{32}\end{bmatrix}\begin{bmatrix}\delta_3\\\phi_3\end{bmatrix}\right) + m_2g^Tr_2 + m_3g^Tr_3 \tag{6.12}$$

where $g^T = [0, 0, -g_0]$, is the gravity vector.

The links masses are lumped at the elbow and tip of the robot. Hence, m_2 is the mass of the elbow plus half of the mass of the second link and third link, m_3 is the mass of the payload plus half of the mass of the third link.

The Equations of Motion

Defining the Lagrangian L to be $L = KE - PE$, the equations of motion can be derived as follows

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{c}_i}) - \frac{\partial L}{\partial \theta_i} = \tau_{\theta_i} \quad (i = 1,2,3) \tag{6.13}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\delta}_i}\right) - \frac{\partial L}{\partial \delta_i} = 0 \quad (i = 2,3) \tag{6.14}$$

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{\phi}_i}) - \frac{\partial L}{\partial \phi_i} = 0 \quad (i = 2,3) \tag{6.15}$$

Since the tip masses are assumed to be point masses, their rotational vibration can be neglected. Hence, the ϕ's can be expressed as a function of the θ's and the δ's. Although the derivation was carried out on MACSYMA, the following demonstrates how the ϕ's are cancelled out of the equations of motion.

The Lagrange's equation of motion for the joint angle i is

$$\tau_i = \sum_{j=i}^{3} \{ m_j (\ddot{r}_j + g)^T \left(\sum_{k=1}^{j} (\frac{\partial}{\partial \theta_i}\,^0T_k d_k) \right) \} + I_{mi}\ddot{\theta}_i \quad (i=1,2,3) \tag{6.16}$$

where $m_1 = 0$ for notational convenience, and

$$\ddot{r}_i = \frac{d^2}{dt^2}(^0T_j d_j) \tag{6.17}$$

The Lagrange's equations for the deformations δ_i and ϕ_i are

$$m_3\,^0T_2^T (\ddot{r}_3 + g) + m_2\,^0T_2^T (\ddot{r}_2 + g) + K_{21}\delta_2 + K_{23}\phi_2 = 0 \tag{6.18}$$

$$m_3\,^0T_3^T (\ddot{r}_3 + g) + K_{31}\delta_3 + K_{33}\phi_3 = 0 \tag{6.19}$$

$$m_3 V_3^T (\ddot{r}_3 + g) + K_{23}^T \delta_2 + K_{22}\phi_2 = 0 \tag{6.20}$$

$$K_{33}^T \delta_3 + K_{32}\phi_3 = 0 \tag{6.21}$$

where V_3 is defined as

$$V_3 = \frac{\partial}{\partial \dot{\phi}_2^T}(^0R_1\,^1R_2 \frac{d}{dt}(^2R_{e_2})^{e_2}R_3) \tag{6.22}$$

The flexible rotation angles ϕ's can be expressed as a function of θ and δ from equations (6.16,..., 6.21) as follows: from equation (6.21)

$$\phi_3 = -K_{32}^{-1}K_{33}^T \delta_3 \tag{6.23}$$

Substituting equation (6.23) to equation (6.19)

$$\phi_2 = K_{22}^{-1}V_3\,^0T_3 (K_{31} - K_{33}K_{32}^{-1}K_{33}^T)\delta_3 - K_{22}^{-1}K_{23}^T \delta_2 \tag{6.24}$$

By differentiating equations (6.22) and equation (6.23), the expressions for $\dot{\phi}_i$ and $\ddot{\phi}_i$ can be derived. Consequently, the equations of motion can be obtained independent of ϕ. The final equations of motion become

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + Kq + G(q) = \tau \tag{6.25}$$

where

$$q^T = [\theta_1, \theta_2, \theta_3, \delta_{2y}, \delta_{2z}, \delta_{3y}, \delta_{3z}] \tag{6.26}$$

The procedure for the symbolic derivation, carried out under MACSYMA, is shown in Appendix B. The state-space form of the equations of motion is linked to MATRIX$_X$ according to the framework presented in Section 4.4.5. Note that the arrangement of the states throughout the thesis is

$$x^T = \begin{bmatrix} q^T & \dot{q}^T \end{bmatrix} \tag{6.27}$$

6.5 Model Tuning

The robot model may not be accurate in determining the system natural frequencies due to the uncertainty in some parameters. In this section, a technique is described to tune the robot model so that the natural frequencies of the model match the experimental values.

Section 4.5 describes a technique to calculate the locked natural frequencies for the robot. The idea is to match these values with the experimental values.

Using the control systems package MATRIX$_X$, the locked natural frequencies of the robot are matched to the experimental data by updating the robot parameters using the gradient approach. The procedures for tuning the robot model are shown in the following block diagram (Figure (6.1)).
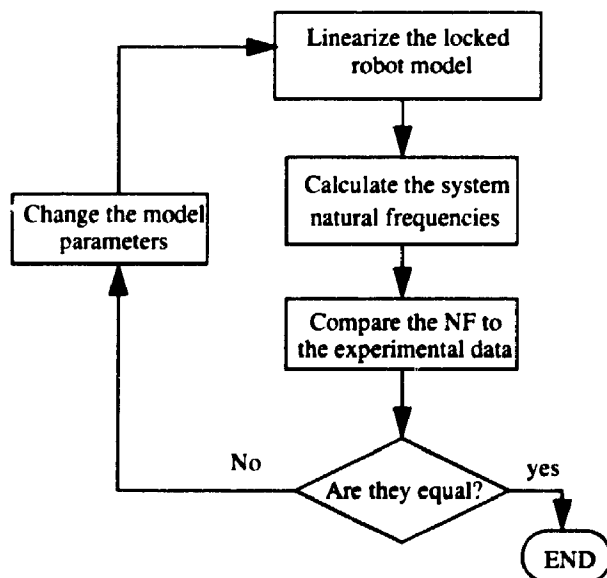
Figure (6.1): Block diagram for the tuning algorithm.

The robot parameters compared to the identified parameters are shown in Table (6.1). The natural frequencies are given in Table (6.2).

Table 6.1: A comparison between the estimated and identified robot parameters.

Parameters	Estimated	Identified
I_{m1} (kg.m^2)	1.683	2.4
I_{m2} (kg.m^2)	1.638	2.2
I_{m3} (kg.m^2)	1.01	1.4
$(EI)_2$ (N.m^2)	1107.	890.0
$(EI)_3$ (N.m^2)	202.2	140.0
$(GJ)_2$ (N.m^2)	873.4	721.0
$(GJ)_3$ (N.m^2)	160.	150.
m_2 (kg)	7.3	9.1
m_3 (kg)	1.7	1.9

Table 6.2: A comparison between the meas.·red and predicted lockeu natural frequencies.

Mode. No.	Measured (Hz)	Predicted (Hz)
1. (vertical)	2.32	2.34
2. (horizontal)	2.59	2.59
3. (vertical)	7.0	6.90
4. (horizontal)	7.68	7.90

It should be noted that the identified parameters are higher and unreasonable sometimes. This is attributed to the low order of the model and its simplicity. However, this "identified" model is suitable for control design. Better matching of the locked natural frequencies and robot parameters can be obtained by increasing the order of the model.

The frequency response functions (FRFs) obtained for the linearized (around zero angles) reduced-order robot model are shown in Figures (6.2,...,6.5). It is shown that each

FRF is a double integrator at low frequencies. Due to the relatively high inertia at the elbow joint, the second zero lies close to the second pole and thus, the corresponding peak at the second pole is reduced.

The level of the FRF between the elbow torque and the base angle indicates the weak coupling between the horizontal and vertical motion.
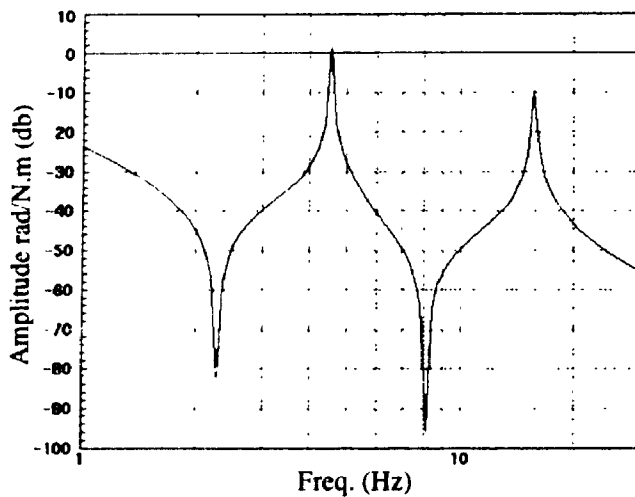
Figure (6.2): Frequency response of the base angle to the base torque.
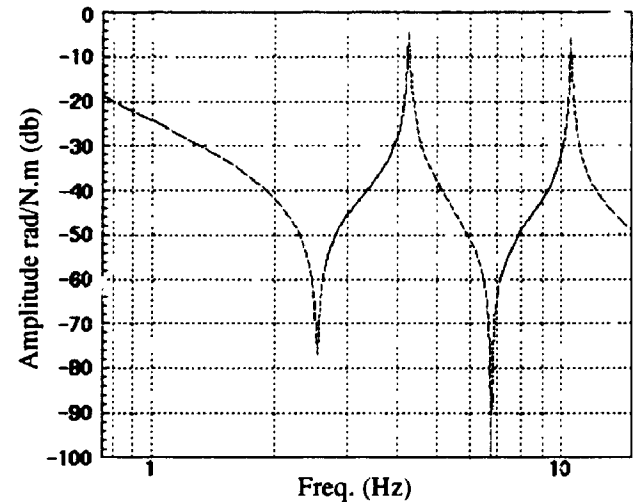
Figure (6.3): Frequency response of the shoulder angle to the shoulder torque.
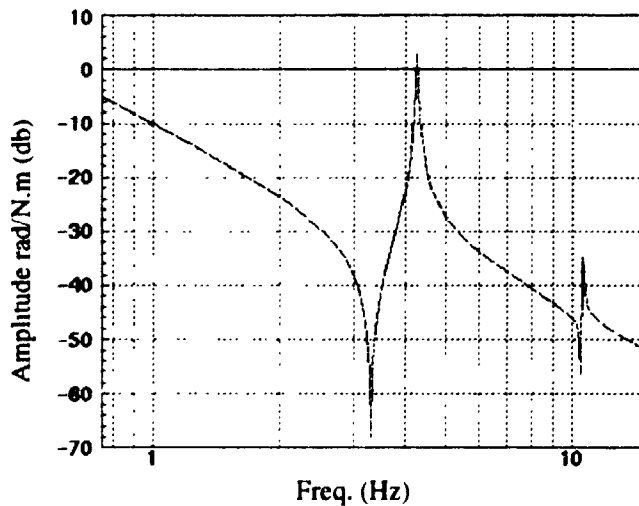
Figure (6.4): Frequency response of the elbow angle to the elbow torque.
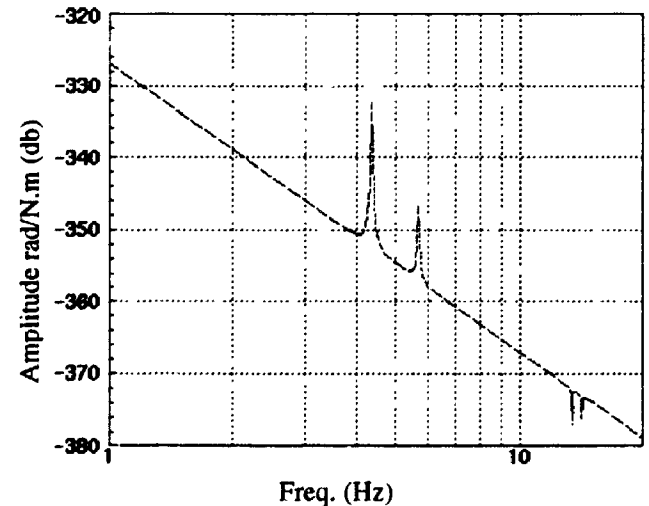
Figure (6.5): Frequency response of the base angle to the elbow torque.

- CHAPTER 7 -

PD CONTROL

In this chapter, the development and implementation of a Proportional-Derivative (PD) control technique will be presented. Unlike rigid robots, it is not sufficient to control flexible manipulators by collocated control algorithms. Consequently, advanced control strategies are needed to achieve better performance.

7.1 The Control Strategy

The block diagram for the PD controller is shown in Figure (7.1). The control input u_i (which implies the motor torque τ_{mi} according to equation (4.10)) for joint i is given by

$$u_i = k_{pi}(\theta_r - \theta) + k_{di}(\dot{\theta}_r - \dot{\theta}) \tag{7.1}$$

where k_{pi} is the proportional gain, k_{di} is the derivative gain for joint i. The control law implemented on the robot uses the angular measurement only as follows

$$u_i = (k_p + k_d \frac{s}{\tau s + 1})(\theta_r - \theta) \tag{7.2}$$

where s is Laplace operator, and τ is the time constant of the derivative filter taken to be (.01-.1). It can be seen from the control law that the coupling between the joints is ignored and the feedback gains for each joint servo are then set to compensate for the coupling in the system. Hence, the resulting feedback gains might have to be large to overcome the coupling in the system and achieve better response. However, the gains are limited due to amplifier saturation and the bandwidth is limited due to system dynamics.
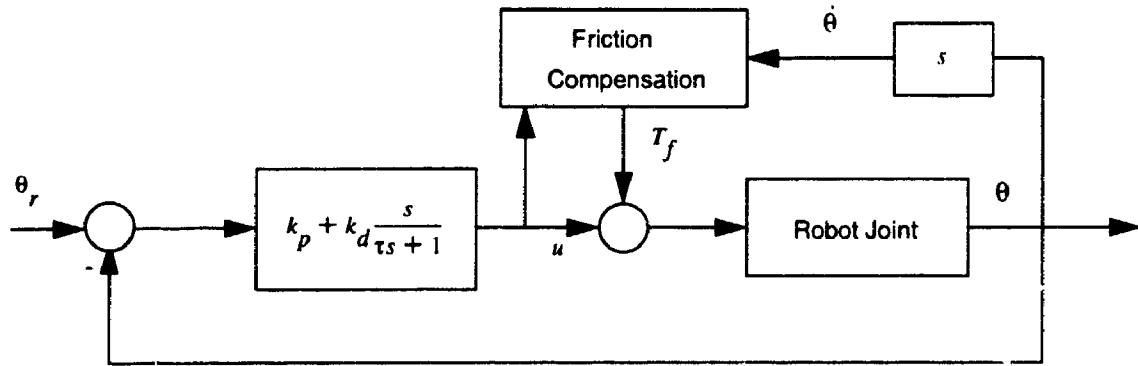
Figure (7.1): Block diagram for an individual robot joint under PD.

Slotine and Li [77] showed that for any passive system, the PD feedback is always stable assuming no phase loss due to sensor /actuator dynamics or computational delays. This is why PD controllers are commonly used for robot control in industry.

7.2 Limitations

In general, PD control is a collocated control technique. Hence the bandwidth will not exceed the first cantilevered frequency. For elastic manipulators, a general rule for PD control is that the achievable bandwidth should be limited to half the fundamental natural frequency to avoid excessive vibration at the tip.

By increasing the gains, the dominant roots for the system will tend to the cantilevered zeros, and the end-point oscillation will decay only under the influence of the structural damping present in the links (which is very small).

This can be illustrated by the simple example of the single link manipulator. The open-loop transfer function of the hub torque τ to the angle θ is given by [71]

$$\frac{\theta(s)}{\tau(s)} = \frac{K_{OL}(s^2 + \Omega^2)}{s^2(s^2 + \omega_n^2)} \qquad (7.3)$$

where one vibration mode is considered only for simplicity, $j\omega_n$ is the pole and $j\Omega$ is the cantilever zero. The characteristic equation for this system under PD control is

$$1 + k_p\left(1 + \frac{k_d}{k_p}\frac{s}{\tau s + 1}\right)\frac{K_{OL}\,(s^2 + \Omega^2)}{s^2\,(s^2 + \omega_n^2)} = 0 \qquad (7.4)$$

Usually, k_d is less than k_p. Consider the ratio $\frac{k_d}{k_p}$ equal to 0.8, and τ to be 0.0 for simplicity. The root locus for the system is as shown in Figure (7.2).

$$\frac{\theta(s)}{\tau(s)} = \frac{23.1\,(s^2 + (2.45)^2)}{s^2\,(s^2 + (10.05)^2)}$$
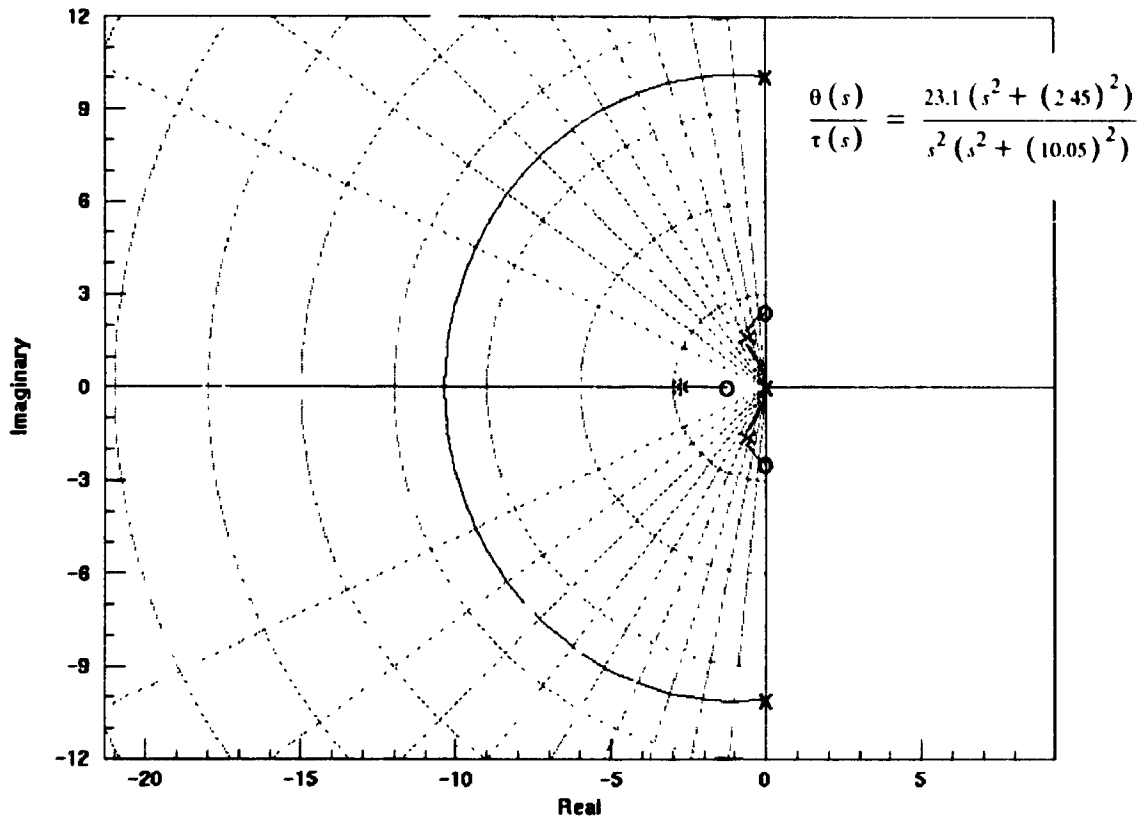
Figure (7.2): Root-locus for a single link under PD.

It is shown that for this particular case, the rigid body roots tend toward the cantilever zeros Ω, whereas the flexible roots tend toward the compensator zero and infinity. It can also be shown that the dominant roots are always "trapped" by the cantilevered zeros. Also, from the iso-damping ratio lines, it is shown that for a maximum damping ratio, the bandwidth will be approximately half the cantilevered frequency.

7.3 PD Controller Design

The successive-loop-closure technique was used to design the PD gains for each joint of the manipulator [71]. The ratio of the derivative to the proportional gains was kept equal to 0.8 for all the joints as it was found to give good results. Using MATRIX$_X$, the open-loop system is linearized around a nominal configuration ($\theta_2 = 40°$, $\theta_3 = -40°$). The PD gains for the base were designed first and closed, and then with the base loop closed, the shoulder gains were designed and closed, and finally, the elbow gains were designed.

The PD gains were chosen to maximize the closed loop bandwidth while maintaining sufficient damping to the closed loop poles. However, there was a trade-off between lowering the damping of the rigid-body poles and the damping of the closed-loop high frequency poles. The open-loop pole and zero locations are indicated by 'x' and 'o' symbol respectively, the location of the chosen closed-loop are indicated by the '*' symbol.

The root locus for the base joint is given in Figure (7.3). The diagonal lines
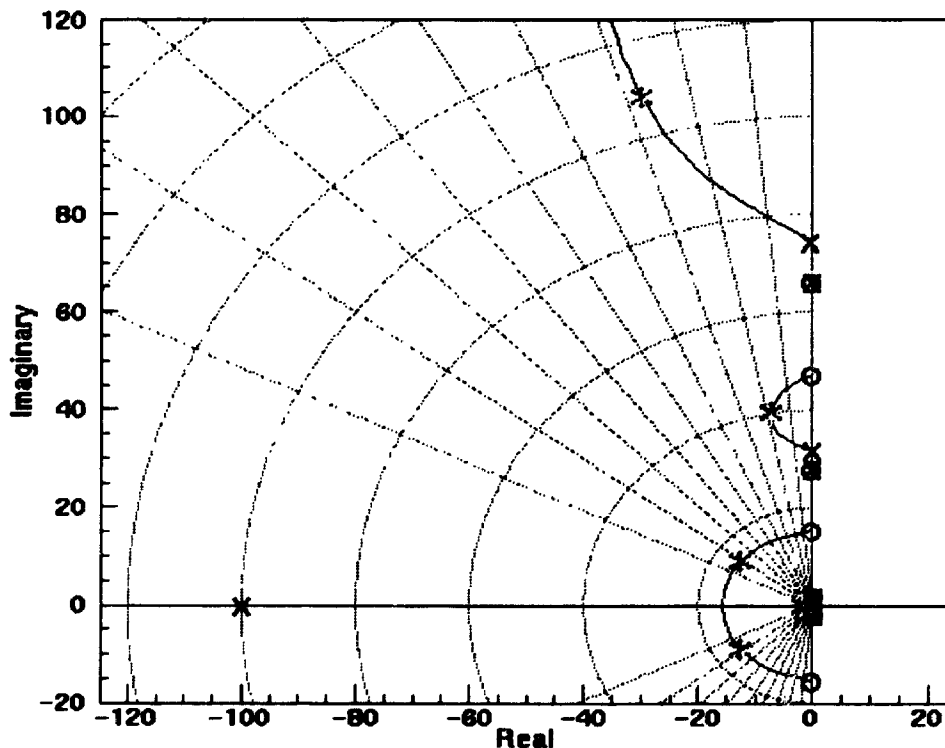
Figure (7.3): PD control design for the base.

represent the constant damping ratio lines. The dominant closed-loop roots are all reasonably damped including small damping at the poles associated to the vibration. It should be noted that the real-valued open-loop poles are due to the differentiator-filter block since τ was chosen to be 0.01.

Closing the base joint, the resulting root-locus for the shoulder joint is given in Figure (7.4). For the subsequently closed loops, the location of their roots is indicated by 'xo' symbol. Note that since there is no coupling between the base and shoulder, the root locus branches do not pass through the poles which affect the base loop.
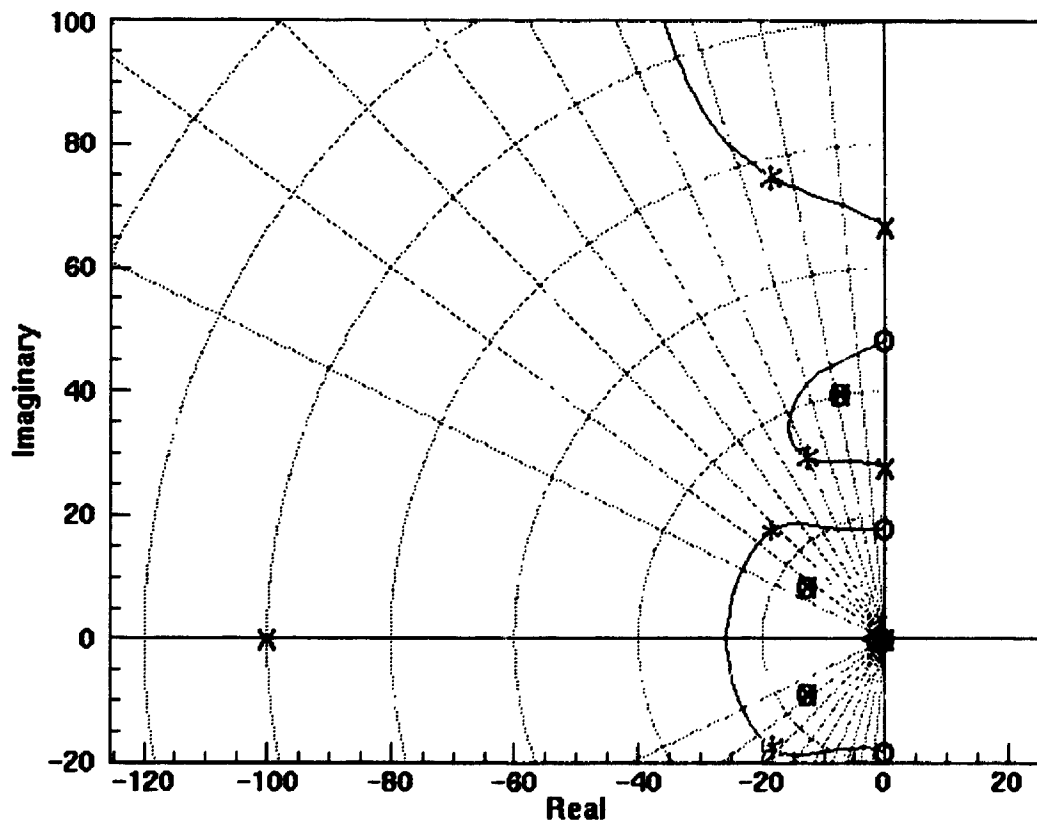
Figure (7.4): PD control design for the shoulder.

The closed loop poles were chosen to give reasonable values for the damping ratios of the poles. Similarly, the root-locus for the elbow joint is shown in Figure (7.5).
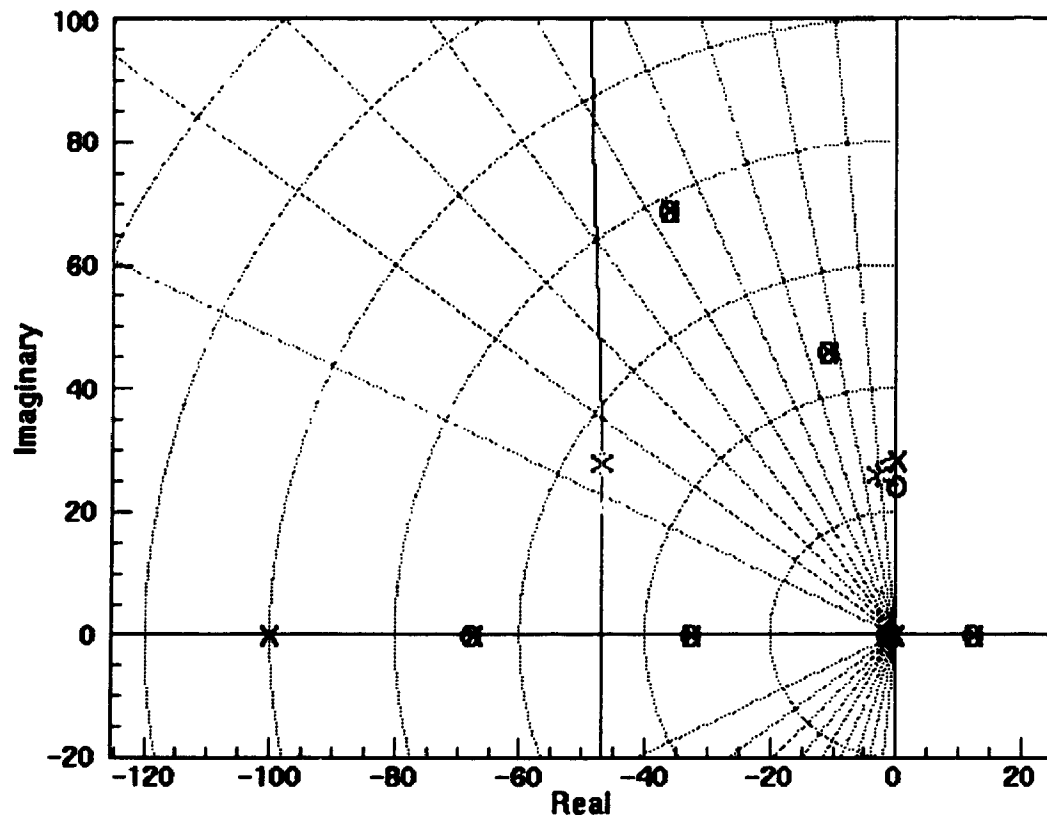
Figure (7.5): PD control design for the elbow.

Finally, the corresponding values for the gains are as shown in Table (7.1)

Table 7.1: The PD controller gains.

Robot Joint	Proportional gain	Derivative gain
Base	2.6	2.08
Shoulder	4.5	3.6
Elbow	2.8	2.24

The closed-loop roots can be obtained by linearizing the system around any arbitrary position. Note that since the system is nonlinear, the roots will vary with the linearization point. Table (7.2) shows the closed-loop roots for a straight horizontal configuration ($\theta_2 = 0°$, $\theta_3 = 0°$) which corresponds to maximum inertia.

Table 7.2: PD-control closed-loop roots.

Pole No.	Root Location (rad/sec)	ω_n (Hz)	ζ
1	-1.21	0.19	1.00
2	-1.45	0.23	1.00
3	-1.58	0.25	1.00
4	-7.26	1.16	1.00
5	$-13.7 \pm 9.37\,j$	2.64	0.825
6	-32.51	5.17	1.00
7	$-8.51 \pm 32.14\,j$	5.29	0.24
8	$-22.71 \pm 35.94\,j$	6.72	0.20
9	$-46.65 \pm 26.36\,j$	8.53	0.87
10	$-16.45 \pm 76.41\,j$	12.44	0.21
11	$-21.32 \pm 124.18\,j$	20.05	0.17

Higher gains are implemented to demonstrate the reduction in the steady-state error in spite of the increase of the tip vibration and the insignificant increase in the bandwidth.

7.4 Step Response

It is shown from Table (7.2) that the closed-loop low-frequency roots which are associated with the vibration are sufficiently damped. In addition, the system time-constant, associated with the rigid-body motion, is large. This results in making the robot motion sufficiently slow to avoid exciting the system flexibility.

The response of the system is tested under a step input. This trajectory can not be implemented in practice due to the infinite change of acceleration associated with it. However, it can demonstrate how fast the system can respond. The response of the robot

joints is shown in Figures (7.6,7.7, and 7.8). The tip deflection of the second and third links

in the horizontal and vertical directions (δ_{iy}, δ_{iz}) are shown in Figures (7.9, 7.10).
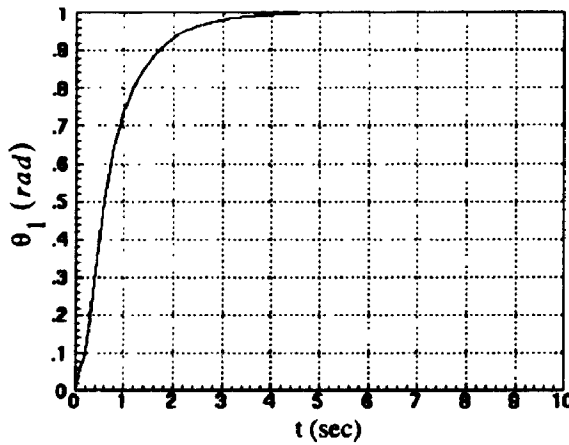
Figure (7.6): Step response of the base under PD.
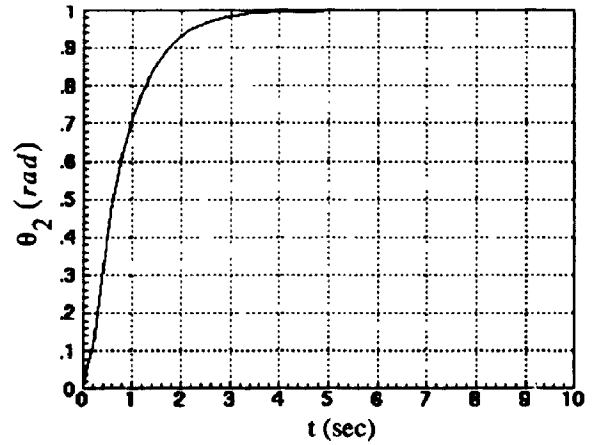
Figure (7.7): Step response of the shoulder under PD.
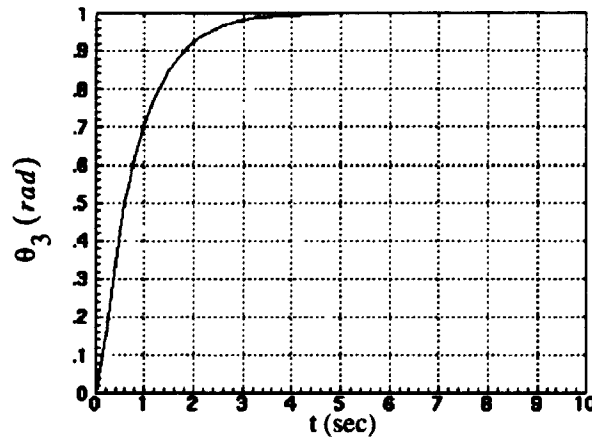
Figure (7.8): Step response of the elbow under PD.
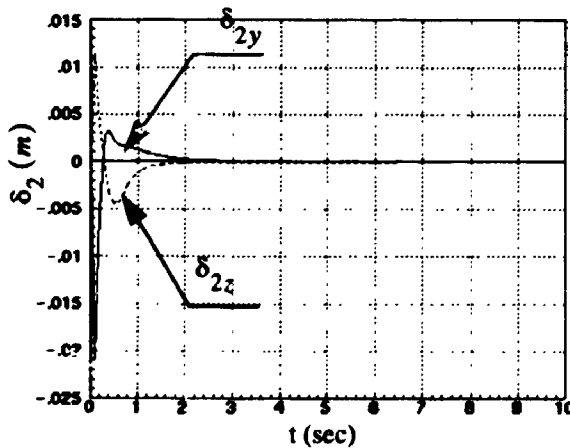
Figure (7.9): Tip deflections of the second link under PD.
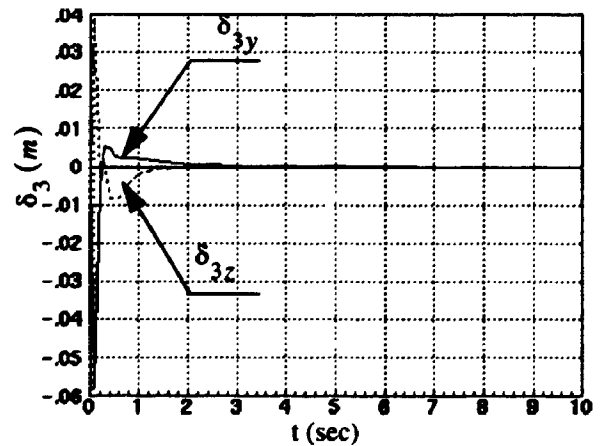
Figure (7.10): Tip deflections of the third link under PD.

It can be seen that the robot takes almost five seconds to reach the desired position.

Note that the simulation is done without taking the gravity and friction into consideration.

If the PD gains are increased to reduce the steady-state error (as in the experimental case),

the speed of response will be almost the same since the bandwidth can not exceed the first

cantilevered zero. However, the links tip will suffer residual vibration due to the reduction

in the damping ratio as seen from the root-locus plots in Section 7.3.

Figures (7.11, 7.12) show the response of the base and shoulder joints under

proportional gains of 5.0, 10.0, 8.0 and derivative gains of 4.0, 8.0, 6.0 respectively. The

tip deflections are shown in Figures (7.13, 7.14).
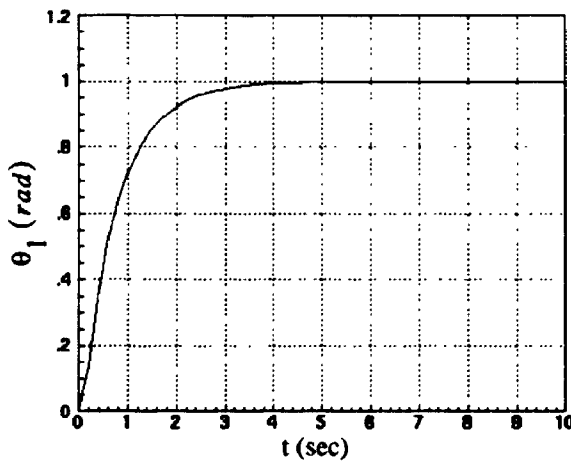
Figure (7.11): Step response of the base under high-gains PD.
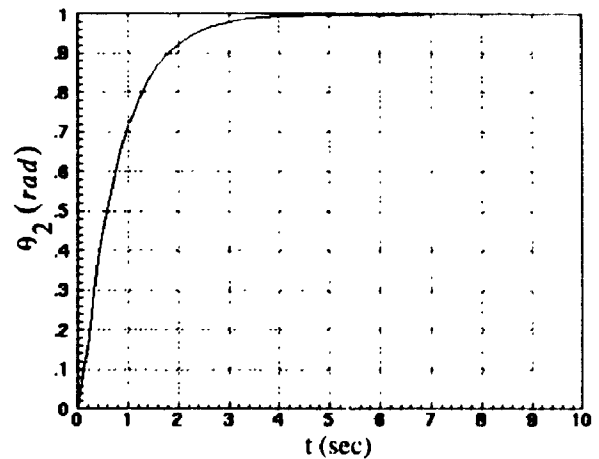
Figure (7.12): Step response of the shoulder under high-gains PD.
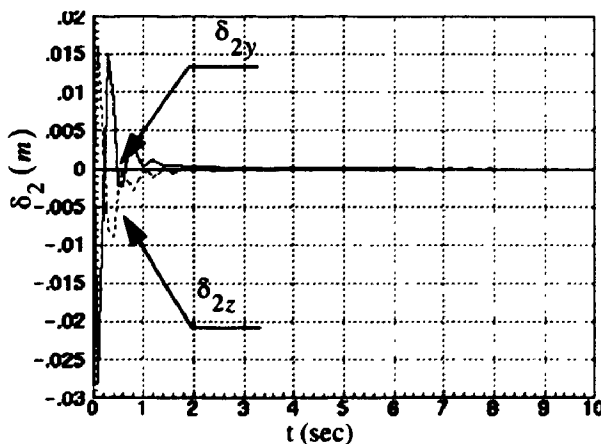
Figure (7.13): Tip deflection of the second link under high-gains PD.
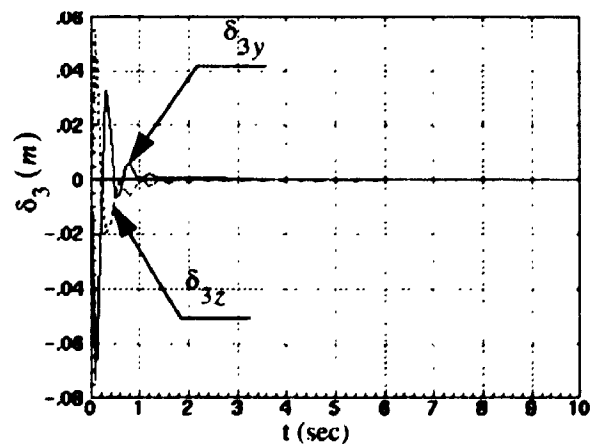
Figure (7.14): Tip deflection of the third link under high-gains PD.

7.5 Implementation Aspects

Implementing the PD controller alone will give poor response and a steady-state error due to the friction and gravity effects. The following demonstrates some experimental aspects that have to be considered while implementing the control algorithm.

7.5.1 Gravity Compensation

The gravity compensation terms are required for the second and third motors in order to improve the transient response and reduce the steady-state error (without using integral terms in the controller). This is accomplished by feeding-forward the equivalent voltages, which result in torques, to balance the gravity.

By lumping the mass of the second and third links at the joints, according to the simplified equations of motion derived in Section 6.4, the equivalent voltages are given by

$$V_{m2} = (-m_2 L_2 g \cos(\theta_2) - m_3 g (L_2 \cos(\theta_2) + L_3 \cos(\theta_2 + \theta_3))) / K_{T2} \qquad (7.5)$$

$$V_{m3} = (-m_3 g (L_2 \cos(\theta_2) + L_3 \cos(\theta_2 + \theta_3))) / K_{T3} \qquad (7.6)$$

where m_2 is the elbow mass plus half of the links masses and m_3 is the tip mass plus half of the second link mass.

7.5.2 Friction Compensation

Friction constitutes the main source of nonlinearity in the robot joints. Friction results in a steady-state error and residual vibration as the flexible arm tends to stop. The nonlinear effect of friction can be dramatically reduced through a model-based compensation mechanism that compensates for the stiction and coulomb friction.

The coulomb friction is compensated for by feeding forward a torque according to the equation

$$T_{fi} = \begin{cases} T_{vi}\,\mathrm{sgn}\,(\dot{\theta}_i) & |\dot{\theta}_i| \geq 0.015 \\ T_{Ti}\,\mathrm{sgn}\,(u_i) & |\dot{\theta}_i| < 0.015 \end{cases} \tag{7.7}$$

where T_{vi}, T_{Ti} are 0.95 and 0.8 of the coulomb friction torque values identified experimentally and u_i is the control action. The compensator was implemented this way to avoid vibration due to input torque commutation in the neighborhood of null velocity and also to eliminate the residual vibration when the arm tends to stop [26].

The stiction effect is reduced by keeping the bearing surfaces from coming to rest with each other. This can be accomplished by applying to the motor a high frequency square wave torque command, called "dither" T_h, that keeps the surfaces oscillating at a very small amplitude [34]. The amplitude was taken so that it would give one encoder count at a frequency of 100 Hz (after several experimental trials) to avoid exciting the system high frequency unmodelled dynamics. The torques values, at the robot side, (which can be transformed to input voltages) for friction compensation are as shown in Table (7.3).

Table 7.3: Friction compensation torques for the robot joints.

	T_v (N.m)	T_T (N.m)	T_h (N.m)
Base joint	11.22	7.95	7.95
Shoulder joint	8.88	7.01	7.01
Elbow joint	9.07	7.26	7.71

7.5.3 Trajectory Generation

Selecting a trajectory generation scheme is an important task in robot motion. Mujtaba [59] studied five different generation schemes, namely, bang-bang acceleration,

critically damped step response, cosine trajectory, sinusoid imposed on a ramp, and polynomial interpolation. It has been concluded that a fifth-order polynomial spline fit gave the best combination of smooth changes in the acceleration and minimal computation.

The coefficients for the polynomial can be obtained by assuming zero velocity, acceleration at the start, end of the trajectory, and by the minimization of the mean square of the jerk over the trajectory (see Hogan and Flash [33]). The time histories for the position, velocity and acceleration are given by [33,34]

$$q(t) = q_0 + (6\tau^5 - 15\tau^4 + 10\tau^3)A$$
$$\dot{q}(t) = (30\tau^4 - 60\tau^3 + 30\tau^2)A/T \qquad (7.8)$$
$$\ddot{q}(t) = (120\tau^3 - 180\tau^2 + 60\tau)A/T^2$$

where, q is one of the coordinates that describes the motion, q_0 is the starting coordinate, A is the amplitude, T is the time for the slew and τ is the normalization time (range from zero to one).

Figure (7.15) shows position velocity and acceleration for a 1-sec rise-time step.
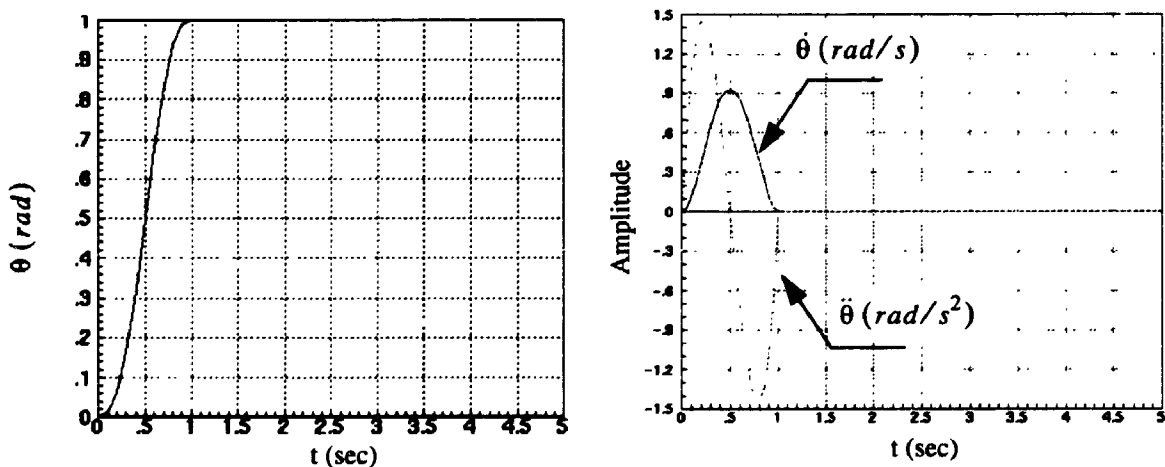
Figure (7.15): Position and velocity for a one-second
rise-time step trajectory.

7.6 Simulation Results

The system is simulated under a step of one-second rise time. The joint angles response are as shown in Figures (7.16,7.17, and 7.18) where the desired trajectory is represented by dashed lines and the actual response is represented by solid lines. The links tip deflections are as shown in Figures (7.19,7.20). It is shown that the robot tracking performance is not good due to the low bandwidth of the system.

Another simulation was carried out with higher gains in order to demonstrate the trade-off between the improvement in the tracking performance of the system and tip vibration. The response of the base and shoulder is shown in Figures (7.21,7.22). The links tip deflections are shown in Figures (7.23,7.24).
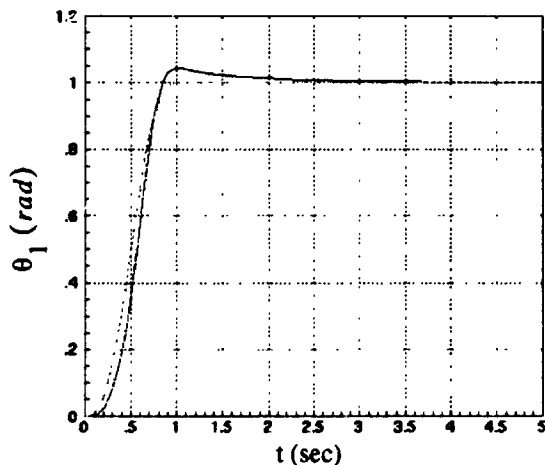
Figure (7.16): Response of the base joint under PD control.
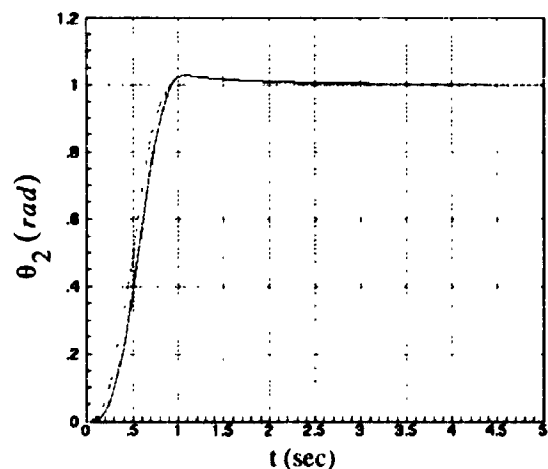
Figure (7.17): Response of the shoulder joint under PD control.
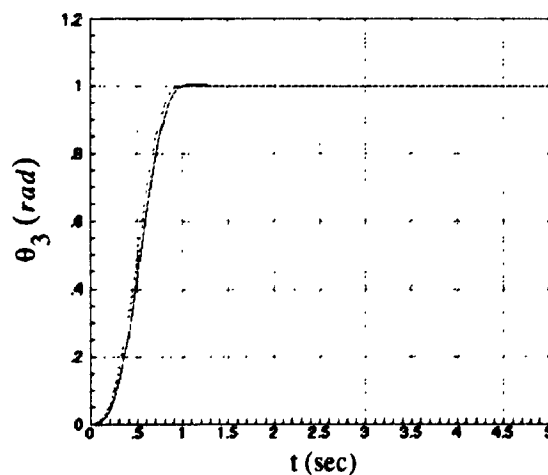
Figure (7.18): Response of the elbow joint under PD control.
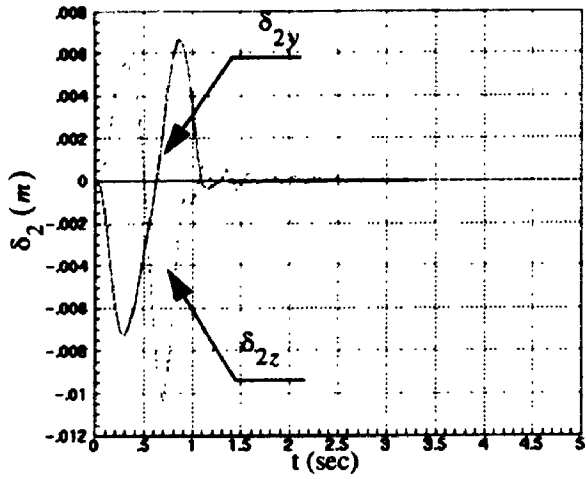
Figure (7.19): Second link tip deflections under
PD control.
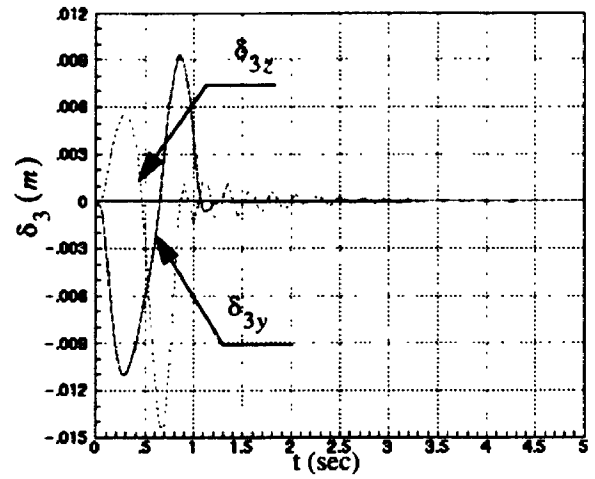
Figure (7.20): Third link tip deflections under
PD control.
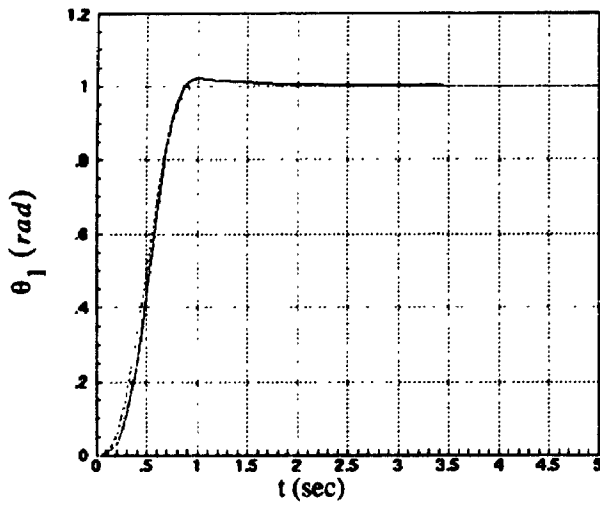
Figure (7.21): Response of the base joint under
high-gains PD control.
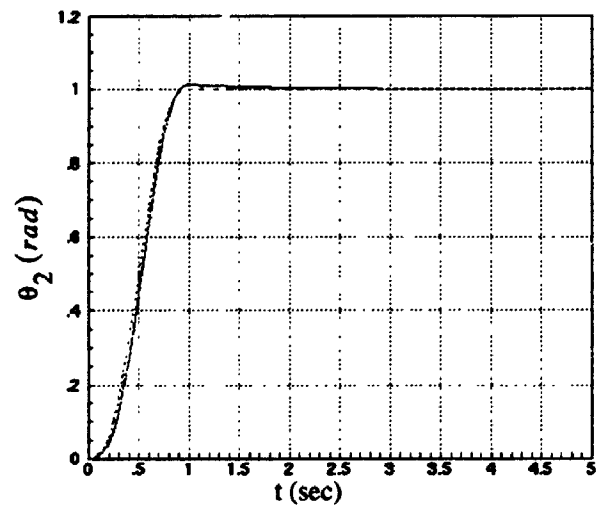
Figure (7.22): Response of the shoulder joint
under high-gains PD control.
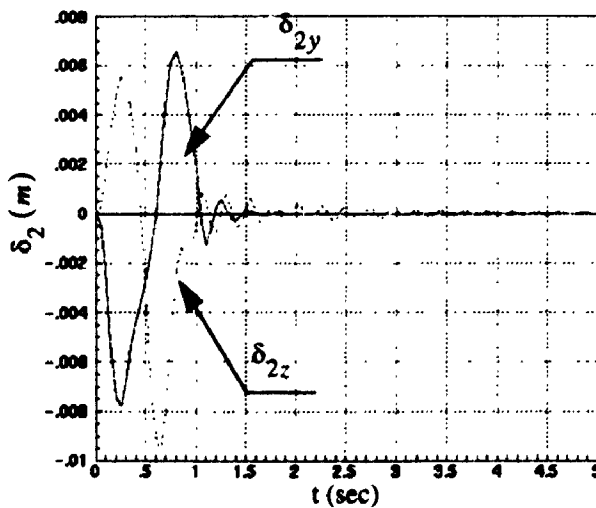
Figure (7.23): Second link tip deflections under
high-gains PD control.
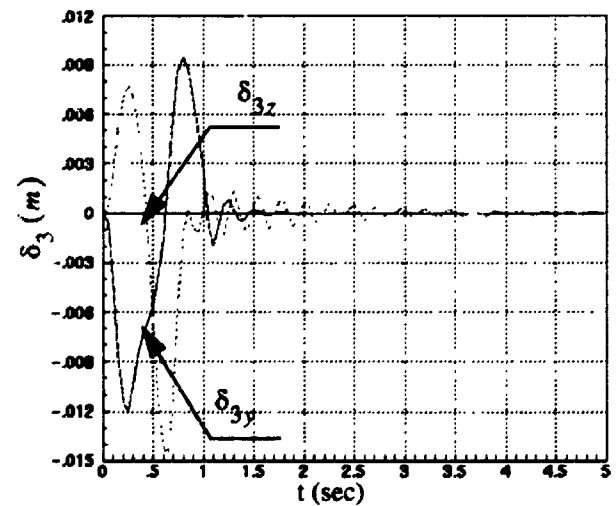
Figure (7.24): Third link tip deflections under
high-gains PD control.

7.7 Experimental Results

The PD controller, with gravity and friction compensation, was implemented on the robot at a sampling frequency of 5 kHz. The estimator, presented in Section 8.4, was implemented along with the PD to estimate the tip deflections. The robot was initially at rest at the home position defined to be straight up. The trajectory was set to be $40°$ step in one second.

The response of the robot joints, under the PD gains shown in Table (7.1), is shown in Figures (7.25, 7.26, and 7.27). The dashed-lines represent the desired trajectory while the solid-lines represent the actual joints motion. It is shown that there is a little delay in the robot response due to the relatively low-bandwidth of the system. There is also a steady-state error at the end of motion due to friction (about 2% to 5%).

The deflection at the tips of the links (in mm) are shown in Figures (7.28 and 7.29). It is shown that the robot undergoes residual vibrations (after the 1 sec. maneuver) that are unacceptable and take a long time to damp out. The torque at the shoulder is shown in Figure(7.30). The chatter presented in the torque signal is for friction compensation.

The response for high-gains PD controller (as presented in Section 7.4) is shown in Figures (7.31 to 7.35). It is shown that with high-gains, the joints tracking is better and the steady-state error is very small. However, the tip vibrations are excessive and take a longer time to damp out.

Consequently, a non-collocated controller is required to improve the performance of the robot.
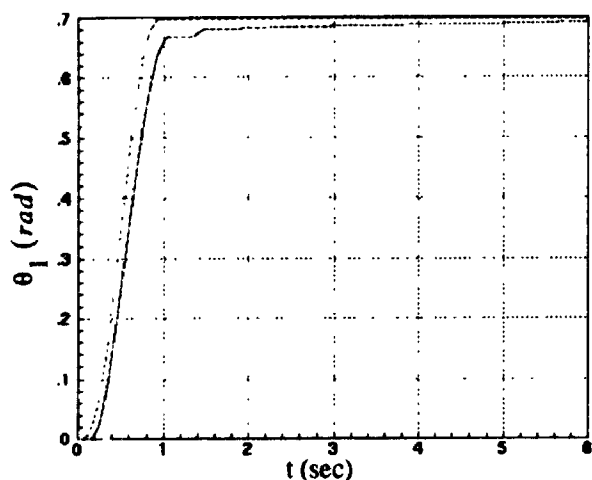
Figure (7.25): Experimental response of the base joint under PD control.
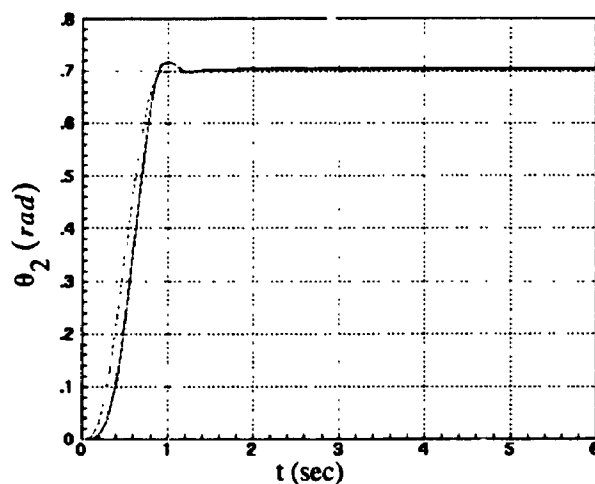
Figure (7.26): Experimental response of the shoulder joint under PD control.
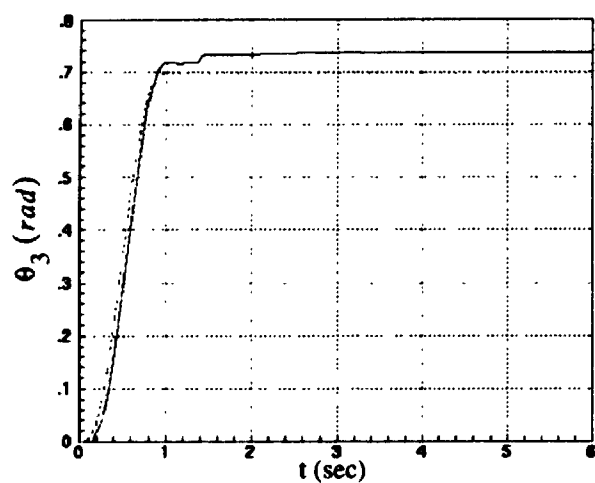
Figure (7.27): Experimental response of the elbow joint under PD control.
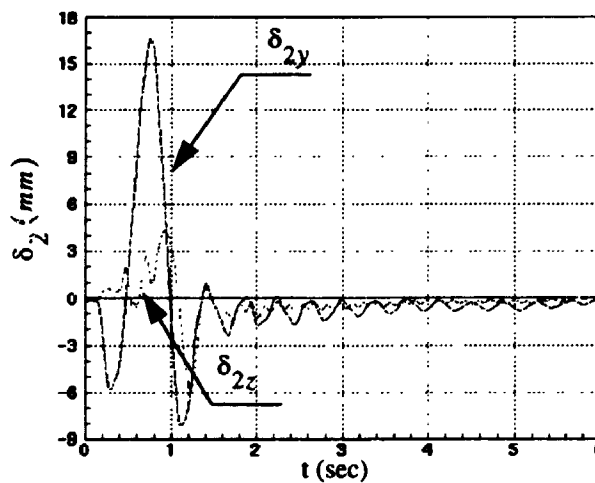
Figure (7.28): Experimental tip deflections of the second link under PD control.
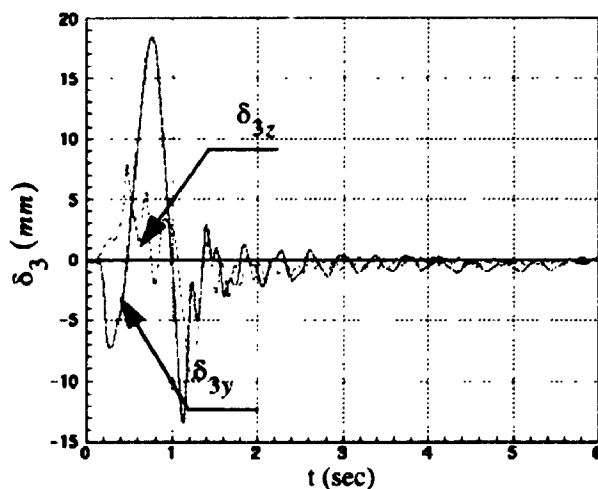
Figure (7.29): Experimental tip deflections of the third link under PD control.
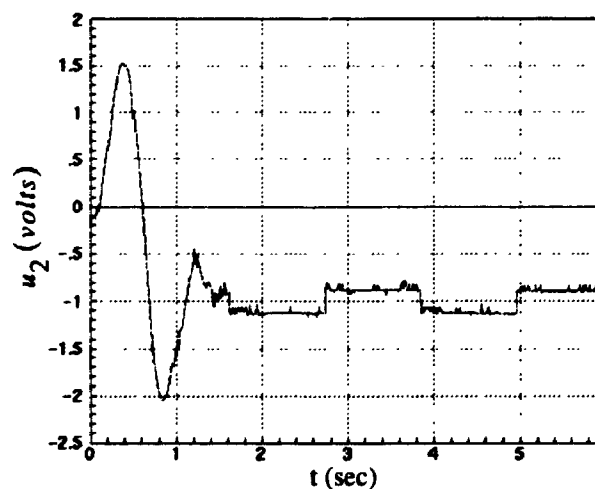
Figure (7.30): Shoulder-motor input voltage under PD control.
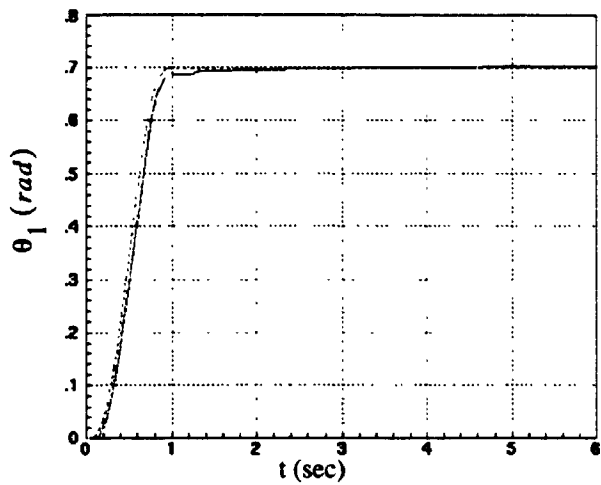
Figure (7.31): Experimental response of the base under high-gains PD control.
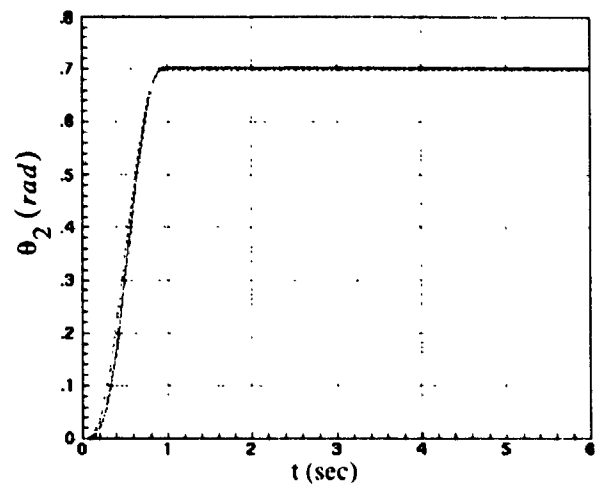
Figure (7.32): Experimental response of the shoulder under high-gains PD control.
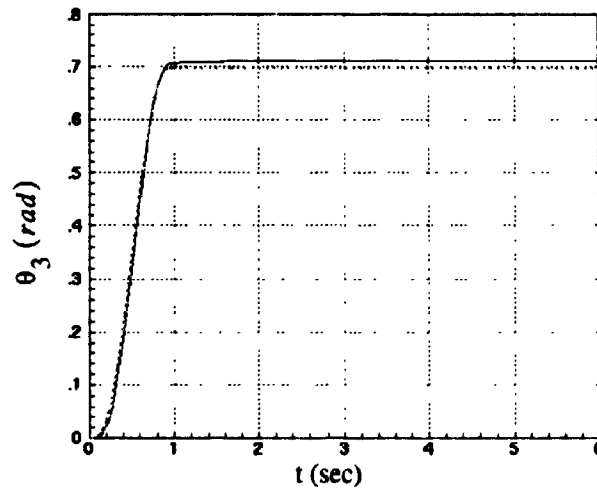
Figure (7.33): Experimental response of the elbow under high-gains PD control.
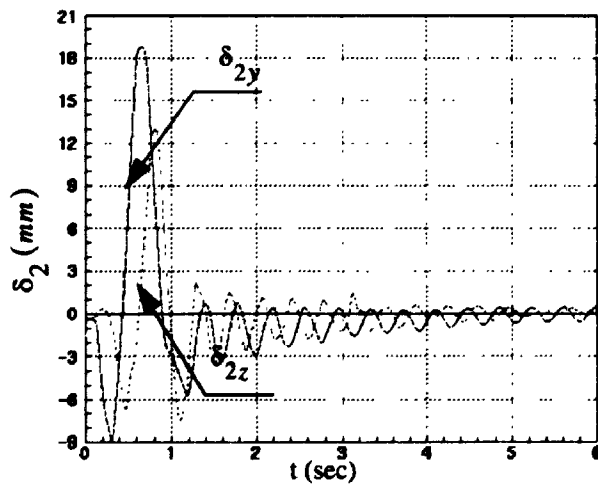
Figure (7.34): Experimental tip deflections of the second link under high-gains PD.
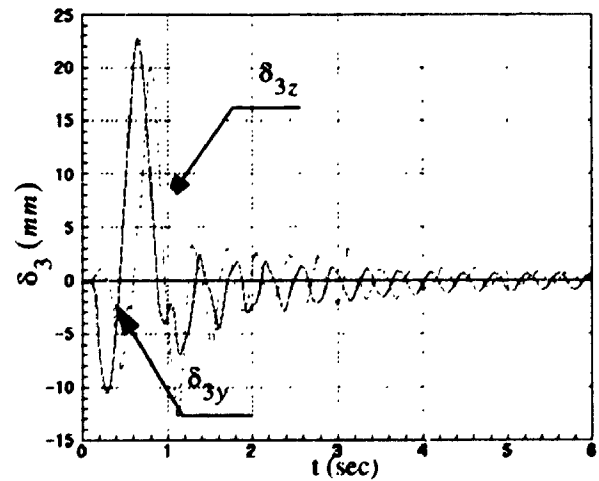
Figure (7.35): Experimental tip deflections of the third link under high-gains PD.

- CHAPTER 8 -

OPTIMAL REGULATOR CONTROL

This chapter presents the first trial toward the objective of the thesis which is controlling the end-point vibration of the manipulator. The vibration measurement is accomplished by means of accelerometers attached to the end of each link. The controller that will integrate the vibration measurements with the joint angles will be designed so that it minimizes certain performance criteria using optimal control theory.

8.1 The System Controllability and Observability

Before deriving control algorithms for the flexible manipulator, the system controllability and observability have to be addressed. While the robot joints are always controllable and observable due to the presence of an actuator and encoder (sensor) at each joint, the flexible states are not controllable in some positions.

Although the robot is a nonlinear system, the controllability and observability of the flexible states can be carried out on the linearized system since the vibration can be assumed to be linear locally.

8.1.1 Controllability

A dynamic system is controllable if for every $t_0, x(t_0), t_1$ it is possible to find an unconstrained control input $u(\tau)$ $t_0 \leq \tau \leq t_1$ such that $x(t_1)$ is at any desired states [24].

108

For a linear system described by

$$\dot{x} = Ax + Bu \qquad (8.1)$$

$$y = Cx \qquad (8.2)$$

the controllability is equivalent to the algebraic condition [24]

$$rank \left(B \mid AB \mid A^2 B \mid \ldots \mid A^{n-1} B \right) = n \qquad (8.3)$$

For the robot, the system states are controllable except when $\theta_2 = 90°$ as the out of plane (horizontal) vibration cannot be controlled by the base motion.

8.1.2 Observability

A dynamic system is observable in the interval $t_0 \le t \le t_1$ if every $x(t_0)$ can be uniquely determined given $u(t)$ and $y(t)$, $t_0 \le t \le t_1$, knowing (A, B, C) .

For the linear system defined by equations (8.1, 8.2), the observability is equivalent to the algebraic condition [24]

$$rank \left(C^T \mid A^T C^T \mid A^{T(2)} C^T \mid \ldots \mid A^{T(n-1)} C^T \right) = n \qquad (8.4)$$

For the experimental manipulator, the estimation of the flexible states is accomplished by mounting two accelerometers at the end of each link. Throughout the entire space, the system is found to be observable.

Since the system is nonlinear, linear observers [24] cannot be used to estimate the states. Limitations are posed on implementing many nonlinear observers [54] in real time due to the complex nature of the equations of motion of flexible manipulators. In this work, on-line estimation of the states from the acceleration measurements was done as shown in Section 8.4, and Section 8.9.

8.2 The LQR Block Diagram

The Linear Quadratic Regulator (LQR) algorithm was chosen because of its wide use in flexible manipulators control. Several authors tend to use LQR algorithms because they can successfully handle Multi-Input Multi-Output (MIMO) systems with lightly damped poles [61]. However, LQR controllers are sensitive to how well the system is modelled. Also, since the controller is linear, it is only suitable around the design point. Consequently, a gain-scheduling technique is implemented to guarantee optimal performance for all the robot configurations.

The block diagram for the controller is shown in Figure (8.1). It consists of two parts: a state-estimator and a regulator.

The LQR controller works as follows: the estimator, in general, takes the control torques, the joint angles and the links tip acceleration as inputs and outputs the flexible states. The regulator takes the difference between the desired and actual states and outputs the control torques that will bring the system to the desired configuration.

The design of the LQR controller is performed in the continuous-time domain, and then converted to the discrete-time domain for real-time implementation according to the procedures presented in Appendix B.
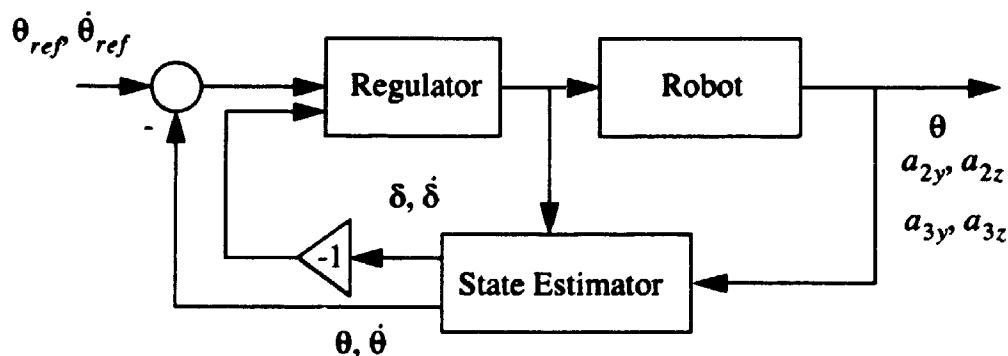
Figure (8.1): Block diagram for the LQR controller.

8.3 Regulator Design

The optimal state feedback problem involves minimizing the continuous quadratic cost function J given by

$$J = \frac{1}{2} \int_0^\infty (x^T R_{xx} x + u^T R_{uu} u)\, dt \tag{8.5}$$

where R_{xx} is the state weighting matrix, and R_{uu} is the control effort weighting matrix. The solution of the LQR problem stated above is to use a controller of the form

$$u = -R_{uu}^{-1} BPx = -kx \tag{8.6}$$

where P is the solution of the matrix Riccati equation [80,24] given by

$$PA + A^T P - PBR_{uu}^{-1} B^T P + R_{xx} = 0 \tag{8.7}$$

The choice of R_{xx} and R_{uu} determines how the regulation error and actuation effort are traded off to reach an optimal design. Several approaches exist to choose the weighting matrices aided by control design packages (e.g. MATRIX$_x$) as follows: R_{xx} and R_{uu} can be chosen iteratively to achieve maximum bandwidth [61], to get certain values of PD gains [99], or to weight each state by how well the control of the state is desired [71].

For the flexible manipulator case, R_{xx} and R_{uu} are chosen by trial and error to achieve high PD gains for the joint angles and best performance in damping the tip vibration experimentally. Note that high gains may work fine in simulation but will produce instability for the experimental manipulator due to the excitation of the system unmodelled dynamics and the amplification of sensors noise. The control weighting matrices were chosen to be (refer to Section 6.4 for the corresponding state representation)

$$R_{xx} = diag\,(\,[\,90,\,10,\,10,\,4,\,0.5,\,8,\,0.5,\,400,\,600,\,100,\,60,\,100,\,100,\,100\,]\,) \tag{8.8}$$

$$R_{uu} = 3 \times diag\,(\,[\,2,\,1,\,0.5\,]\,) \tag{8.9}$$

A particular point of interest is the choice of the manipulator angles around which the system is linearized. Hollars [34] showed that for maximum robustness of the controller against the change of the inertia matrix, the linearization point should be where the inverse determinant of the mass matrix is minimum. This corresponds to an elbow angle of 90° and a shoulder angle of 0°.

The regulator gains are found to be

$$
k = \begin{bmatrix}
4.25 & 0 & 0 & 0 & -0.43 & 0 & -0.06 & 8.17 & 0 & 0 & 0 & 1.22 & 0 & 0.30 \\
0 & 3.33 & -0.08 & 0.37 & 0 & 0.14 & 0 & 0 & 14.1 & -0.49 & -17.68 & 0 & 10.88 & 0 \\
0 & 0.75 & 2.97 & 0.81 & 0 & -0.67 & 0 & 0 & 1.68 & 8.14 & -40.57 & 0 & -15.20 & 0
\end{bmatrix} \quad (8.10)
$$

Note that for this particular manipulator the vertical and horizontal motions are decoupled as seen by the zero locations of the regulator. The closed-loop pole locations, obtained by linearizing the system around ($\theta_2 = 0°$, $\theta_3 = 90°$), are presented in Table (8.1).

Table 8.1: Regulator-control closed loop roots.

Pole No.	Root Location (rad/sec)	ω_n (Hz)	ζ
1	-2.52	0.40	1.00
2	-3.20	0.51	1.00
3	-6.10 ± 4.16 j	1.18	0.83
4	-7.54 ± 19.84 j	3.38	0.35
5	-8.1 ± 20.91 j	3.57	0.36
6	-22.02 ± 29.96 j	5.92	0.59
7	-37.62	5.99	1.00
8	-17.03 ± 59.07 j	9.78	0.28
9	-131.46	20.90	1.00

As shown in Table (8.1), the values of the dominant poles are higher than the corresponding dominant poles of the closed loop system under PD (shown in Table (7.2)). Hence, it is expected that the system response is faster than the PD.

8.4 Estimator Design

In implementing the regulator, the system states are assumed to be known. Since not all the states are measurable, they need to be estimated.

For the flexible manipulator, the joint angles are known from the encoder outputs. The joint velocities can be obtained by filtering and differentiating the joint position signals. The elastic terms δ's are observed by a set of accelerometers attached to the tip of each link and pointed to the Y and Z axes with respect to the links local coordinates.

The accelerometer signals can be integrated on-line to give the tips velocities

$$v_i = \frac{d}{dt}(r_i) \quad (i=2,3) \tag{8.11}$$

By substituting for $\phi_{ix}, \phi_{iy}, \phi_{iz}$ from Chapter 6, the velocities v_i can be expressed as a function of the first derivative of the elastic terms δ as follows

$$v = \psi\dot{\delta} + \eta \tag{8.12}$$

where ψ is a function of $\theta, \dot{\theta}$ and δ. η is a function of $\theta, \dot{\theta}$. From equation (8.11), δ can be obtained by least squares synthesis as follows

$$\dot{\delta} = (\psi^T\psi)^{-1}\psi^T(v-\eta) \tag{8.13}$$

The elastic terms δ can be obtained by integrating equation (8.13).

The following were considered in implementing the estimator for the system

- The inversion of the matrix $(\psi^T\psi)_{4\times4}$ was implemented symbolically to achieve fast execution of the code. However, the sampling rate was limited to 1 KHz.

- No noise model was assumed. However, the noise was attenuated by filtering accelerometer signals using 1-40 Hz band-pass filter and due to double integration.

• A linearized version of this estimator can be derived from chapter 6 as follows:

By attaching accelerometers to the tips of the elastic links along the $(XYZ)_{ei}$, the output acceleration a_i can be given by

$$a_i = {}^0T_i^T (\ddot{r}_i + g) \tag{8.14}$$

Using equations (6.18 to 6.21), and neglecting the coriolis and centrifugal terms, the relationship between the sensor output a_i and the deformation δ_i can be [99]

$$\begin{bmatrix} \delta_2 \\ \delta_3 \end{bmatrix} = - \begin{bmatrix} m_2 N_1^{-1} & m_3 N_1^{-1}{}^2 R_3 - m_3 N_1^{-1} N_2 N_3^{-1} \\ 0 & m_3 N_3^{-1} \end{bmatrix} \begin{bmatrix} a_2 \\ a_3 \end{bmatrix} \tag{8.15}$$

where

$$N_1 = K_{21} - K_{23} K_{22}^{-1} K_{23}^T \tag{8.16}$$

$$N_2 = K_{23} K_{22}^{-1} V_3 {}^0T_3 (K_{31} - K_{33} K_{32}^{-1} K_{33}^T) \tag{8.17}$$

$$N_3 = K_{31} - K_{33} K_{32}^{-1} K_{33}^T \tag{8.18}$$

The final result is

$$\begin{bmatrix} \delta_{2y} \\ \delta_{2z} \\ \delta_{3y} \\ \delta_{3z} \end{bmatrix} = \begin{bmatrix} -\dfrac{m_2 L_2^3}{3E_2 I_2} & 0 & -\dfrac{m_3 L_2^3}{3E_2 I_2}(2L_2\cos\theta_3 + 3L_3) & 0 \\ 0 & -\dfrac{m_2 L_2^3}{3E_2 I_2} & 0 & -\dfrac{m_3 L_2^3}{6E_2 I_2}(2L_2 + 3L_3\cos\theta_3) \\ 0 & 0 & -\dfrac{m_3 L_3^3}{3E_2 I_3} & 0 \\ 0 & 0 & 0 & -\dfrac{m_3 L_3^3}{3E_3 I_3} \end{bmatrix} \begin{bmatrix} a_{2y} \\ a_{2z} \\ a_{3y} \\ a_{3z} \end{bmatrix} \tag{8.19}$$

where a_{2y}, a_{2z} and a_{3y}, a_{3z} are the accelerations at the tip of the second and third links, respectively, pointed toward the reference frame $(XYZ)_0$. δ_{2y}, δ_{2z} and δ_{3y}, δ_{3z} are the deflections at the second and third links, respectively.

● This estimator will give acceptable results for the purpose of vibration control since the effect of the coriolis and centrifugal terms is small under low speed operations.

● The implementation of the estimator given by equation (8.19) does not require a lot of computations. Hence, it may be used for controllers that need a lot of computations such as the adaptive inverse dynamics algorithm.

8.5 Step Response

The system was simulated under a unit step command without taking into consideration the gravity and friction. The gains for the LQR were calculated by linearizing the system around ($\theta_2 = 0°$, $\theta_3 = 90°$). The joint angles response is as shown in Figures (8.2, 8.3, and 8.4), the links-tip deflections in Figures (8.5, 8.6).

It is shown that the system reaches the final position in less than 3 seconds. The response of the system is faster than the response under PD-control due to the improvement of the bandwidth. However, the deflection of the links tips is larger due to the significant increase of speed. The end-points, under this control strategy, settle down quickly after 2
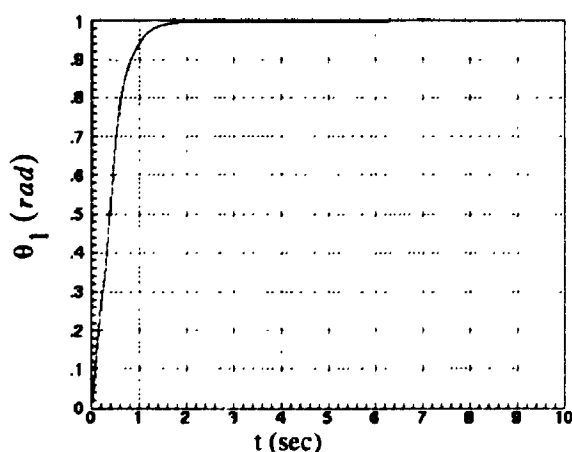
Figure (8.2): Response of the base joint under LQR control.
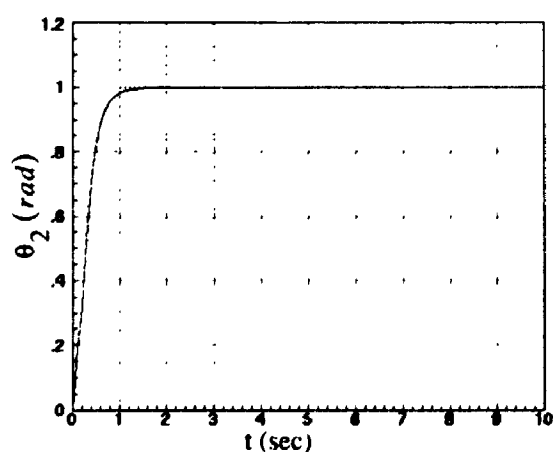
Figure (8.3): Response of the shoulder joint under LQR control.
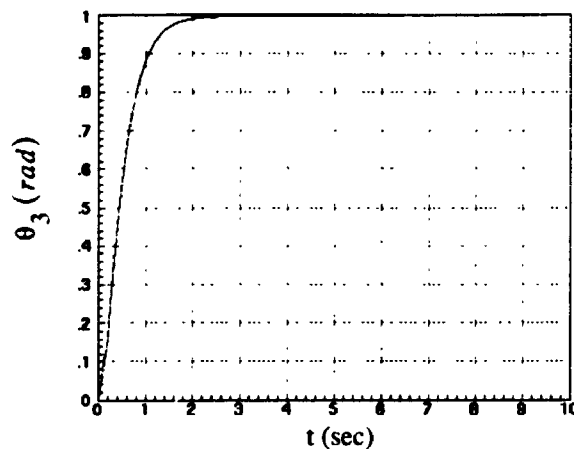
Figure (8.4): Response of the elbow joint
under LQR control.
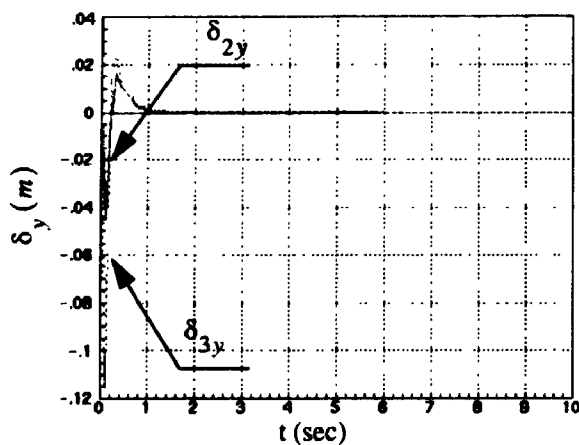
Figure (8.5): Links tip deflections of the vertical
direction under LQR control.
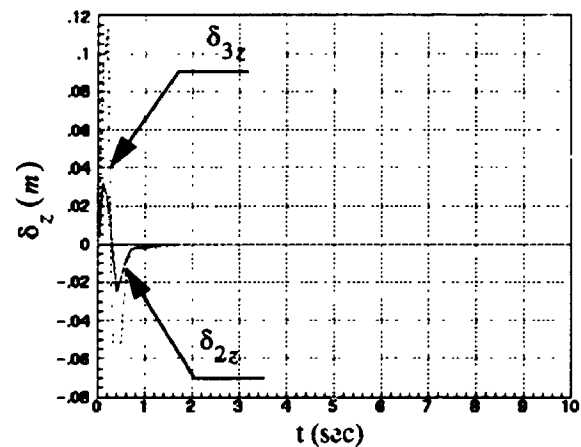
Figure (8.6): Links tip deflections in the horizontal
direction under LQR control.

second without vibration.

In order to show the effect of linearization angles used to calculate the LQR gains on the system performance, the response of the system based on a linearization configuration around ($\theta_2 = 70°$, $\theta_3 = -60°$) is as shown in Figures (8.7, 8.8, 8.9, and 8.10).

It is shown that the vertical performance remains acceptable. However, the horizontal motion suffers residual vibration. Consequently, adaptive control techniques have to be used to keep acceptable performance for all robot configurations. The first algorithm that was implemented on the robot was the gain-scheduling technique.
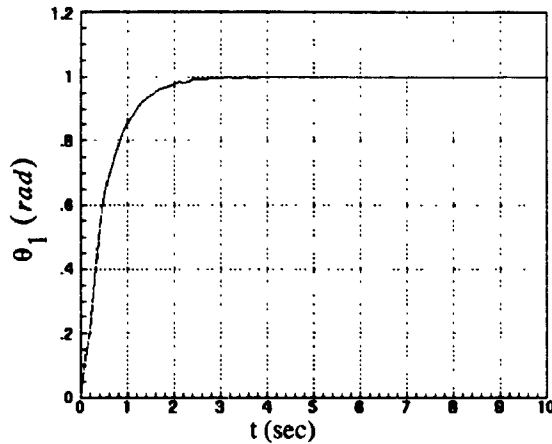
Figure (8.7): Response of the base joint under off-design LQR control.
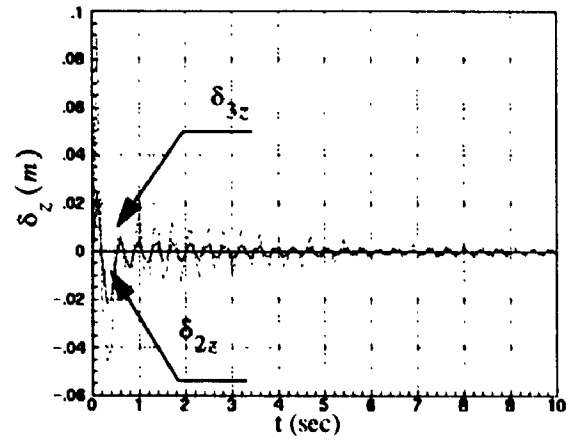
Figure (8.8): Links tip deflections in the horizontal direction under off-design LQR control.
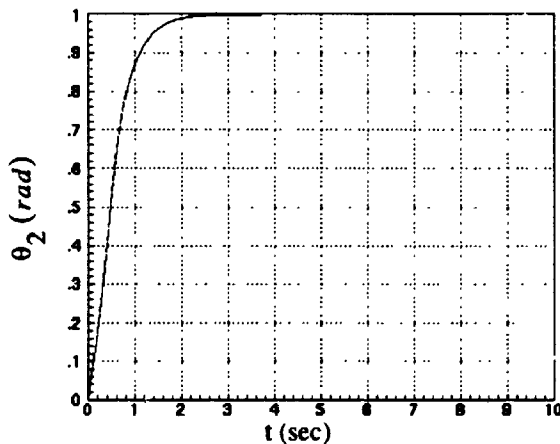
Figure (8.9): Response of the shoulder joint under off-design LQR control.
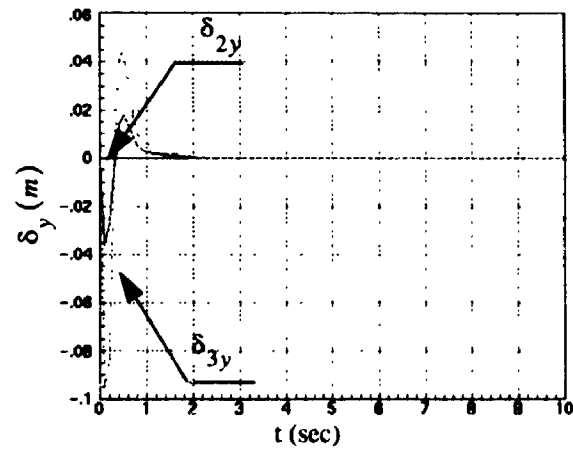
Figure (8.10): Links tip deflections of the vertical direction under off-design LQR control.

8.6 Simulation

The system is simulated under a step of one-second rise time. The response of the joint angles is shown in Figures (8.11,8.12, and 8.13) where the desired trajectory is represented by dashed lines and the actual response is represented by solid lines. The links tip deflections are as shown in Figures (8.14, 8.15). It is shown that the tracking performance is better than that which correspond to the PD controller and the tip vibration is less.
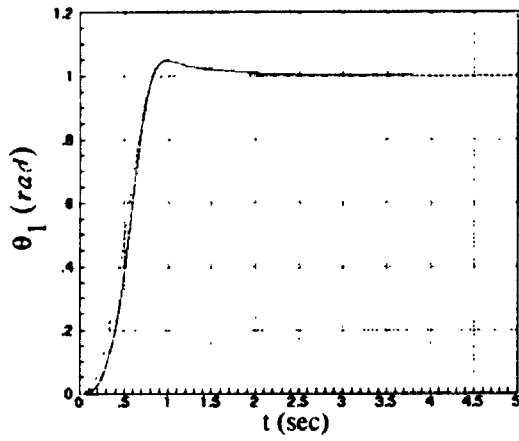
Figure (8.11): Response of the base joint
under LQR control.
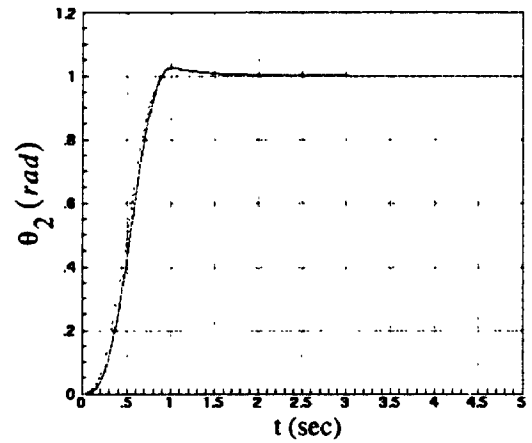
Figure (8.12): Response of the shoulder joint
under LQR control.
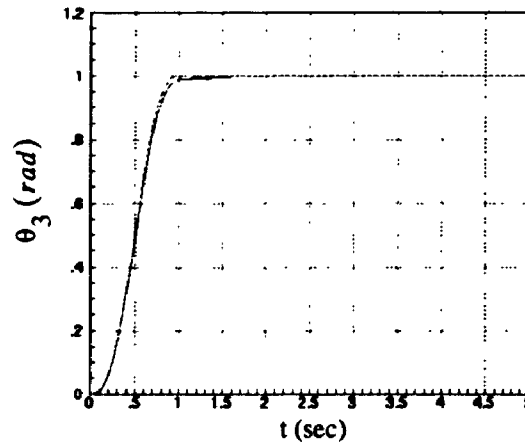
Figure (8.13): Response of the elbow joint
under LQR control.
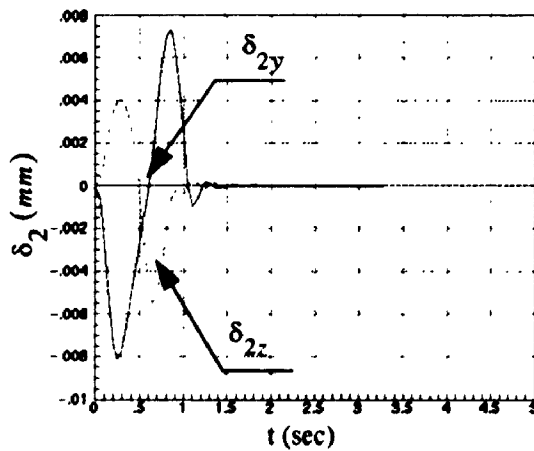
Figure (8.14): Second link tip deflections
under LQR control.
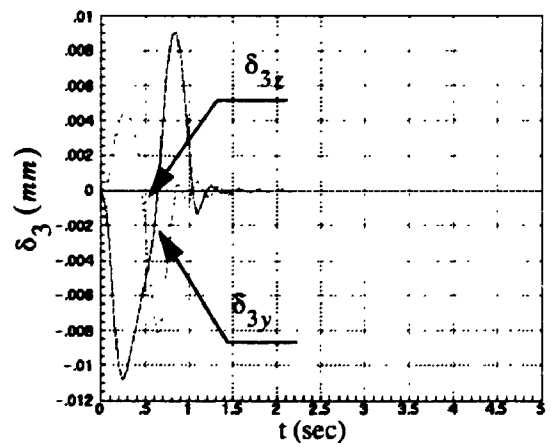
Figure (8.15): Third link tip deflections
under LQR control.

8.7 Gain Scheduling Control

The optimal feedback gains depend on the choice of the linearization point. Although the system may remain stable, the performance will be degraded if the design point is different from the operating point. It is known that the damping and bandwidth are adversely affected by a large change of the elbow and shoulder angles. Notice the change in the natural frequencies with the elbow and shoulder change in Section (5.1.4).

The performance can be improved using the gain-scheduling technique. The gains of the controller, based on the same cost function, are computed, discretized and stored off-line. As the robot sweeps across the workspace, the gains of the vertical motion change according to the elbow angle whereas the horizontal motion gains are a function of the elbow and shoulder angles.

As presented in [68], the main advantage of gain-scheduling is that linear design methods are applied to the linearized system at each operating point. Also, gain-scheduled systems tend to respond rapidly to changes in the operating conditions. The major difficulties are the selection of the scheduling procedure and the proof of stability.

In the flexible manipulator case, the gains were calculated as the angles are stepped by 0.1 radians. The gains were scheduled using a linear interpolation technique. Figure (8.16) shows the variation of some of the regulator gains with the changes of angles. Note that the values of the gains used for the plotting are the discretized values based on the algorithm in Appendix B. This technique will give a consistent performance at all the robot configurations.

It should be noted that the gains are functions of the angles only and not the angular velocities which may cause the response to deteriorate for high angular velocities. Also, the
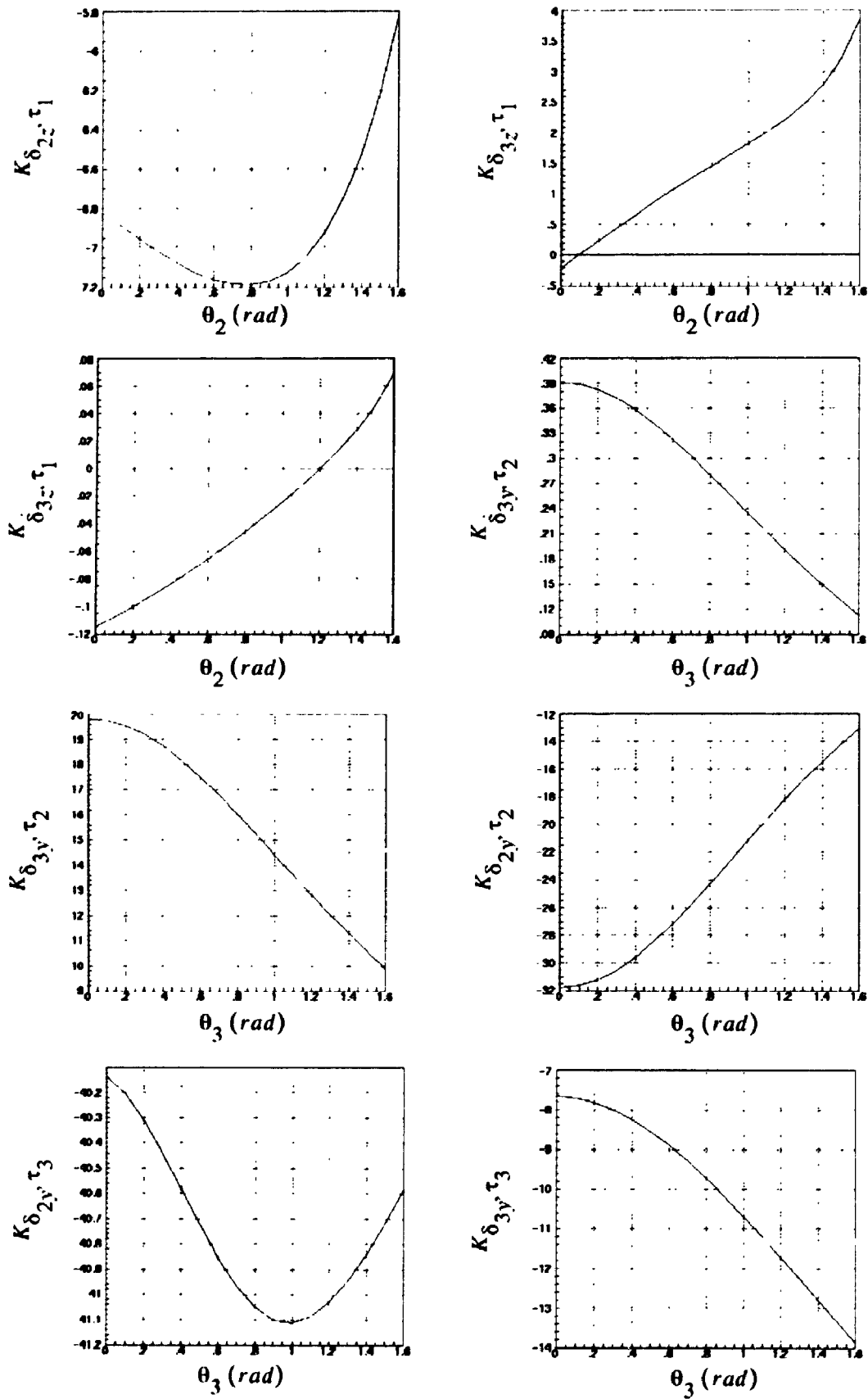
Figure (8.16): Variation of the LQR gains with the joint angles.

payload change is not taken into account which may even result in instability for high

payloads. In order to schedule the gains for the payload, a payload identification technique

is required [15]. However, experiments with payload identification were not successful due

to the presence of noise and unmodelled dynamics. Also, this technique usually leads to a

poor transient response for the system due to the time taken to identify the payload. This

led to the implementation of Lyapunov-based adaptive controllers.

8.8 Experimental Results

The gain-scheduling linear quadratic regulator (GS-LQR) controller, with gravity

and friction compensation was implemented at a sampling rate of 1 kHz, while the sensors

reading, averaging and filtering used 4 kHz to reduce the noise. The maneuver was set to

be 40° step in one second for all the joints which ends with an elbow-up posture.

The response for the robot joint under GS-LQR are shown in Figures (8.17,8.18,

and 8.19) where the desired trajectory is represented by dashed-lines and the actual

response is represented by solid lines. It is shown that the tracking is better than the PD
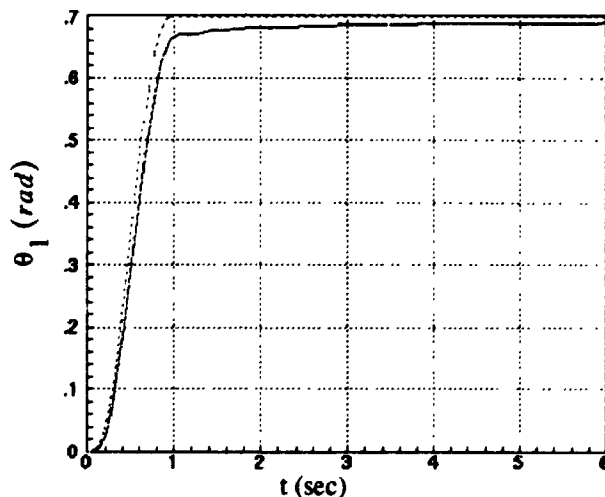
Figure (8.17): Experimental response of the base joint under GS-LQR control.
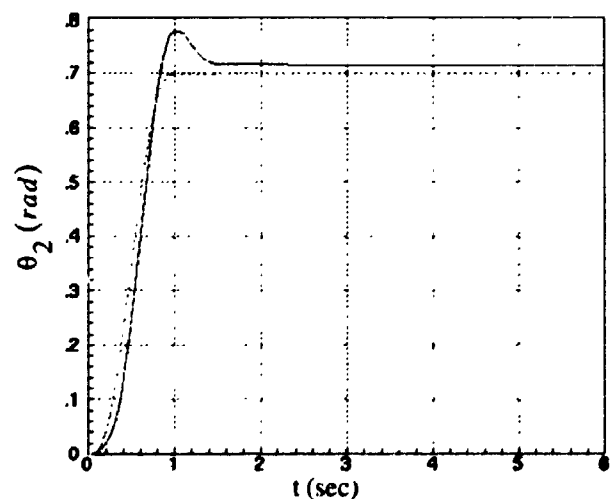
Figure (8.18): Experimental response of the shoulder joint under GS-LQR control.
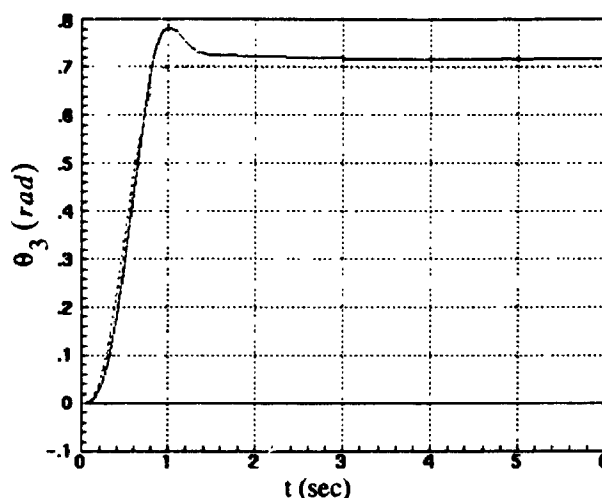
Figure (8.19): Experimental response of the elbow
joint under GS-LQR control.
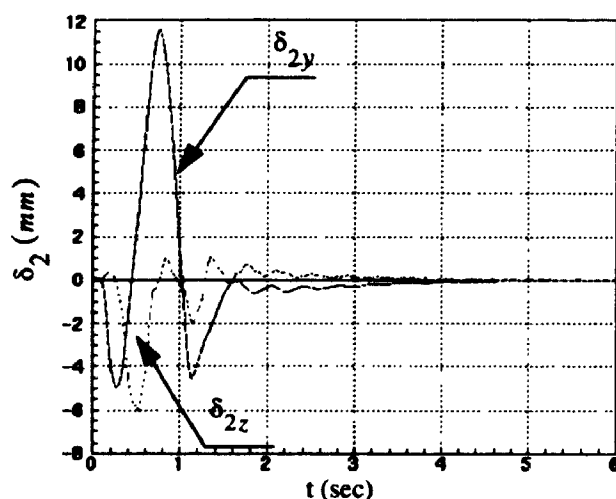
Figure (8.20): Second link tip deflections under GS-
LQR control.
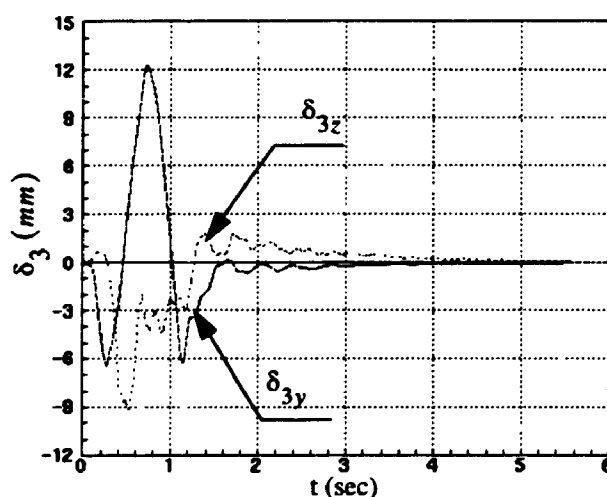
Figure (8.21): Third link tip deflections under GS-
LQR control.

although there exists a little overshoot at the elbow and shoulder joints.

The deflection at the tips of the links (in mm) are shown in Figures (8.20, 8.21). It is shown that the residual vibration (after 1 sec. maneuver) are quickly damped under the controller action. However, the end-point deflections take some time to reach zero due to the effect of friction.

In order to demonstrate the effect of friction, the same maneuver was run without friction compensation. The base response is shown in Figure (8.22). It is shown that the robot undergoes a large steady-state error and the tracking performance is poor. The
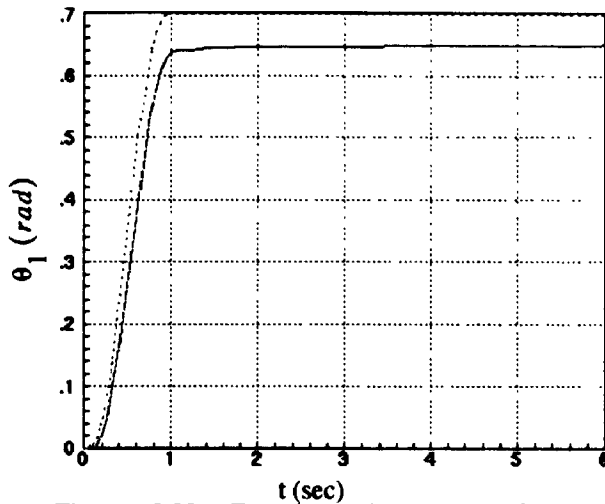
Figure (8.22): Experimental response of the base under GS-LQR without friction compensation.
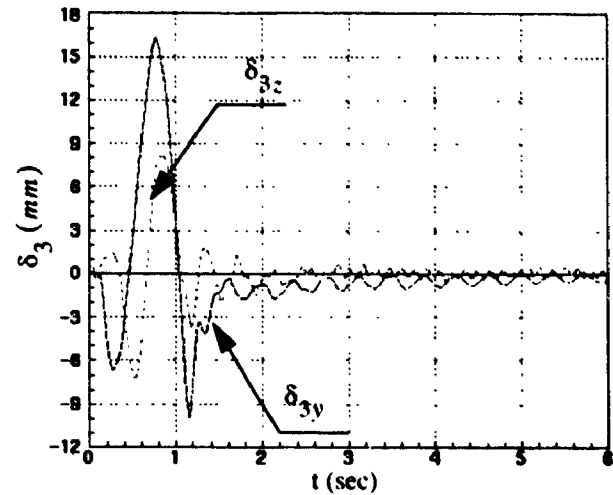
Figure (8.23): Third link tip deflections under GS-LQR control without friction compensation.

deflection of the tip of the third link is shown in Figure (8.23). It is shown that the robot undergoes excessive residual vibration. Since the torques required to damp the vibration are relatively small compared to the torques required for the rigid body motion, they can not overcome the joints coulomb friction which results in having residual vibration at the end of motion.

By increasing the payload by 0.5 kg, the response of the robot joints are shown in Figures (8.24, 8.25, and 8.26). It is shown that the robot joints are still able to track the desired trajectory even with an increase of the tip mass. This is attributed to the significant reduction of the effect payload dynamics on the motors due to the high ratio of the gear reducers implemented at the joints.

However, the tip of the links continue to vibrate and take longer to damp out. Since the tip mass is changed, the GS-LQR will operate off its design-point due to the change of the robot properties. Note that the GS-LQR controller is adaptive with respect to the change in the configuration and not to the change of payload.

This led to the design of more advanced controllers which will be presented in the following chapters.
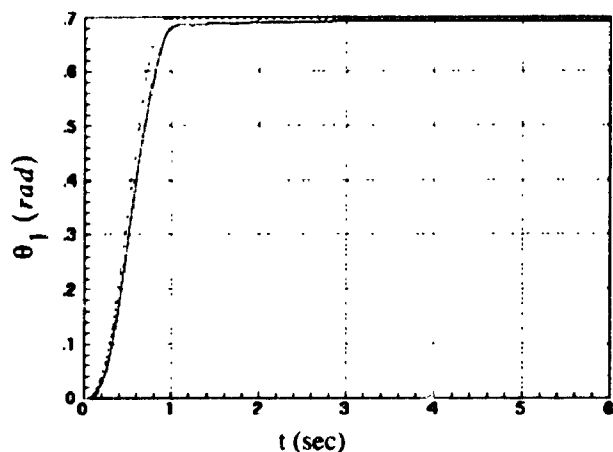
Figure (8.24): Response of the base with higher tip mass under GS-LQR control.
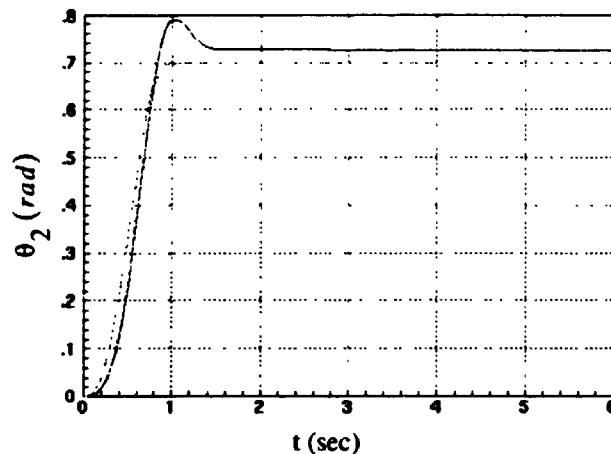
Figure (8.25): Response of the shoulder with higher tip mass under GS-LQR control.
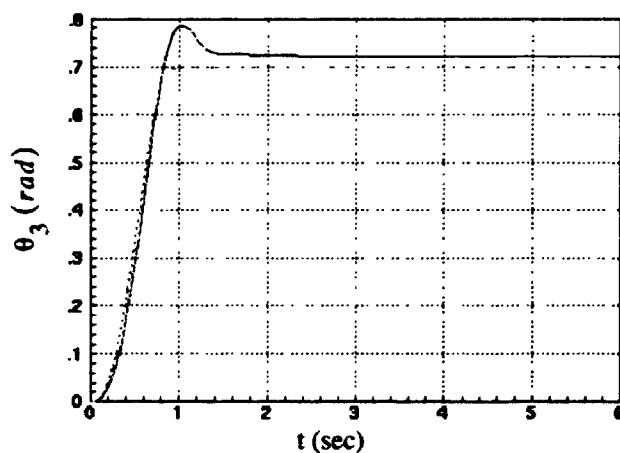
Figure (8.26): Response of the elbow with higher tip mass under GS-LQR control.
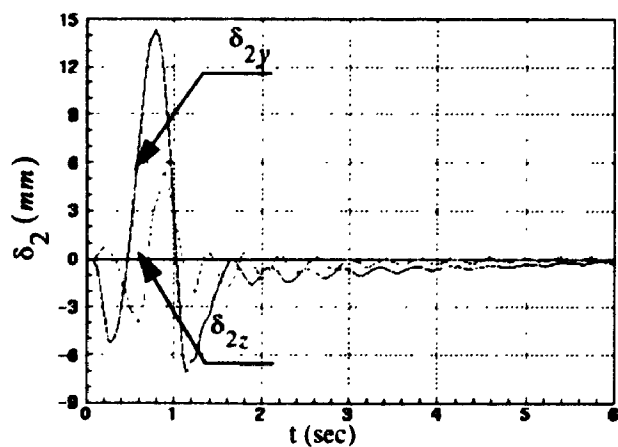
Figure (8.27): Second link tip deflections with higher tip mass under GS-LQR control.
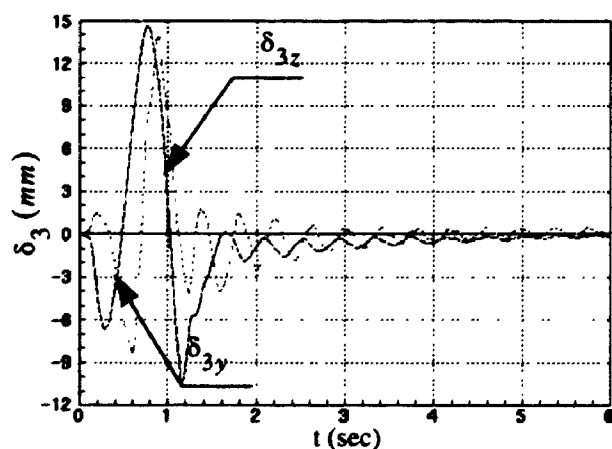
Figure (8.28): Third link tip deflections with higher tip mass under GS-LQR control.

8.9 Development of a Robust Nonlinear Observer

In this section, a robust observer is developed to be implemented for estimating the states of flexible manipulators. Throughout the thesis work, estimating the states was done using either equations (8.13) or (8.19). Although acceptable performance was obtained, the previous estimators can not be implemented for the general case of flexible manipulators. Also, obtaining the joint velocities from position measurements by differentiation may cause problems if the position signals are contaminated by noise.

The developed observer uses the sliding modes approach to improve the performance of a quasi-linear observer, as will be shown.

8.9.1 Development of a Quasi-Linear Observer

As explained in Section 8.4, the joint angles and the velocities of the links tip are measured. Hence, a linear observer can be designed based on the linearized system (equations 8.1,8.2) using optimal control theory. The equation for the full order observer can be given by [24]

$$\dot{\hat{x}} = A\hat{x} + Bu + K_e(y - \hat{y}) \tag{8.20}$$

where, y is the measurements vector given by

$$y = [\theta_1, \theta_2, \theta_3, v_{2y}, v_{2z}, v_{3y}, v_{3z}] \tag{8.21}$$

and K_e is the observer gain calculated by minimizing a cost function give by

$$J = \int_0^\infty (x^T Q_{xx} x + y^T Q_{yy} y)\, dt \tag{8.22}$$

The end-point velocities of the robot may be sufficient to estimate all the states. However, more measurements are used to improve the observer robustness.

The observer was designed by linearizing the equations of motion around ($\theta_2 = 0°$,

θ_3 — $90°$) and changing the weighting matrices Q_{xx}, Q_{yy} so that it would be twice as fast as the linearized system under regulator control. The weighting matrices are

$$Q_{xx} = diag\,(\,[\,10, 10, 10, 1, 1, 1, 1, 100, 100, 100, 10, 10, 10, 10\,]\,) \qquad (8.23)$$

$$Q_{yy} = 0.05 \times diag\,(\,[\,1, 1, 1, 10, 10, 10, 10\,]\,) \qquad (8.24)$$

In order to further improve the robustness in implementing the observer, the estimated output $\hat{y}$ was not calculated based on the linearized equations of motion (equation 8.2), rather the nonlinear functions relating the estimated links tip velocities to the estimated states were used by differentiating equations (6.9,6.10) symbolically.

The robot joints are commanded, in simulation, to track a trapezoidal trajectory, shown in Figure (8.29), under the high gains PD control presented in Section 7.4. The estimated states (represented by dotted-lines) were compared with the actual states (represented by solid-lines) as shown in Figures (8.30 to 8.34).

It is shown that the estimation for the joint velocities is poor although the estimation for the flexible states is acceptable.

8.9.2 Development of a Sliding Observer

Sliding observers are nonlinear state estimators based on the theory of variable structure systems [77]. Designing controllers based on the sliding modes approach feature good robustness for specific classes of nonlinear tracking problems. However, their applicability is limited since they involve large control authority and control chattering.

Sliding observers offer advantages similar to those of sliding controllers. Further, the chattering issues in sliding observers are only linked to numerical implementation and hence does not have influence on the system response.
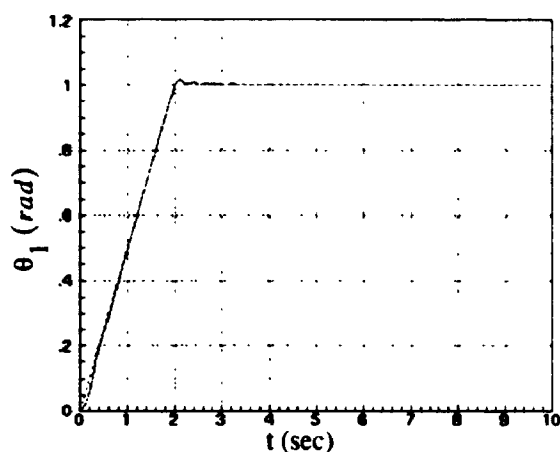
Figure (8.29): Response of the base under PD for a trapezoidal trajectory.
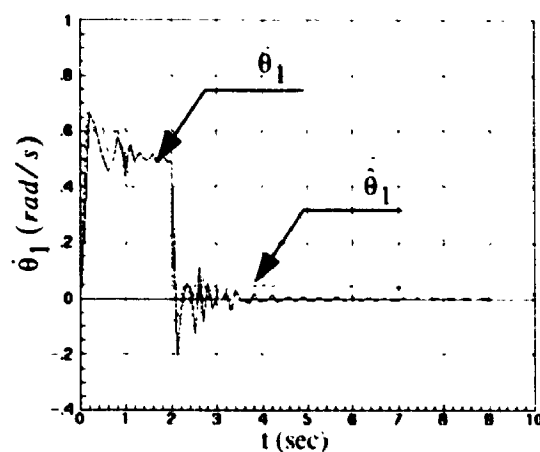
Figure (8.30): Estimated and actual base velocity using the quasi-linear observer.
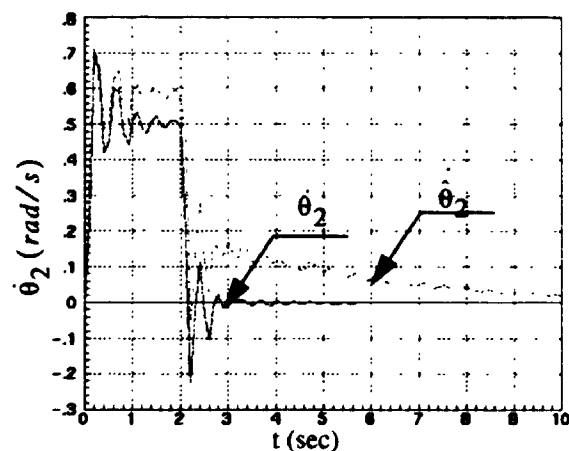
Figure (8.31): Estimated and actual shoulder velocity using the quasi-linear observer.
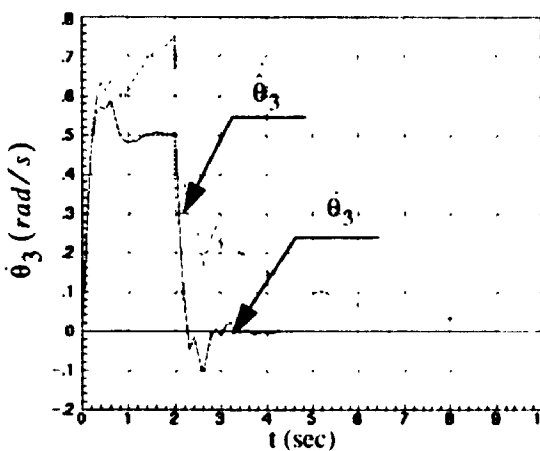
Figure (8.32): Estimated and actual elbow velocity using the quasi-linear observer.
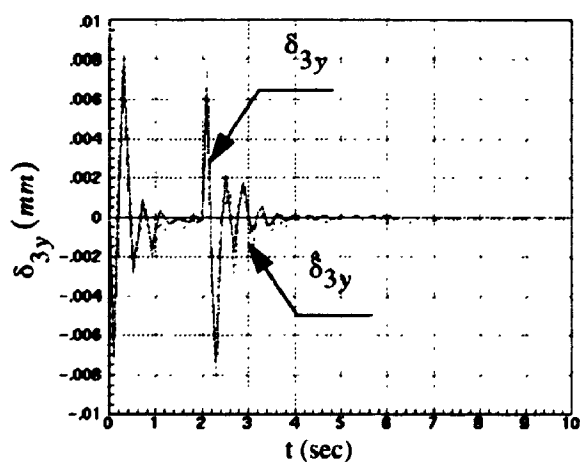
Figure (8.33): Estimated and actual third link vertical deflection using the quasi-linear observer.
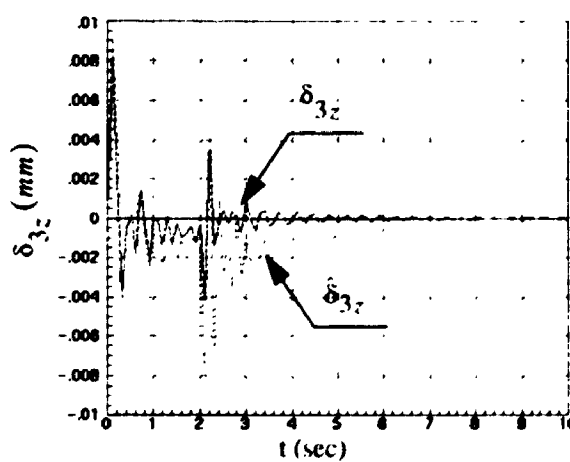
Figure (8.34): Estimated and actual third link horizontal deflection using the quais-linear observer.

The equation for the sliding observer is given by

$$\dot{x} = A\hat{x} + Bu + K_e(y - \hat{y}) + K_s I_s \qquad (8.25)$$

where K_s is the sliding observer gain, I_s is given by

$$I_s = \{sign(y - \hat{y})\} \qquad (8.26)$$

and $sign(y)$ is the signum function defined to be $sign(y) = y/|y|$.

Hence, the sliding observer is basically the linear observer (or Luenberger observer) with the additional "switching" term $K_s I_s$ that will be used to guarantee robustness against modelling errors/uncertainties.

Misawa and Hedrick [54] presented two methods to design the observer gain K_s for specific classes of nonlinear systems. The method used here is as follows: since the quasi-linear observer gave acceptable results in estimating the flexible states, the error vector used in I_s was taken to be the joint angles position error only, and K_s was taken to be

$$K_s = (10 - 100)B \qquad (8.27)$$

The system was simulated under the conditions present in Section 8.9.1. Figures (8.35 to 8.38) show the response of the observer where K_s was taken to oe $25B$. It is shown that the performance of the observer is better than the quasi-linear observer. Note that high values of K_s give better tracking performance for the joint velocities but may cause instability while estimating the flexible states and increase the observer sensitivity to noise.

The response of the observer while the robot payload is changed is shown in Figures (8.39 and 8.40). It is shown that the observer performance remains acceptable even with the change in parameters. This demonstrates the advantages of using such observer in estimating the states for flexible manipulators.
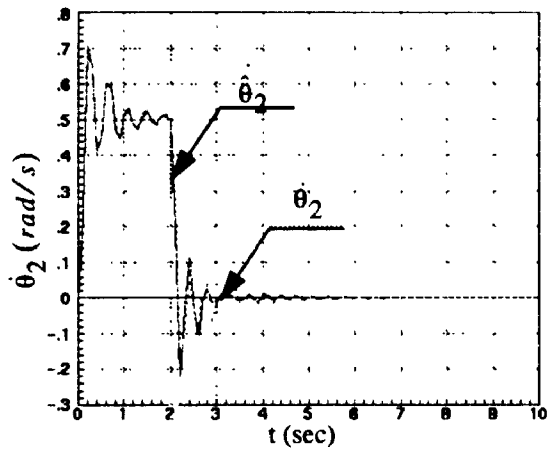
Figure (8.35): Estimated and actual shoulder velocity using the sliding observer.
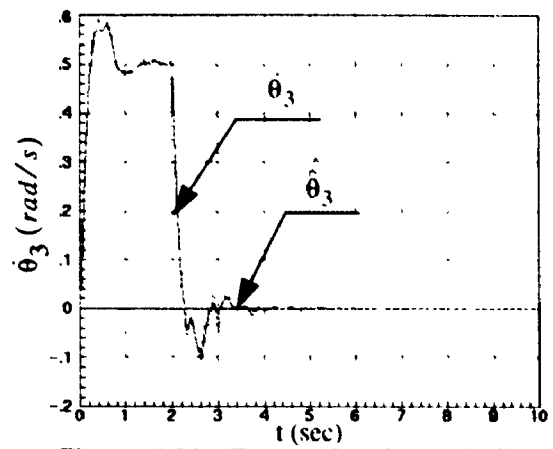
Figure (8.36): Estimated and actual elbow velocity using the sliding observer.
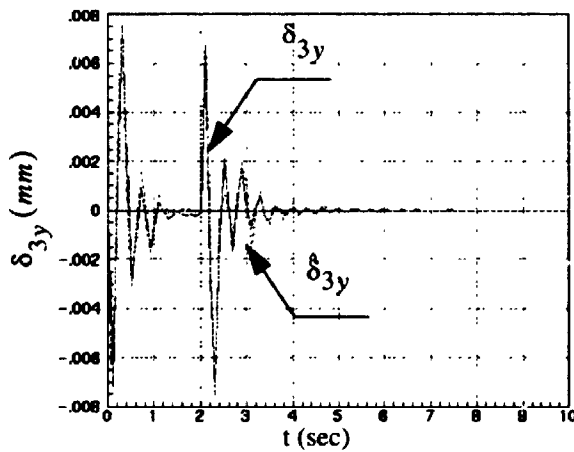
Figure (8.37): Estimated and actual third link vertical deflection using the sliding observer.
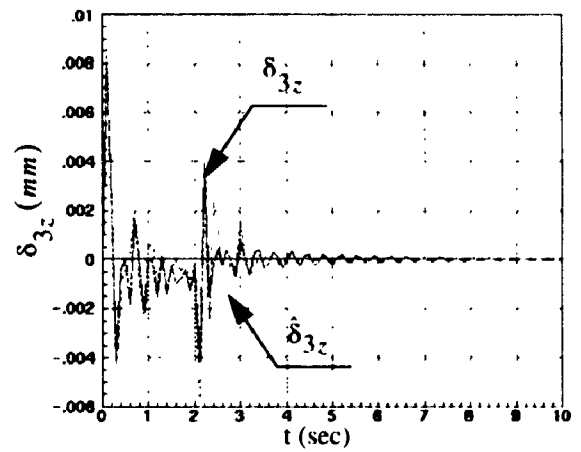
Figure (8.38): Estimated and actual third link horizontal deflection using the sliding observer.
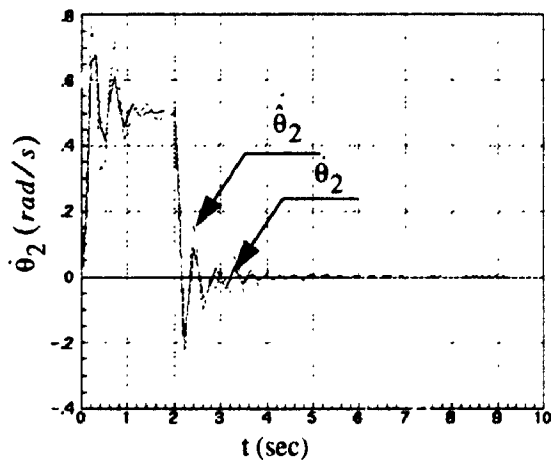
Figure (8.39): Estimated and actual shoulder velocity using the sliding observer for an increased payload.
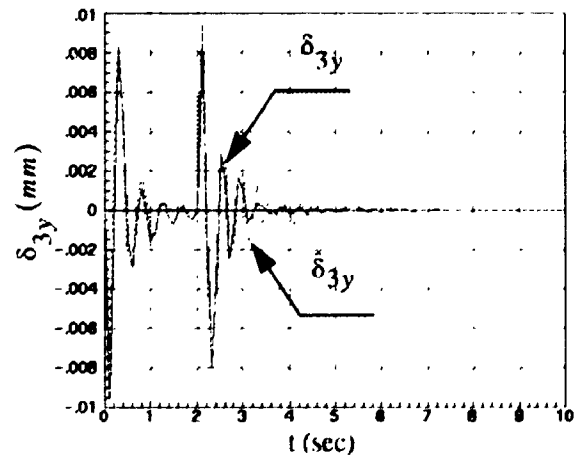
Figure (8.40): Estimated and actual third link vertical deflection using the sliding observer for an increased payload.

- CHAPTER 9 -

MODEL REFERENCE ADAPTIVE CONTROL

In this chapter, a general nonlinear controller is derived for manipulators with flexible links. The controller is based on the model reference approach. The adaptation technique was selected to be in the integral form and the gains were chosen to guarantee the stability of the system in the Lyapunov sense.

The reference model was derived from the linearized state equations with an optimal state feedback. The controller was enhanced further by choosing the reference model to be the full linearized system under optimal control.

9.1 Introduction

Model reference adaptive control (MRAC) is not new as a basic principle (see Landau [46]). Several authors applied this technique in the control of rigid robots. Dubowsky and DesForges [21] proposed an adaptive controller structure based on the gradient method. Stoten [82] used a model reference strategy to successfully control a two-link rigid manipulator. Stoten then extended this method to the Minimal Controller Synthesis (MCS) algorithm and proved the robustness of both methods [83].

For rigid robots, MRAC has some advantages over the computed torque method as the effect of modelling and parameter disturbance errors can be neglected and the amount of computation required is comparatively small.

Siciliano et al. [74] used MRAC to successfully control a single link flexible

manipulator. They showed that in order to satisfy the perfect model following conditions, the reference model was chosen from the linearized model of the system as optimally controlled. The nominal trajectory is commanded using a dynamic filter, which limits this method to the single link case due to the nonlinear interaction between the joint variables for the multi-link case.

9.2 The Controller Structure

The complete MRAC strategy is shown in Figure (9.1). The idea behind the control algorithm is to make the plant follow a perfect model (the reference model) which enables the synthesis of the LMRC (linear model reference controller) gains k_r, k and to generate ideal states x_m and output trajectories y_m following the reference trajectory. The adaptive loop changes the gains k_r and k by δk_r and δk as the state error x_e is detected.

As shown in Chapter 4, the robot equations of motion, in the state-space form, are
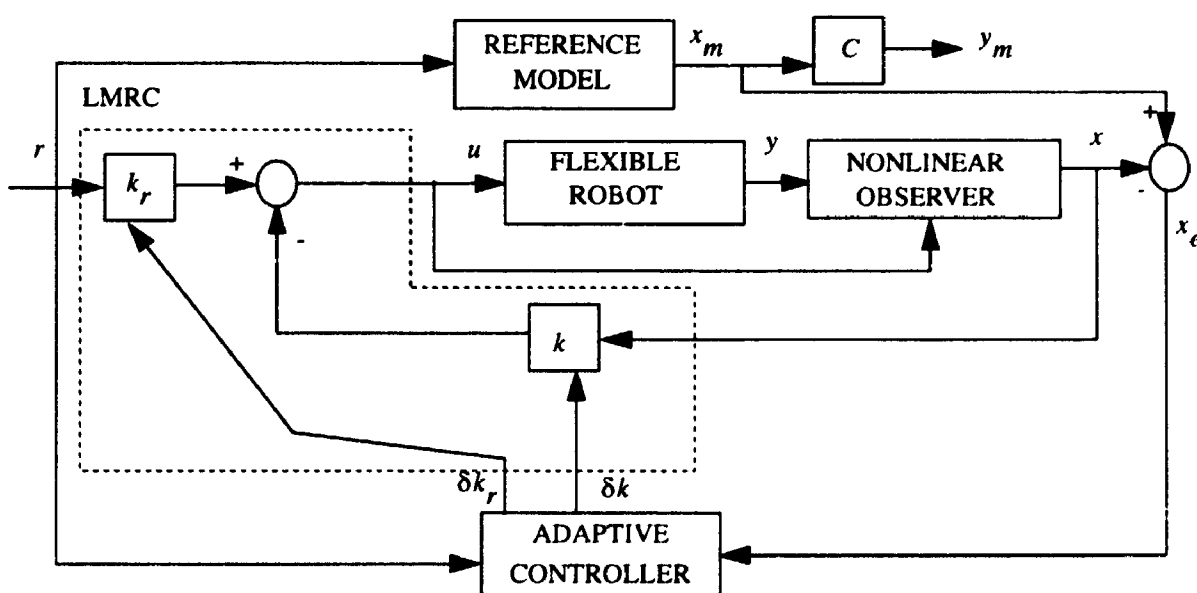
Figure (9.1): MRAC block diagram.

given by [82]

$$\dot{x} = Ax + Bu \tag{9.1}$$

$$y = Cx \tag{9.2}$$

Let the reference model be

$$\dot{x}_m = A_m x_m + B_m r \tag{9.3}$$

where

$$A_m = \begin{bmatrix} 0 & I \\ A_1 & A_2 \end{bmatrix}, \ B_m = \begin{bmatrix} 0 \\ B_0 \end{bmatrix} \tag{9.4}$$

Let the standard linear state feedback tracking controller be

$$u = -kx + k_r r \tag{9.5}$$

and defining the model following error to be

$$x_e = x_m - x \tag{9.6}$$

By substraction of equations (9.3) from (9.1) and by substituting for u by equation (9.5). This leads to

$$\dot{x}_e = A_m x_e + (A_m - A + Bk)x + (B_m - Bk_r)r \tag{9.7}$$

Consequently, the error dynamics reduces to

$$\dot{x}_e = A_m x_e \tag{9.8}$$

if

$$A_m - A + Bk = 0 \tag{9.9}$$

$$B_m - Bk_r = 0 \tag{9.10}$$

This implies that the error tends to zero ($x_e \rightarrow 0_{n,1}$) as t increases, whatever the initial finite error $x_e(\upsilon)$ may be. Hence, the desired objective of $x \rightarrow x_m$ is achieved. This

implies that $y \rightarrow y_m$, assuming that C matrix is the same for both the plant and the reference model. Performing the pseudo inverse of B in equations (9.9) and (9.10) gives

$$k = B^\dagger (A - A_m) \tag{9.11}$$

$$k_r = B^\dagger B_m \tag{9.12}$$

where, $B^\dagger = (B^T B)^{-1} B^T$ is the pseudo inverse for B.

By back substitution to equation (9.9) and equation (9.10). Then,

$$(I - BB^\dagger) (A_m - A) = 0 \tag{9.13}$$

$$(I - BB^\dagger) B_m = 0 \tag{9.14}$$

which are called the *Ezberger*'s conditions [46,82]. These conditions are used to verify the choice of the reference model although small terms on the right-hand side (instead of the zeros) can be accepted without too much loss of closed-loop performance.

9.3 The Reference Model

The gain matrix k is chosen to be the solution for the minimization of a performance index problem for the linearized system. Then

$$A_m = A_l - B_l k \tag{9.15}$$

where, A_l and B_l are the linearized system matrices. This choice satisfies the *Ezberger*'s conditions. B_m is chosen to give a "perfect model following" condition. The manipulator joint angles must reach certain values equal to the reference trajectory in a given time as well as equal to the reference model. This implies that

$$\bar{y}_m = \bar{y} = \bar{r} \tag{9.16}$$

where, r is the reference trajectory. The "bar" notation implies the steady state values.

This requirement is satisfied when $\dot{x}_m$ is zero, which gives

$$\bar{x} = -A_m^{-1} B_m \bar{r} \tag{9.17}$$

where, the inverse always exists. Consequently,

$$\bar{y}_m = -CA_m^{-1} B_m \bar{r} \tag{9.18}$$

From equation (9.17), this simply leads to

$$B_m = -A_m C^\dagger \tag{9.19}$$

9.4 The MRAC Adaptive Loop

Equation (9.5) can be extended to include the adaptive terms as follows

$$u = -(k+\delta k)x + (k_r + \delta k_r)r \tag{9.20}$$

Substituting with the control action into equation (9.1), and by subtraction of equation (9.3) from the result. This leads to

$$\dot{x}_e = A_m x_e + (A_m - A + B(k+\delta k))x + (B_m - B(k_r + \delta k_r))r \tag{9.21}$$

Let

$$A_c = A - B(k+\delta k) \tag{9.22}$$

$$B_c = B(k_r + \delta k_r) \tag{9.23}$$

Then, equation (9.21) becomes

$$\dot{x}_e = A_m x_e + (A_m - A_c)x + (B_m - B_c)r \tag{9.24}$$

In order to satisfy the model following conditions and using equations (9.9) and (9.10), then

$$A_m - A_c = B\delta k \tag{9.25}$$

$$B_m - B_c = -B\delta k_r \tag{9.26}$$

which means that the gains will vary according to the difference between the reference model and the plant. Substituting back to equation (9.7)

$$\dot{x}_e = A_m x_e + B\delta kx - B\delta k_r r \tag{9.27}$$

In order to guarantee the stability of the overall system, defining the Lyapunov function candidate V to be [74,46]

$$V = x_e^T P x_e + tr((A_m - A_c)^T \Phi (A_m - A_c)) + tr((B_m - B_c)^T \Psi (B_m - B_c)) \tag{9.28}$$

where, P, Φ, Ψ are positive definite matrices. Taking the derivative of V results in

$$\dot{V} = \dot{x}_e^T P x_e + x_e^T P \dot{x}_e + 2tr((A_m - A_c)^T(-\Phi \dot{A}_c)) + 2tr((B_m - B_c)^T(-\Psi \dot{B}_c)) \tag{9.29}$$

substituting from equations (9.7), (9.25) and (9.26). Then,

$$\dot{V} = x_e^T(A_m^T P + P A_m)x_e + 2tr((B\delta k)^T(P x_e x^T - \Phi \dot{A}_c)) + 2tr((-B\delta k_r)^T(P x_e r^T - \Psi \dot{B}_c)) \tag{9.30}$$

since,

$$A_m^T P + P A_m = -Q \tag{9.31}$$

where Q is a positive definite matrix, and equation (9.31) is the Lyapunov equation form [46,71].

From equations (9.25) and (9.26), it can be shown that

$$\dot{A}_c = -B\delta \dot{k} - \dot{B}\delta k \tag{9.32}$$

$$\dot{B}_c = B\delta \dot{k}_r + \dot{B}\delta k_r \tag{9.33}$$

since A_m, B_m are constant matrices. Assuming that the gains change are much faster than the system changes, that is

$$\delta k, \delta k_r \gg \dot{B}_c, \dot{A}_c \tag{9.34}$$

then,

$$\dot{V} = -x_e^T Q x_e + 2tr\left((B\delta k)^T (Px_e x^T + \Phi B\delta k)\right) + 2tr\left((-B\delta k_r)^T (Px_e r^T - \Psi B\delta k_r)\right) \tag{9.35}$$

In order to guarantee that $\dot{V}$ is negative (i.e. stability in the Lyapunov sense), the terms $Px_e x^T + \Phi B\delta k, Px_e r^T - \Psi B\delta k_r$ should vanish. This is achieved by multiplying both equations by B^T and solving for $\delta k, \delta k_r$. Then

$$\delta k = -(B^T \Phi B)^{-1} B^T Px_e x^T \tag{9.36}$$

$$\delta k_r = (B^T \Psi B)^{-1} B^T Px_e r^T \tag{9.37}$$

These equations can be integrated given the initial conditions to give $\delta k, \delta k_r$. Assuming that the terms $-(B^T \Phi B)^{-1}$ and $(B^T \Psi B)^{-1}$ can be considered as constant terms. By setting $C_e = B^T P$, then

$$y_e = B^T Px_e = C_e x_e \tag{9.38}$$

Then, the solution for $\delta k, \delta k_r$ can be put in an integral form

$$\delta k = \int_0^t \alpha_1 y_e x^T d\tau \tag{9.39}$$

$$\delta k_r = \int_0^t \alpha_2 y_e r^T d\tau \tag{9.40}$$

where, $\alpha_1, \alpha_2 > 0$ should be large enough to overcome the changes in B while keeping $\dot{V}$ negative and hence assure stability in the Lyapunov sense [77].

This formula is similar to the form derived by Landau [46] and Stoten [82]. However, in their formulae they added low-value constant terms β's to the integration in order to improve the convergence of the gains. The final formulae for the adaptation gains are

$$\delta k = \int_0^t \alpha_1 y_e x^T d\tau + \beta_1 y_e x^T \tag{9.41}$$

$$\delta k_r = \int_0^t \alpha_2 y_e r^T d\tau + \beta_2 y_e r^T \qquad (9.42)$$

In general, the ratio of α to β is taken to be around 10:1, and α_2, β_2 are taken to be equal to α_1, β_1 for simplicity.

9.5 Steps for Designing MRAC

The steps for designing a model reference adaptive controller are summarized in the following points:

1. An optimal regulator is designed based on the linearized form of the state-space equations (equation 9.1) where the weighting functions can be selected to give a maximum real part for the poles of the linearized system using MATRIX$_X$ [37].

2. The reference model is designed using equations (9.15) and (9.19).

3. The k_r matrix is estimated using equation (9.12).

4. The Q matrix is chosen to be a diagonal matrix with positive elements. Then, the P matrix is calculated by solving Lyapunov equation (9.31) using the routine shown in Appendix B.

5. The C_e matrix is estimated and the y_e vector is calculated based on equation (9.38).

6. The adaptation gains are calculated based on equations (9.41) and (9.42).

9.6 Simulation of MRAC

The joint angle trajectory is chosen to be a step command with a trajectory shown

in Figure (9.2) and an amplitude of 1 radian for all the joints. This particular trajectory has infinite acceleration at the start and end which results in large tip deflection and residual vibration. The manipulator is initially at rest at the home position. The optimal regulator was designed similar to that given in Chapter 8 and the system was linearized around the home position for the calculations of the LQR gains. The payload was taken to be 0.5 kg.

Two different sets of simulation were carried out, one with constant state feedback and the other with the adaptation algorithm where α and β were chosen arbitrary to 20.0 and 1.0, respectively. Q was chosen to be (refer to Section 6.4 for the corresponding state)

$$Q = diag\,(10, 10, 10, 1, 1, 1, 1, 100, 100, 100, 10, 10, 10, 10) \qquad (9.43)$$

The response for the joint angles is as shown in Figures (9.3, 9.4, and 9.5) and the tip vibration is as shown in Figures (9.6, 9.7). It is shown that the adaptive controller behaves better than the constant feedback controller as the latter operates off the design point with respect to the configuration change.

The tip mass is changed to 1.55 kg. The joint angles response, for both controllers, is shown in Figures (9.8, 9.9, and 9.10). The tip vibration is shown in Figures (9.11, 9.12). It is shown that since the constant feedback controller is employed off the design point, with respect to the configuration and parameters changes, the vibration is large, although the system remains stable. However, the adaptive controller retains its response and behaves as expected, but the system is a little slower than the previous case. The slow motion of the system under MRAC is attributed to the choice of the reference model. In conclusion, this controller can not be used for trajectory tracking.
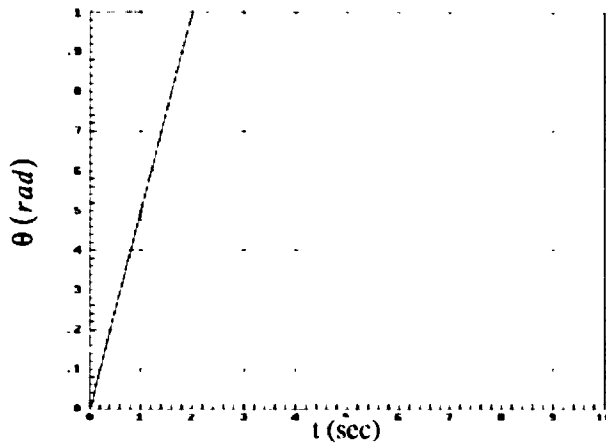
Figure (9.2): Desired trajectory for all the angles.

Figure (9.3): Response of the base joint (0.5 kg tip mass).

Figure (9.4): Response of the shoulder joint (0.5 kg tip mass).

Figure (9.5): Response of the elbow joint (0.5 kg tip mass).

Figure (9.6): Robot end-point deflection in the vertical direction (0.5 kg tip mass).
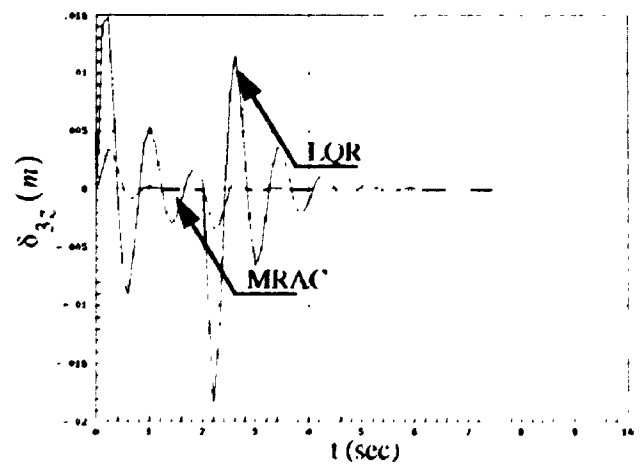
Figure (9.7): Robot end-point deflection in the horizontal direction (0.5 kg tip mass).
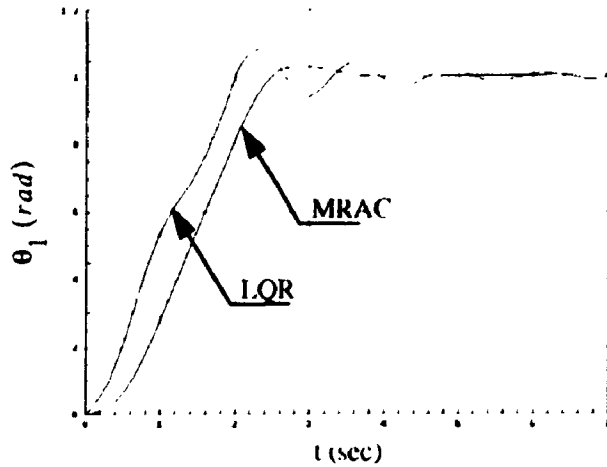
Figure (9.8): Response of the base joint (1.55 kg tip mass).
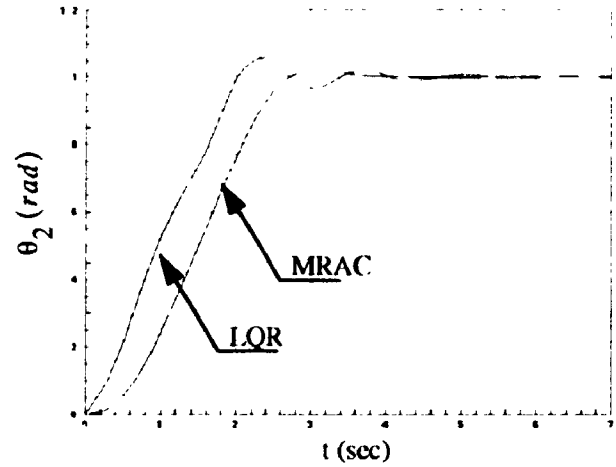
Figure (9.9): Response of the shoulder joint (1.55 kg tip mass).
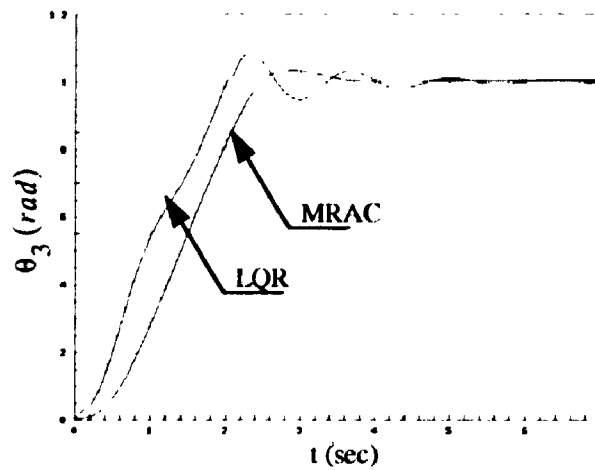
Figure (9.10): Response of the elbow joint (1.55 kg tip mass).
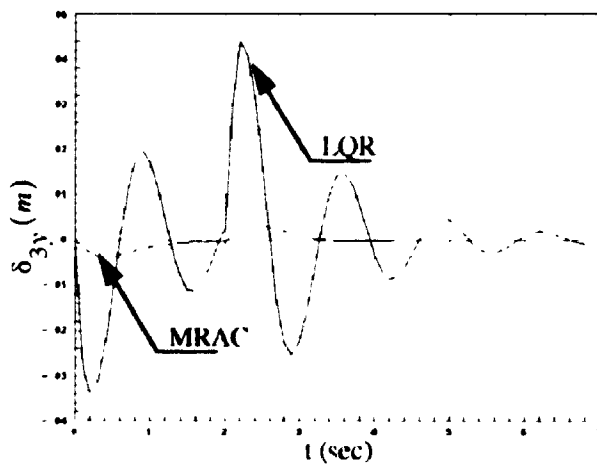
Figure (9.11): Robot end-point deflection in the vertical direction (1.55 kg tip mass).
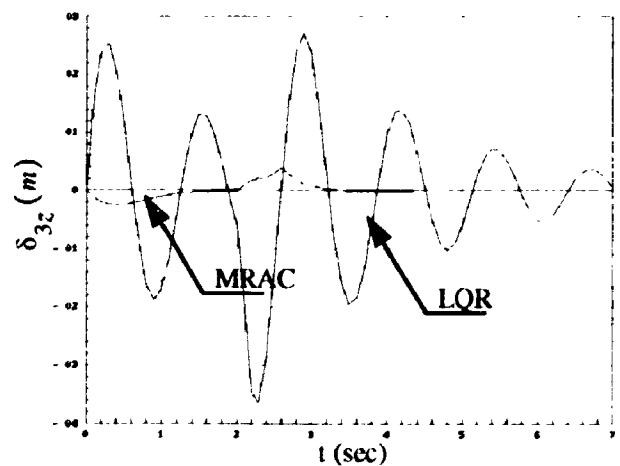
Figure (9.12): Robot end-point deflection in the horizontal direction (1.55 kg tip mass).

9.7 Experimental Results

The model reference adaptive control law is applied to the system at a sampling rate of 1 kHz. Note that the control law was implemented with friction and gravity compensation. The gravity torques were calculated for the nominal payload of 1.55 kg. Also, for this control law, the reference inputs are the desired angles only and it does not require the velocity as input.

The values of α and β were set to be 5.0, 0.5. Note that they are adjusted by trial and error to achieve better tracking and vibration damping for the robot. It was noticed that high values of α and β excite the system unmodelled dynamics, and especially the torsional modes of the elbow joint present at 26.5 Hz. The system tends also to vibrate excessively, with high values of the adaptation gains, in the horizontal direction at some positions (e.g. the straight horizontal configuration).

The system response to a one-second step is shown in Figures (9.13 to 9.17), where the reference input is represented by dashed-lines and the respose by solid lines. It is shown that the robot joints do not track the reference input. This is attributed to the choice of the reference model as it has relatively low bandwidth.

The low values of the end-points deflections are due to the slow movement of the robot. However, the residual vibrations are damped quickly by the control action.

By attaching 0.5 kg to the robot tip, the response the system to the one-second time step is shown in Figures (9.18 to 9.22). It is shown that the robot behavior is the same as its behavior with the regular load. The end-point deflections changed slightly. However, they are close to the values of the previous experiment.
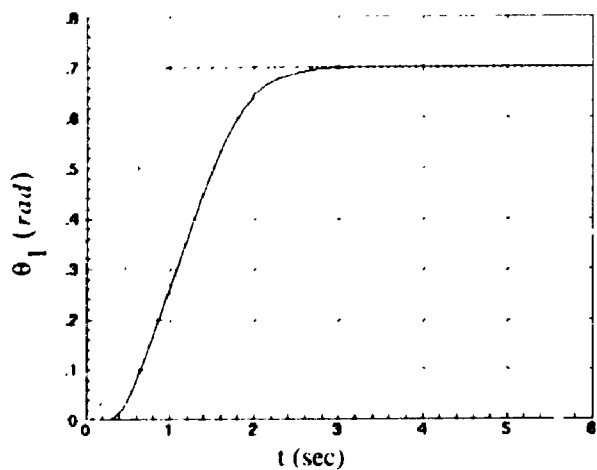
Figure (9.13): Experimental response of the base under MRAC.
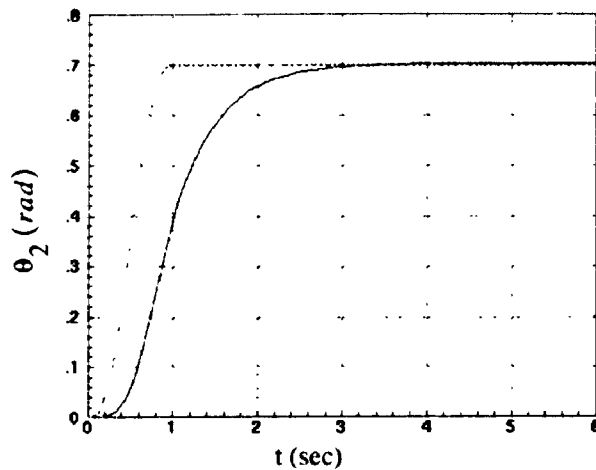
Figure (9.14): Experimental response of the shoulder under MRAC.
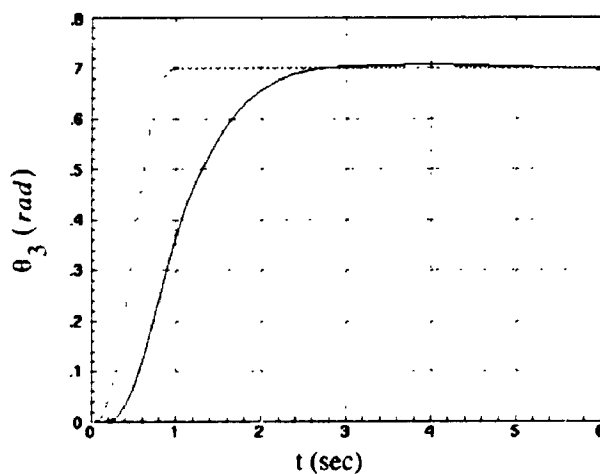
Figure (9.15): Experimental response of the elbow under MRAC.
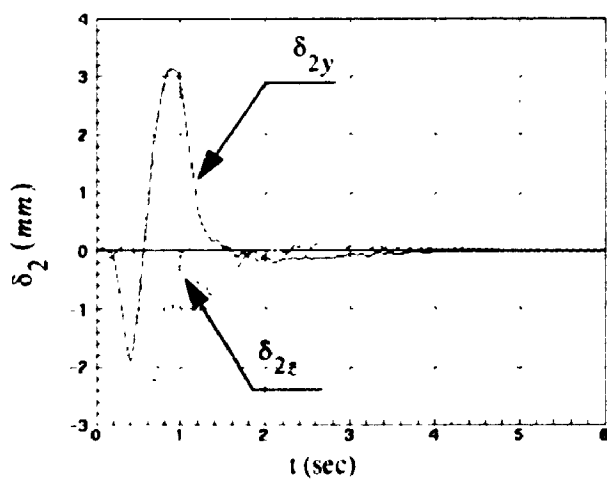
Figure (9.16): Second link tip deflections under MRAC.
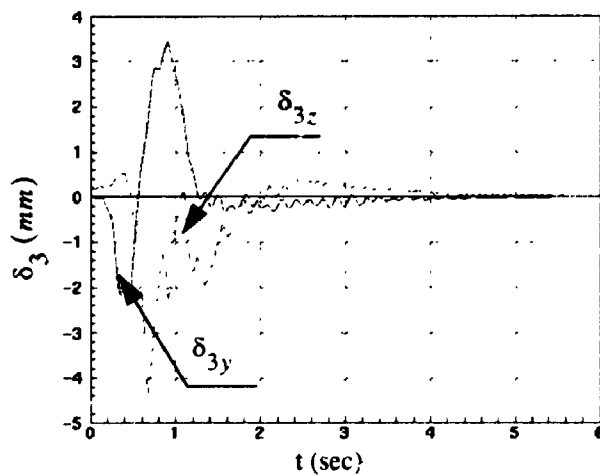
Figure (9.17): Third link tip deflections under MRAC.
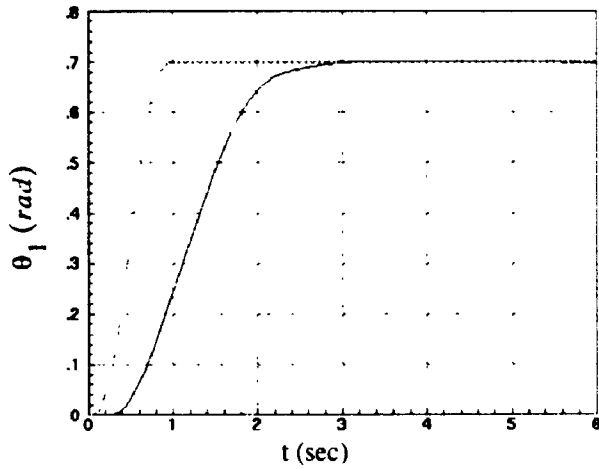
Figure (9.18): Response of the base with higher
tip mass under MRAC.
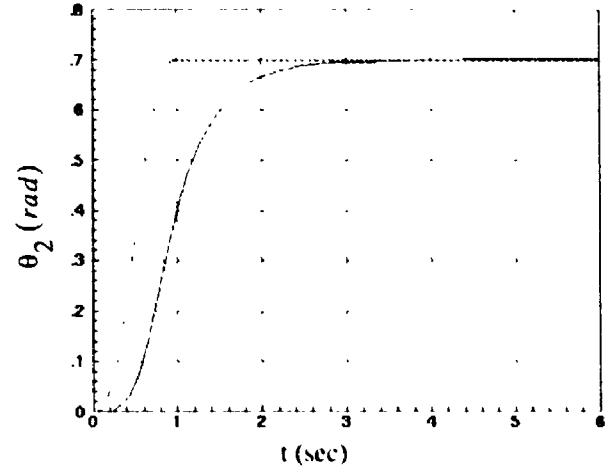
Figure (9.19): Response of the shoulder with higher
tip mass under MRAC.
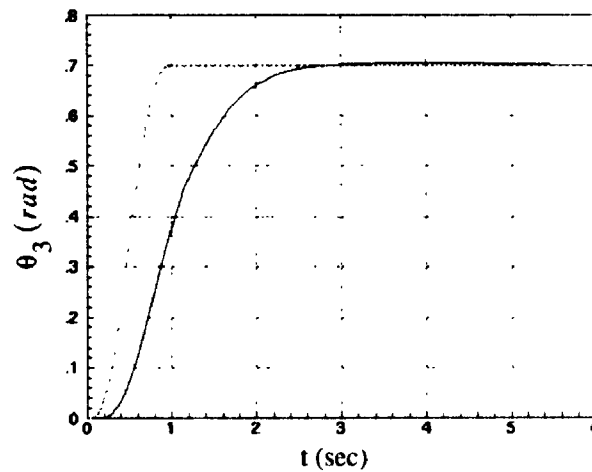
Figure (9.20): Response of the elbow with higher
tip mass under MRAC.
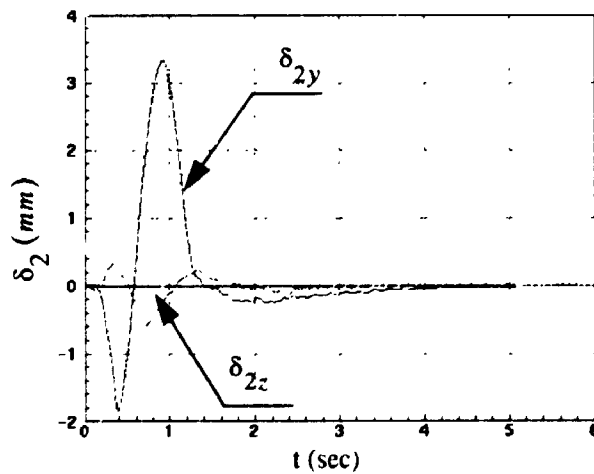
Figure (9.21): Second link tip deflections with
higher tip mass under MRAC.
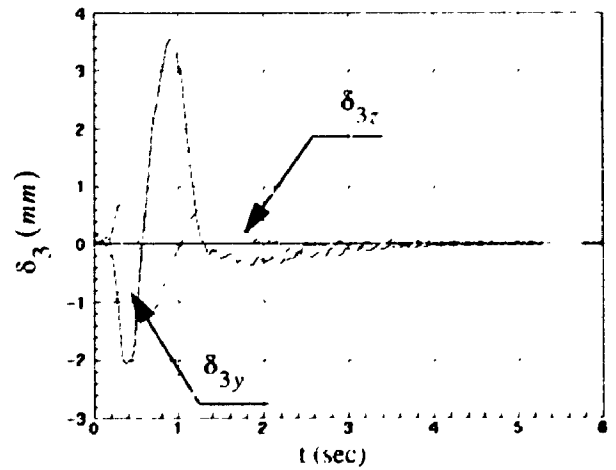
Figure (9.22): Third link tip deflections with
higher tip mass under MRAC.

9.8 Enhancement of the Reference Model

It was shown in the previous simulation that the response of the system is slower than the LQR and it does not track the trajectory perfectly. This is attributed to the choice of the reference model. The performance can be improved by choosing a faster reference model that satisfies the *Ezberger*'s conditions.

Given the linearized system matrices as shown in Figure (9.23), where A_l, B_l are the linearized state matrices, the reference model can be taken to be the linearized system as optimally controlled.
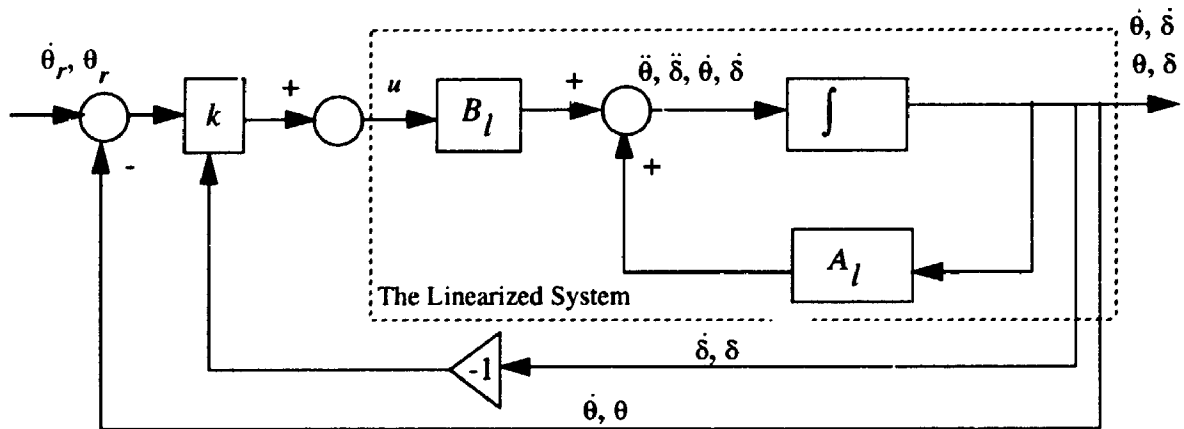
Figure (9.23): The Optimally-Controlled Linearized System.

Hence, A_m, B_m can be given according to Figure (9.23) to be

$$A_m = A_l - B_l k \qquad (9.44)$$

$$B_m = B_l k_r \qquad (9.45)$$

$$k_r = kC^\dagger \qquad (9.46)$$

where C is a 6x14 matrix that corresponds the joint angles and velocities as outputs from the state variable (equation (9.22)), and k is chosen similar to Section 9.3. Also, It can be shown that this choice satisfies the *Ezberger*'s conditions.

9.9 Simulation Results

Similar to section 9.7, the reference model was taken to be the linearized system under optimal state feedback. Using MATRIX$_X$, the reference model was synthesized using equations (9.44 to 9.46).

Simulation was carried out on two cases, with 0.5 kg payload and with 1.55 kg payload. α and β were chosen arbitrary to be 0.1 and 0.01, respectively. Q was chosen to be

$$Q = diag\,(0.2, 1, 1, 0.1, 0.01, 0.1, 0.01, 1, 5, 5, 1, 0.1, 1, 0.5) \tag{9.47}$$

Figures (9.24 to 9.28) shows the response of the system under the adaptive controller. It is shown that the response of the system is as fast as the system under LQR control. However, under MRAC the system has little overshoot and the end-point deflection is less. This is attributed to the model following characteristics of the system as it tends to follow the reference model.

The response of the system with higher, and unexpected payload is shown in Figures (9.29 to 9.33). It is shown that the system under the adaptive controller preserves its characteristics and the response is almost the same as that with the increased payload.

It was also found out that very high values of α and β, which implies faster adaptation, produce instability to the system. Hence, there is always a trade-off between instability and faster adaptation. The values of α can be calculated based on equation (9.36), while the values of β are usually taken to be one-tenth the values of α.
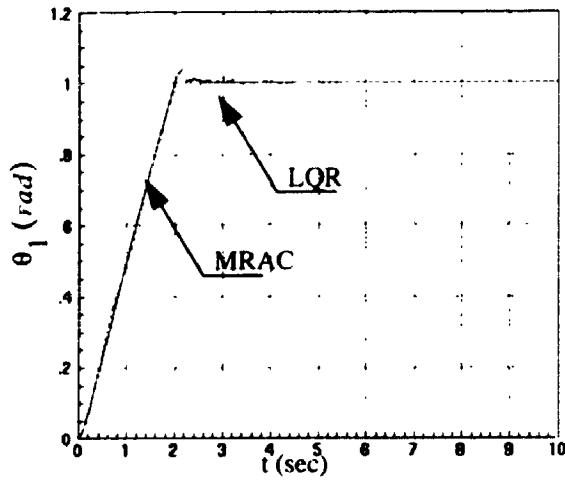
Figure (9.24): Response of the base joint angle (0.5 kg tip mass).
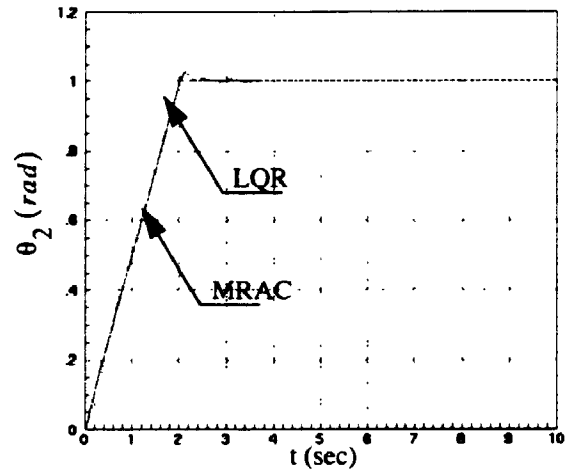
Figure (9.25): Response of the shoulder joint (0.5 kg tip mass).
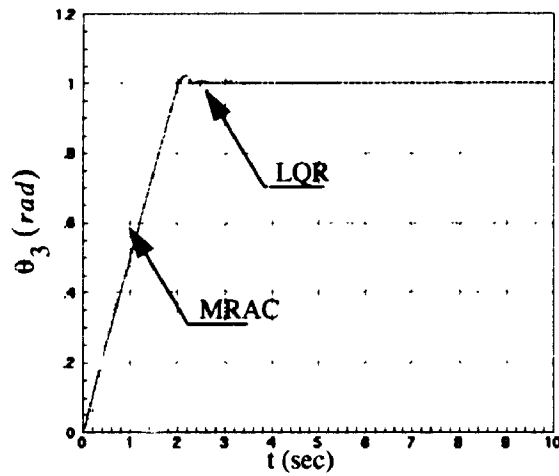
Figure (9.26): Response of the elbow joint (0.5 kg tip mass).
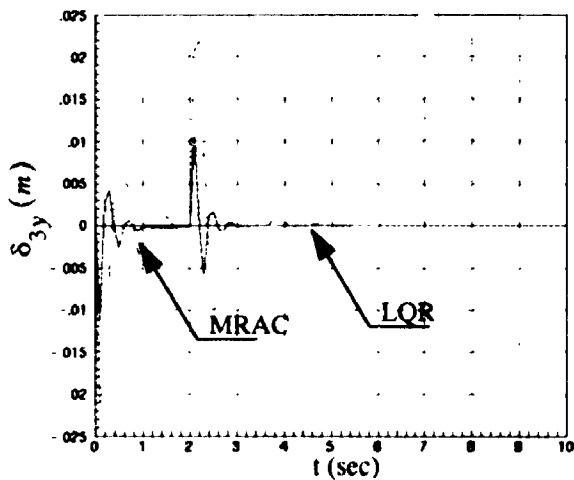
Figure (9.27): Robot end-point deflection in the vertical direction (0.5 kg tip mass).
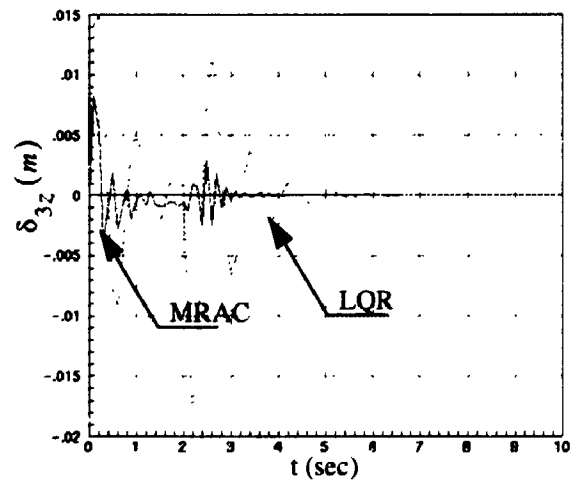
Figure (9.28): Robot end-point deflection in the horizontal direction (0.5 kg tip mass).
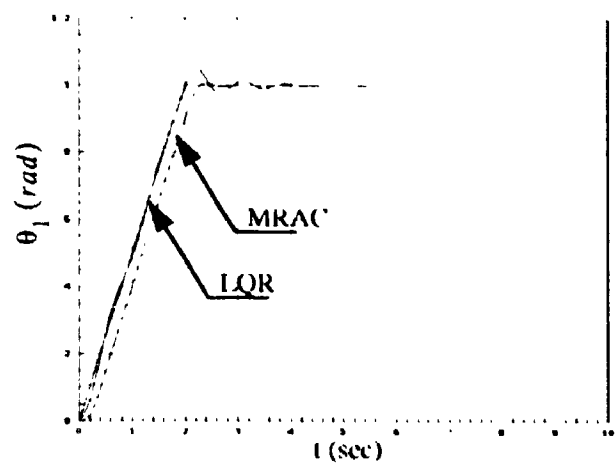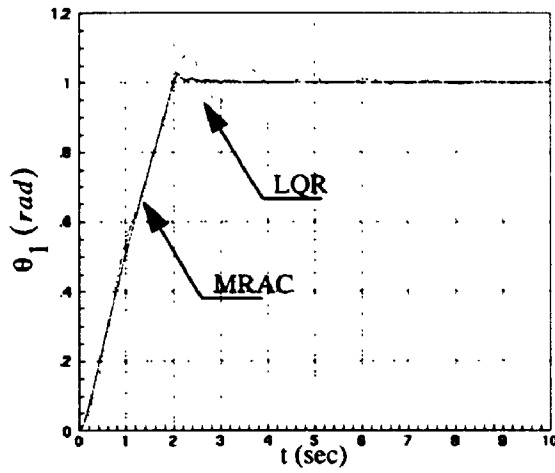
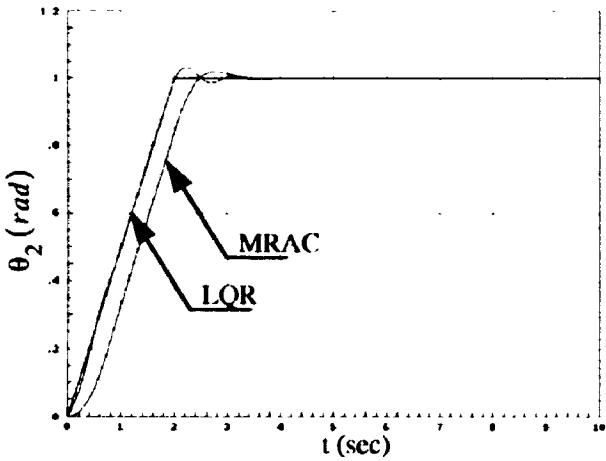Figure (9.29): Response of the base joint angle (1.55 kg tip mass).
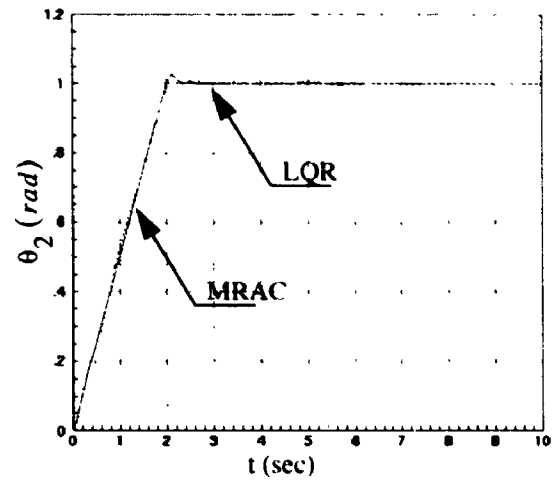
Figure (9.30): Response of the shoulder joint (1.55 kg tip mass).
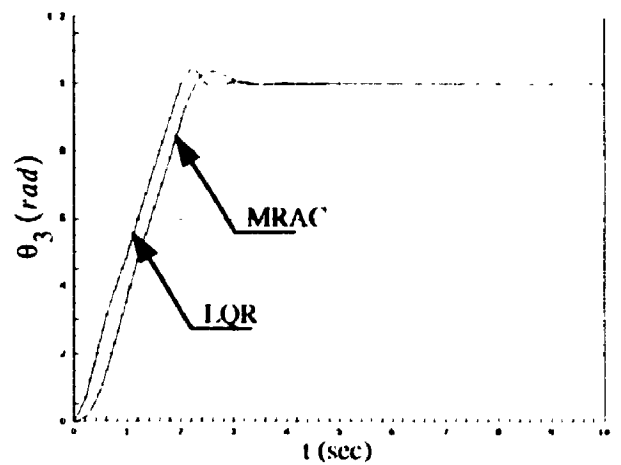

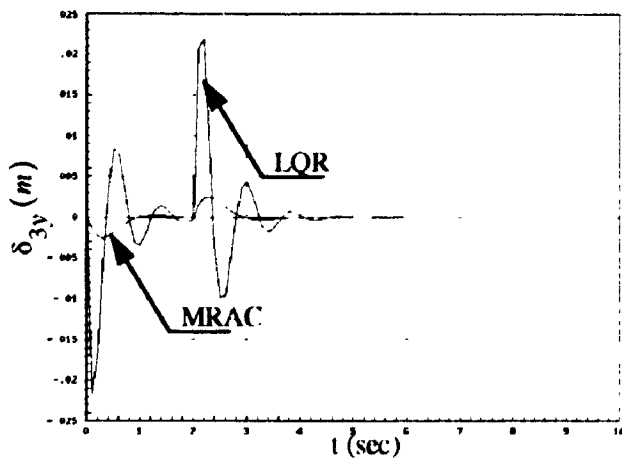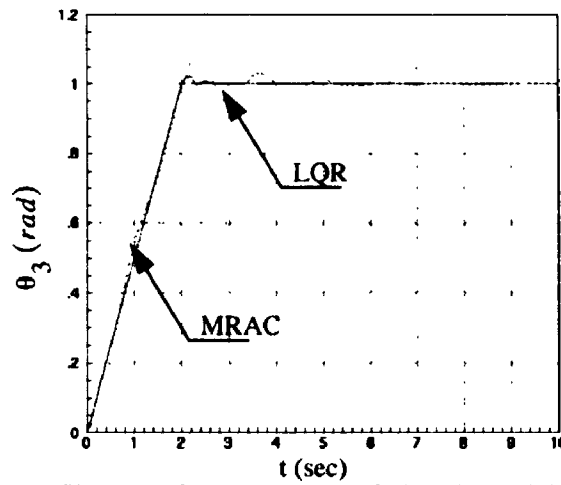Figure (9.31): Response of the elbow joint (1.55 kg tip mass).
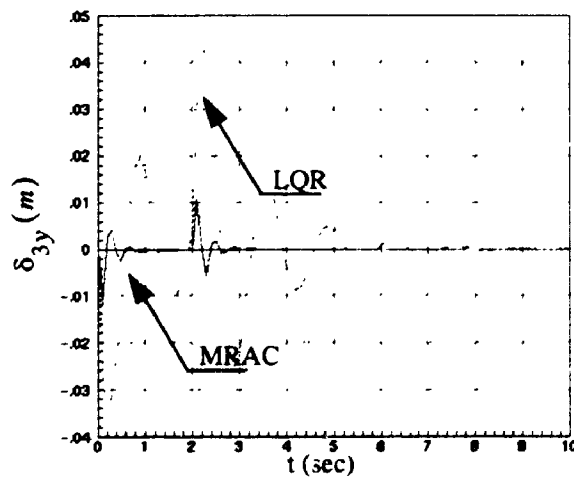
Figure (9.32): Robot end-point deflection in the vertical direction (1.55 kg tip mass).
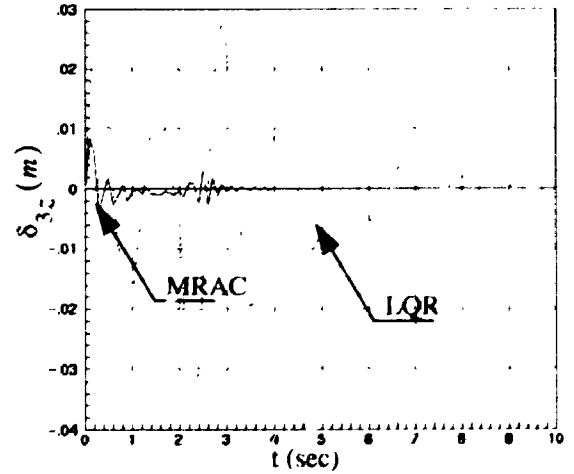
Figure (9.33): Robot end-point deflection in the horizontal direction (1.55 kg tip mass).

9.10 Experimental Results

The control law was applied to the system at a sampling rate of 1 kHz with friction and gravity compensation. Similar to section 9.7, the gravity torques were calculated for the nominal payload of 1.55 kg. Note that this control law requires the input to be the reference position and velocity. The values of α and β were set, by trial and error, to be 1.0, 0.1, respectively.

The response to a one-second time step is shown in Figures (9.34 to 9.38), where the reference input is represented by dashed-lines and the actual response by solid lines. It is shown that the robot tracks the desired trajectory accurately, especially for the shoulder and the elbow. Little overshoot was present in the base, which is attributed to the choice of the reference model. Notice that there is no steady-state error at the joints due to the integral action present in implementing this particular control algorithm.

The end-point deflections are higher than the case presented in section 9.7 due to the relatively faster motion but the residual vibrations are quickly damped under the control action.

By attaching 0.5 kg to the tip of the robot, the response of the system to the one-second time step is shown in Figures (9.39 to 9.43). It is shown, as expected, that the robot performance is very close to the regular payload case. The end-points deflections in the vertical direction are relatively higher than the regular load case due to the increase of dynamic loading at the tip.

In conclusion, this control algorithm gives very good results in trajectory tracking as well as vibration damping, but care has to be taken in choosing the reference model for the system.
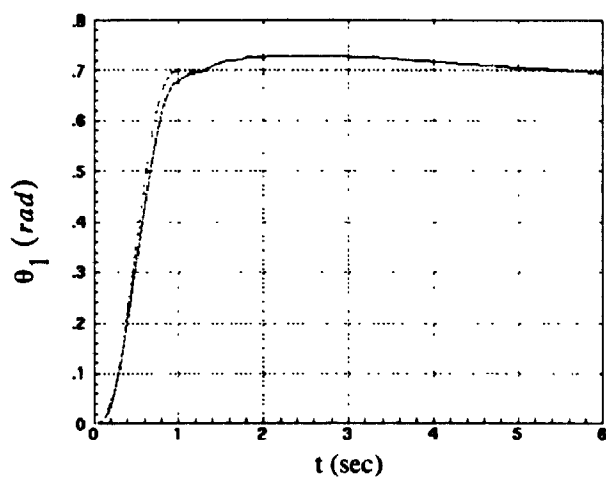
Figure (9.34): Experimental response of the base under modified MRAC.
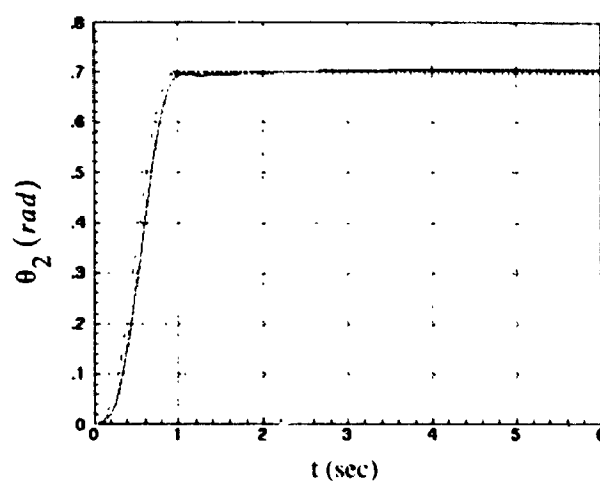
Figure (9.35): Experimental response of the shoulder under modified MRAC.
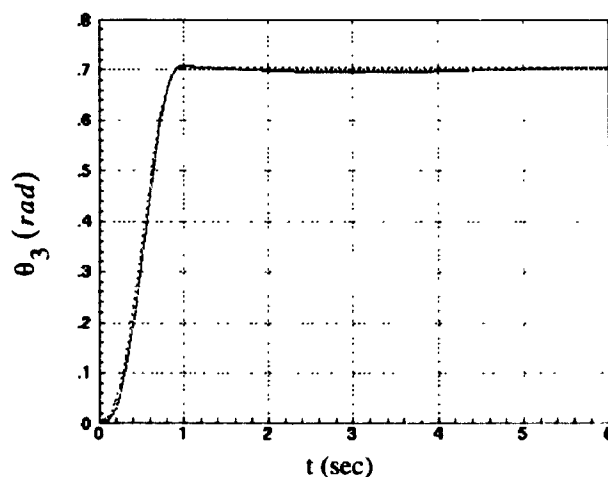
Figure (9.36): Experimental response of the elbow under modified MRAC.
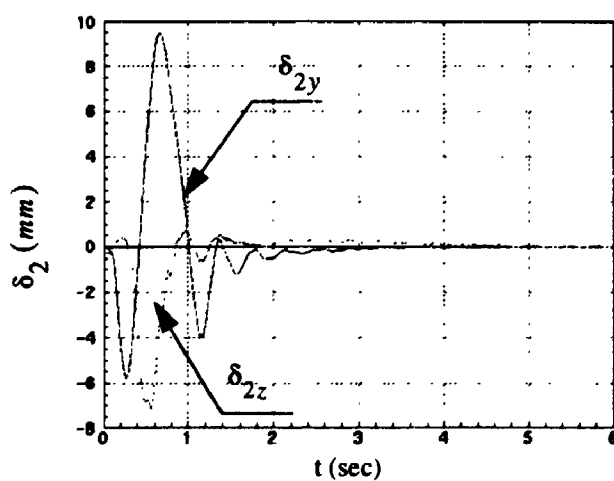
Figure (9.37): Second link tip deflections under modified MRAC.
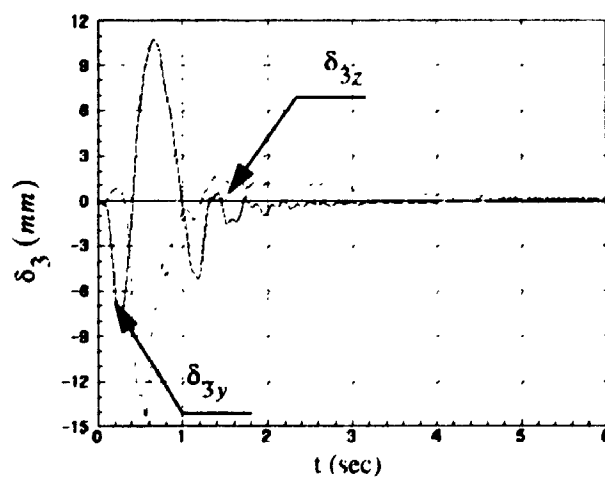
Figure (9.38): Th.rd link tip deflections under modified MRAC.
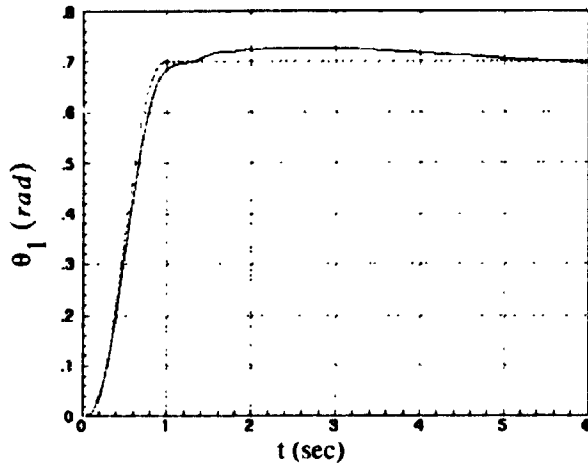
Figure (9.39): Response of the base with higher tip mass under modified MRAC.
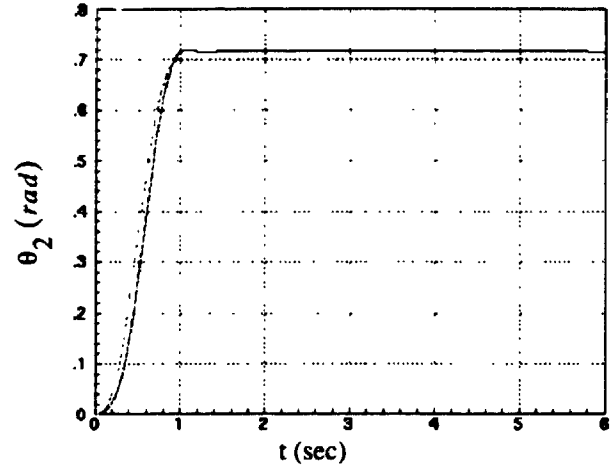
Figure (9.40): Response of the shoulder with higher tip mass under modified MRAC.
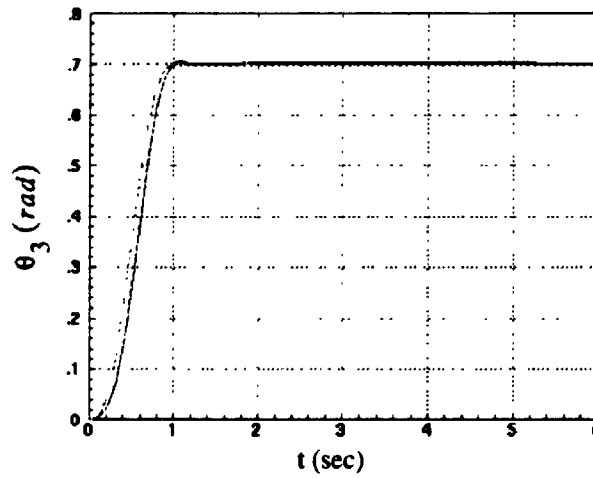
Figure (9.41): Response of the elbow with higher tip mass under modified MRAC.
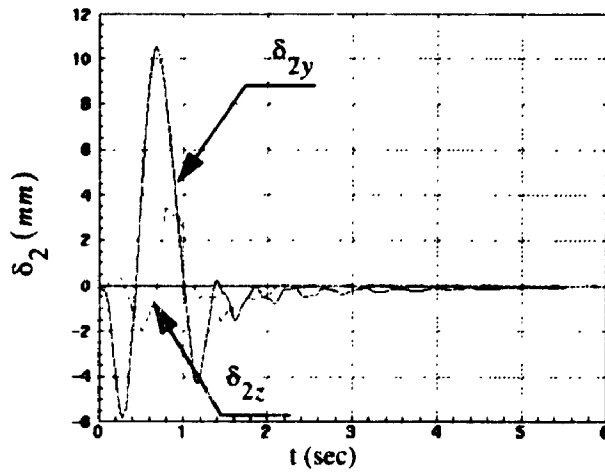
Figure (9.42): Second link tip deflections with higher tip mass under modified MRAC.
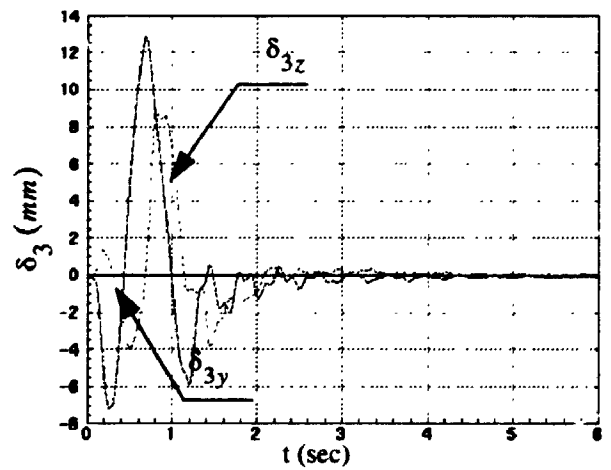
Figure (9.43): Third link tip deflections with higher tip mass under modified MRAC.

- CHAPTER 10 -

INVERSE DYNAMICS CONTROL

The passivity based inverse dynamic approach provides an elegant technique for the adaptive control of rigid robots. In this chapter, an extension of the passivity-based inverse dynamics approach, as applied to a rigid robot, to the case of manipulators with rotary joints and flexible links is derived. It is proved that this technique leads to a globally asymptotically stable system. An adaptive version of the inverse dynamics method is derived and simulated on the experimental manipulator. The internal dynamics of the system was found to be asymptotically stable due to the existence of structural damping.

In order to balance the internal flexible dynamics, a regulator is implemented in the feedback loop in parallel with the inverse dynamics algorithm. Experimental results show good performance of the controller.

10.1 Introduction

The term inverse dynamics (also called computed torque) is a special case of the notation of feedback linearization of nonlinear systems (Slotine and Li [76,78]). Inverse dynamics was used by Craig [18,19] for the adaptive control of rigid manipulators. The same technique was discussed for manipulators with flexible joints by Uhlik [93]. Ortega and Spong [64] classified the inverse dynamics based controllers into two groups. The first class involves obtaining a closed loop system which is linear and decoupled by feeding

151

back the dynamics of the system (Craig [18,19]). The objective of the nonlinear controller for the second class is to preserve the passivity property of the rigid robot and hence achieve global stability (Slotine and Li [77,78]).

When implementing the first technique on manipulators with flexible links, the full linearization property is lost. This was first discussed by De Luca and Siciliano [20] as they attempt to control a nonlinear single flexible link. They showed that a subsystem, which describes the elastic behavior associated with the arm deformation, arises which becomes unobservable and possibly nonlinear. The dynamic characteristics of this subsystems depend on the particular output chosen. If the output was chosen to be the end-point position, it will lead to instability due to the unstable zero dynamics. This unstable zero dynamics also corresponds to the non-minimum phase behavior of the linear model of the single flexible link discussed by Schmitz [71].

10.2 Properties of the Robot Equations of Motion

Centinkunt and Book [14] presented a general symbolic form for the equations of motion for manipulators with compliant links and rotary joints. The general form of the equations of motion is

$$
D(\delta, \theta) \begin{bmatrix} \ddot{\theta} \\ \ddot{\delta} \end{bmatrix} + C(\delta, \theta, \dot{\delta}, \dot{\theta}) \begin{bmatrix} \dot{\theta} \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} F_\theta & 0 \\ 0 & F_\delta \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} \theta \\ \delta \end{bmatrix} + \begin{bmatrix} g(\theta, \delta) \\ 0 \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix} \tag{10.1}
$$

where, $\theta \in R^i$ is the joint angles, and $\delta \in R^{n-i}$ is the vibration generalized coordinates, i is the number of rotary joints, n is the number of degrees of freedom of the system. $D(\delta, \theta) \in R^{n \times n}$ (i) is the inertia matrix. $C(\delta, \theta, \dot{\delta}, \dot{\theta})$ is the coriolis and centrifugal terms matrix. $F_\theta \in R^{i \times i}$ is the linear joint damping matrix. $F_\delta \in R^{n \times n}$ is the structural damping

matrix. $g(\theta, \delta) \in R^i$ is the gravitational forces vector. $K \in R^{(n-i) \times (n-i)}$ is the linear stiffness matrix. $\tau \in R^i$ is the applied joint torque and the zero term in the torque exists if constrained modes are used in the modal expansion formulation, i.e. the slope at the hub is zero[1]. Note that the joint flexibility is not included in the analysis. Define

$$q^T = [\theta^T, \delta^T] \tag{10.2}$$

where, $q \in R^n$. The gravity and stiffness terms were defined as

$$G(q) = \begin{bmatrix} g(\theta, \delta) \\ K\delta \end{bmatrix} \tag{10.3}$$

The final form for the equations of motion is

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + F\dot{q} + G(q) = T \tag{10.4}$$

This form will be used since it is similar to the form of the equations of motion for rigid robots. Here, $T \in R^n$ is the applied torque vector given by the following equation

$$T = [\tau_i, 0_{n-i}]^T = B\tau \tag{10.5}$$

$$\tau = B^\dagger T \tag{10.6}$$

where $B \in R^{n \times i}$ is a mapping matrix, and $B^\dagger = (B^T B)^{-1} B^T$ is the pseudo inverse for B.

Although the equations of motion are nonlinear and complex they have fundamental properties which can be used to facilitate the control design. For the case of a manipulator with rotary rigid joints and flexible links, it has the following properties [64]:

Property 1: The inertia matrix $D(q)$ is symmetric, positive definite and uniformly bounded as a function of q. In addition $0 < \| D(q) \|$.

Property 2: There is an independent control input for each joint but not for each

---

1. Although this case is considered here, the derivation is for the general case.

degree of freedom as opposed to the rigid manipulators case.

Property 3: Some of the constant parameters (e.g. links mass, inertia, etc.) appear as coefficients of known functions of the generalized coordinates. (In case of rigid robots, all the parameters satisfy this condition).

Property 4: The matrix $(\dot{D}(q) - 2C(q, \dot{q}))$ is skew symmetric. The proof of this property is as follows: Neglecting the damping, the derivative of the kinetic and potential energies must equal to the power input by the actuators minus the gravitationa. forces

$$\frac{1}{2}\frac{d}{dt}(\dot{q}^T D\dot{q} + \delta^T K\delta) = \dot{q}^T(\tau - g) \tag{10.7}$$

By expanding the derivative

$$\dot{q}^T D\ddot{q} + \frac{1}{2}\dot{q}^T \dot{D}\dot{q} + \dot{\delta}^T K\delta = \dot{q}^T(\tau - g) \tag{10.8}$$

Then, by substituting from the robot equation (equation 10.1) we get [78]

$$\dot{q}^T(\dot{D} - 2C)\dot{q} = 0 \tag{10.9}$$

Property 5: The stiffness and damping matrices are positive definite and usually constant. The joint damping matrix is diagonal. The structural damping matrix is usually taken to be proportional to the stiffness matrix[2].

The fourth property is the key for the passivity property as will be explained later. Ortega and Spong [64] showed that the inverse dynamics control algorithm can be classified by whether they use the skew-symmetry property or not.

---

2. without loss of generality, the damping matrix will be neglected in the passivity controller derivation

10.3 Passivity Structure for Flexible Manipulators

A mapping between x and y $(x \rightarrow y)$ is said to be passive if and only if [64]

$$<x|y>_T = \int_0^T x^T y \, dt \geq -\beta \tag{10.10}$$

for some $\beta > 0$ and for all T. For rigid robots, the mapping between the torque and the joint velocity is passive. This relationship can be extended to robots with flexible links. The proof is as follows: The Hamiltonian H is the sum of the kinetic and potential energies, i.e.

$$H = \frac{1}{2}\dot{q}^T D(q)\dot{q} + P(q) \tag{10.11}$$

where, $P(q)$ is the potential energy. The total derivative of the Hamiltonian must equal the power input by the actuators, hence

$$\frac{dH}{dt} = \dot{q}^T T \tag{10.12}$$

The mapping $\dot{q}$ to T is

$$<\dot{q}|T> = \int_0^T \dot{q}^T T \, dt = \int_0^T dH = H(T) - H(0) \geq -H(0) = <\dot{\theta}|\tau> \tag{10.13}$$

since $H(T)$ is non negative for all T, which implies that the mapping between the torque τ and joint velocity $\dot{\theta}$ for manipulators with flexible links and rotary joints is passive.

10.4 Joint Based Inverse Dynamics

The joint-based direct inverse dynamics approach will be presented to be compared with the passivity-based inverse dynamics algorithms. The development of the control law is according to the inversion algorithm by Hirschorn [32]. The derivation is accomplished by choosing an output y associated with the flexible manipulator system, the input torque

τ that is capable of exactly reproducing the given trajectory y^d (t) can be derived by means of system inversion technique. If the output is chosen to be the joint angles θ, the inversion technique will lead to a stable system since the relationship between the input torque and output joint angle is passive.

Equation (10.1) can be put in the form

$$
\begin{bmatrix} D_\theta & D^T_{\theta\delta} \\ D_{\theta\delta} & D_\delta \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\delta} \end{bmatrix} + \begin{bmatrix} n_\theta \,(\dot{\theta}, \theta, \delta, \dot{\varepsilon}) \\ n_\delta \,(\dot{\theta}, \theta, \delta, \dot{\delta}) \end{bmatrix} + \begin{bmatrix} F_\theta & 0 \\ 0 & F_\delta \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} g \\ K\delta \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix}
\tag{10.14}
$$

where n_δ, n_θ are nonlinear vec... Taking the inverse of the inertia matrix D to be

$$
B = D^{-1} = \begin{bmatrix} B_{11} & B^T_{12} \\ B_{12} & B_{22} \end{bmatrix}
\tag{10.15}
$$

Equating for $\ddot{\theta}$, $\ddot{\delta}$

$$
\begin{bmatrix} \ddot{\theta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} B_{11} \,(\tau - g - F_\theta\dot{\theta} - n_\theta) + B^T_{12} \,(-n_\delta - F_\delta\dot{\delta} - K\delta) \\ B_{12} \,(\tau - g - F_\theta\dot{\theta} - n_\theta) + B_{22} \,(-n_\delta - F_\delta\dot{\delta} - K\delta) \end{bmatrix}
\tag{10.16}
$$

Choosing the output y to be the joint angle θ and applying the inversion approach [20,64], we can see from equation (10.16) that

$$
\ddot{y} = B_{11} \,(\tau - g - F_\theta\dot{\theta} - n_\theta) + B^T_{12} \,(-n_\delta - F_\delta\dot{\delta} - K\delta)
\tag{10.17}
$$

Choosing θ^d as the desired joint trajectory. Note that θ^d has to be a function twice differentiable with time. The linearizing control torque can be obtained by setting

$$
\ddot{y} = \ddot{\theta}^d
\tag{10.18}
$$

Solving for τ

$$
\tau (\theta^d, \theta, \dot{\theta}, \delta, \dot{\delta}) = g + n_\theta + F_\theta\dot{\theta} + B^{-1}_{11} \,(\ddot{\theta}^d + B^T_{12} \,(n_\delta + F_\delta\dot{\delta} + K\delta))
\tag{10.19}
$$

The input-output relation from the external reference θ^d to the joint angle θ is now

linear and in the form of double integration as a result of choosing the inversion control torque $\bar{\tau}$ given by equation (10.19) [20].

The above control algorithm can be implemented in an open-loop or closed-loop scheme as demonstrated by De Luca and Siciliano [20]. However, the closed-loop implementation is more robust against the uncertainty in the robot parameters. This is achieved by replacing $\ddot{\theta}^d$ by $\ddot{\theta}_r$ where θ_r is the "virtual" reference trajectory. The relationship between $\ddot{\theta}_r$ and $\ddot{\theta}^d$ is given by

$$\ddot{\theta}_r = \ddot{\theta}^d + K_v(\dot{\theta}^d - \dot{\theta}) + K_p(\theta^d - \theta) \tag{10.20}$$

where K_p and K_v are positive definite matrices. The virtual reference trajectory is distinguished from the desired trajectory since it is implemented in the closed loop to guarantee robustness. The closed-loop control torque is obtained by substituting from equation (10.20) to equation (10.19) to be

$$\tau = \bar{\tau}(\ddot{\theta}^d, \dot{\theta}, \theta, \dot{\delta}, \delta) + B_{11}^{-1}(K_p(\theta^d - \theta) + K_v(\dot{\theta}^d - \dot{\theta})) \tag{10.21}$$

The control algorithm (equation (10.21)) can be implemented according to the block diagram shown in Figure (10.1) by setting

$$K(p) = K_v p + K_p \tag{10.22}$$

Figure (10.1): Adaptive inverse dynamics block diagram.

where p is Laplace operator. The implementation of this control algorithm in real-time is almost impossible since it involves the estimation of the full inertia matrix and inverting it. Uhlik [93] showed that the inversion control algorithms have to be implemented at high sampling rate otherwise the continuous time approximation will begin to fail and result in instability. He concluded that for flexible systems, the sampling rate must be 30 to 50 times the highest natural frequency desired to be controlled.

The adaptive version of this control algorithm can not be derived either since the control torque is not a linear function of the constant manipulator parameter due to the inversion of the inertia matrix [64].

10.5 The Passive-Controller Design

The control law for the actuator torques are required to get the manipulator, knowing its parameters, to track a desired trajectory $q^d(t)$, in the presence of model imprecision. This can be achieved by restricting the tracking error to lie on a sliding surface given by [78,77]

$$s = \dot{\tilde{q}} + \Lambda \tilde{q} = 0 \tag{10.23}$$

where Λ is a constant positive definite matrix, and $\tilde{q}(t) = q(t) - q_d(t)$ is the tracking error. Assuming the control law to be

$$T = D(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + G(q) - K_D s \tag{10.24}$$

where $q_r(t) = q_d(t) - K(p)\tilde{q}$ and $K(p)$ is according to the block diagram shown in Figure (10.1). According to Slotine and Li [78], $K(p)$ is taken to be

$$K(p) = \frac{\Lambda}{p} \tag{10.25}$$

or q_r is taken to be

$$q_r = q_d - \Lambda \int_0^t \tilde{q} dt \qquad (10.26)$$

s can be shown to be

$$s = \dot{q} - \dot{q}_r = \dot{\tilde{q}} + \Lambda \tilde{q} \qquad (10.27)$$

$$\dot{s} = \ddot{q} - \ddot{q}_r \qquad (10.28)$$

Choosing the Lyapunov function candidate V to be

$$V(t) = \frac{1}{2} s^T D(q) s \qquad (10.29)$$

Where, $D(q)$ is always positive definite, hence, $V(t)$ is always positive. Differentiating $V(t)$ yields

$$\dot{V}(t) = \frac{1}{2} s^T \dot{D} s + s^T D \dot{s} \qquad (10.30)$$

Substituting for s and $\dot{s}$

$$\dot{V}(t) = \frac{1}{2} s^T \dot{D} \dot{q} - \frac{1}{2} s^T \dot{D} \dot{q}_r + s^T D \ddot{q} - s^T D \ddot{q}_r \qquad (10.31)$$

Substituting for $D\ddot{q}$ and $D\ddot{q}_r$ from the controller and manipulator equations respectively,

$$\dot{V}(t) = \frac{1}{2} s^T \dot{D} \dot{q} - \frac{1}{2} s^T \dot{D} \dot{q}_r + s^T (T - C\dot{q} - G) - s^T (T - C\dot{q}_r - G + K_D s) \qquad (10.32)$$

$$= \frac{1}{2} s^T (\dot{D} - 2C) \dot{q} - \frac{1}{2} s^T (\dot{D} - 2C) \dot{q}_r - s^T K_D s$$

Using *property 4* (equation (10.9)), the final expression for $\dot{V}(t)$ is

$$\dot{V}(t) = -s^T K_D s \qquad (10.33)$$

This expression implies that the control law leads to a globally stable system with zero steady-state error for the joint position. Using equation (10.6), the actuators torques are

$$\tau = B^\dagger (D(q) \ddot{q}_r + C(q, \dot{q}) \dot{q}_r + G(q) - K_D s) \qquad (10.34)$$

which can be viewed as taking the rows that correspond to the joint angles. Hence, the implementation of the control algorithm does not involve calculating all terms of D, C, G, rather the rows that correspond to the control torques are only considered. This will reduce the time required for the calculation and hence make it easier for real-time control implementation.

The control algorithm (equation (10.24)) can be implemented in view of the block diagram shown in Figure (10.1) by setting

$$\ddot{q}_r = \ddot{q}_d + p^2 K(p) e \tag{10.35}$$

In general, $K(p)$ can be chosen to be of the form of PID controller, in order to improve the robustness [69,64], as follows

$$K(p) = K_p + K_v p + \frac{K_I}{p} \tag{10.36}$$

where K_p, K_v and K_I are positive definite matrices, chosen by trial and error to achieve good trajectory tracking performance.

10.6 Extension to the Adaptive Controller

In the previous derivation, the controller is derived assuming that the robot parameters are known. Furthermore, with the use of the sliding modes technique, robust stability is guaranteed for the controller/manipulator. However, the above technique can be extended to the adaptive case as opposed to the direct inverse dynamic technique.

The idea behind the adaptive controller is to update the manipulator parameters in the control equation on-line by implementing a regressor driven by the tracking error. The derivation of this controller will be based on *property 3*. Although the rigid manipulator equations are linear in the parameters, the flexible manipulator equations are linear for

almost all the parameters except for the links length (due to the bending stiffness dependency on length cubed). Hence, the manipulator equations can be written as

$$Y(q, \dot{q}, \ddot{q}, \alpha)a = T \qquad (10.37)$$

where α are the nonlinear parameters (links lengths) and a are the linear parameters, usually the joints inertia, links masses, payload and links stiffness. The global convergence of the controller is demonstrated by considering the following Lyapunov function

$$V(t) = \frac{1}{2}s^T D s + \frac{1}{2}\tilde{a}^T \Gamma \tilde{a} \qquad (10.38)$$

where $\tilde{a} = a - \hat{a}$ is the parameter estimation error vector, and Γ is a positive definite matrix. The controller equation is as follows

$$T = \hat{D}(q)\ddot{q}_r + \hat{C}(q, \dot{q})\dot{q}_r + \hat{G}(q) - K_D s \qquad (10.39)$$

where

$$\begin{aligned} \hat{D}(q) &= \tilde{D}(q) + D(q) \\ \hat{C}(q, \dot{q}) &= \tilde{C}(q, \dot{q}) + C(q) \\ \hat{G}(q) &= \tilde{G}(q) + G(q) \end{aligned} \qquad (10.40)$$

Differentiating V yields

$$\dot{V} \quad s^T D \dot{s} + \frac{1}{2}s^T \dot{D} s + \tilde{a}^T \Gamma \dot{\tilde{a}} \qquad (10.41)$$

Substituting for $\dot{s}$ and s yields

$$\begin{aligned} \dot{V}(t) &= \frac{1}{2}s^T \dot{D}\dot{q} - \frac{1}{2}s^T \dot{D}\dot{q}_r + s^T(T - C\dot{q} - G) - s^T(T - \hat{C}\dot{q}_r - \hat{G} + K_D s) + \tilde{a}^T \Gamma \dot{\tilde{a}} \\ &= \frac{1}{2}s^T(\dot{D} - 2C)\dot{q} - \frac{1}{2}s^T(\dot{D} - 2C)\dot{q}_r - s^T K_D s + s^T(\tilde{D}\ddot{q}_r + \tilde{C}\dot{q}_r + \tilde{G}) + \tilde{a}^T \Gamma \dot{\tilde{a}} \end{aligned} \qquad (10.42)$$

Using *property 4* (equation (10.5)), the final expression for $\dot{V}(t)$ is

$$\dot{V} = -s^T K_D s + s^T(\tilde{D}\ddot{q}_r + \tilde{C}\dot{q}_r + \tilde{G}) + \tilde{a}^T \Gamma \dot{\tilde{a}} \qquad (10.43)$$

From equation (10.37)

$$\tilde{D}\ddot{q}_r + \tilde{C}\dot{q}_r + \tilde{G} = Y\tilde{a} \qquad (10.44)$$

Substituting into equation (10.42) we get

$$\dot{V} = -s^T K_D s + \tilde{a}^T (\Gamma \dot{\tilde{a}} + Y^T s) \qquad (10.45)$$

The above equation suggests that $\dot{V}$ is always negative by choosing an adaptation law so that the term between brackets vanishes. That is

$$\dot{\tilde{a}} = -\Gamma^{-1} Y^T s \qquad (10.46)$$

Thus, the adaptive controller is globally stable. The structure of the control law (equations (10.46) and (10.39)) can be implemented according to the block diagram shown in Figure (10.1). Note that Γ is chosen so that it guarantees parameter convergence.

10.7 Effect of the Flexible Dynamics

The application of the control torque (equation (10.34)) is more involved since there is no applied torque for the elastic degrees of freedom. The dynamics of the flexible variables is of particular interest due to the need to damp out the arm vibration when it reaches a stop.

By expanding equation (10.14) to isolate the flexible variables equation

$$D_\theta \ddot{\theta} + D^T_{\theta\delta} \ddot{\delta} + n_\theta + F_\theta \dot{\theta} + g = \tau \qquad (10.47)$$

$$D_{\theta\delta} \ddot{\theta} + D_\delta \ddot{\delta} + n_\delta + F_\delta \dot{\delta} + K\delta = 0 \qquad (10.48)$$

Equation (10.47) is the rigid variables (θ) equation which is balanced by the control torque using either the direct inverse dynamics approach or the passivity based approach. However, there is no applied torque at the right hand side of equation (10.48) which represents the flexible variables behavior.

The stability of the flexible variables when the arm reaches a stop condition is closely related to the zero-dynamics, as demonstrated by De Luca and Siciliano [20]. Under this condition the angular velocities $\dot\theta$ and the angular accelerations $\ddot\theta$ are set to zeros in equation (10.48). Consequently, the flexible variables equation becomes

$$D_\delta\ddot\delta + r_\delta\dot\delta + K\delta = 0 \qquad (10.49)$$

which is a nonlinear equation of the flexible variables as D_δ is nonlinear. Since D_δ, F_δ, K are positive definite matrices, the internal dynamics are asymptotically stable and eventually the residual oscillation will be damped out. If $F = 0$, i.e, there is no passive damping in the structure, the flexible internal dynamics will be critically stable.

10.8 Simulation Results

The above control algorithm was simulated on the robot based on the parameters shown in Table (6.1). The trajectory function was taken to be a one-second rise-time fifth order step according to Chapter 7. The simulation was carried out on the control system CAD package MATRIX$_X$[37].

In the first simulation run, the model parameters were assumed to be known and the controller was based on equation (10.34) (i.e. it did not include any adaptation). The value of K_D was chosen to be

$$K_D = diag\left(\,[\,40,\,40,\,40\,]\,\right) \qquad (10.50)$$

The outer loop was chosen to be a PD control law, where k_p and k_v are chosen for the base joint to be 10.0, 6.0, for the shoulder 12.0, 8.0, and for the elbow joint to be 12.0, 8.0, respectively. The response for the joint angles is shown in Figures (10.2,10.3 and 10.4).
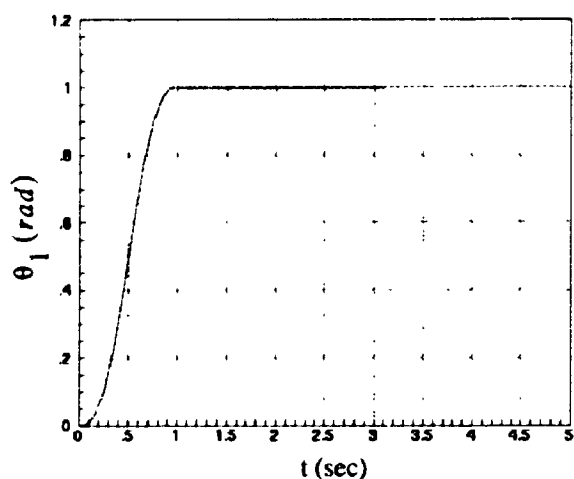
Figure (10.2): Response of the base joint under inverse dynamics control.
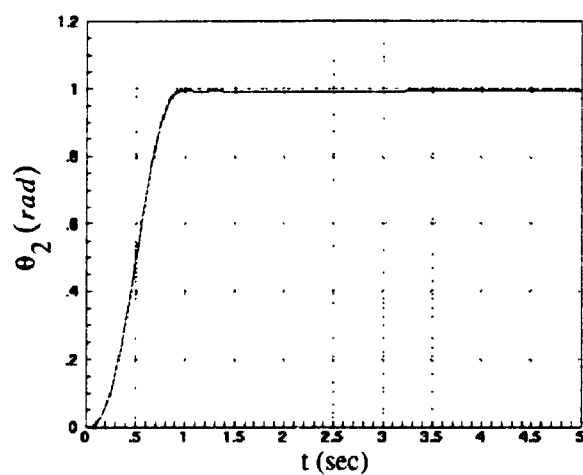
Figure (10.3): Response of the shoulder joint under inverse dynamics control.
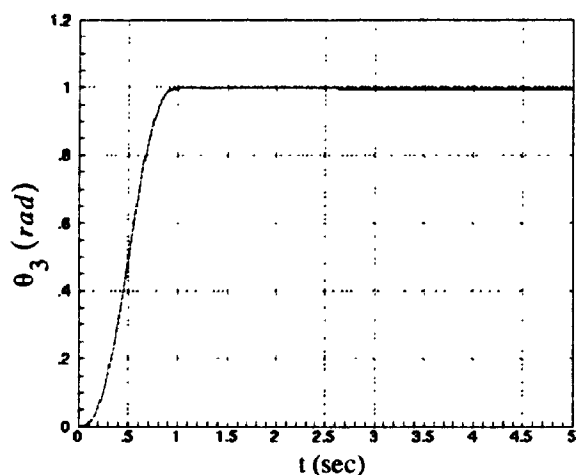
Figure (10.4): Response of the elbow joint under inverse dynamics control.
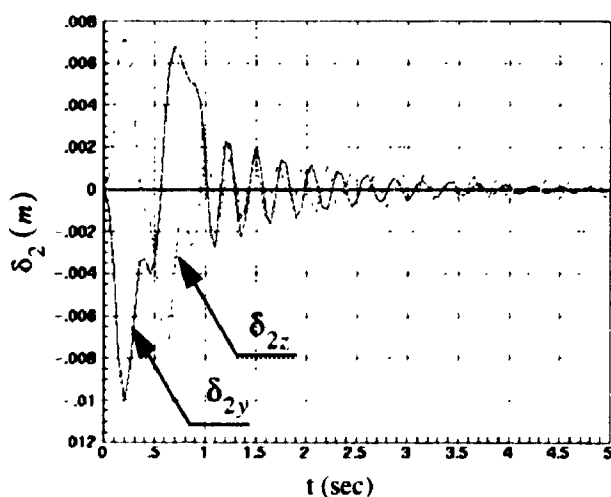
Figure (10.5): Tip deflection of the second link under inverse dynamics control.
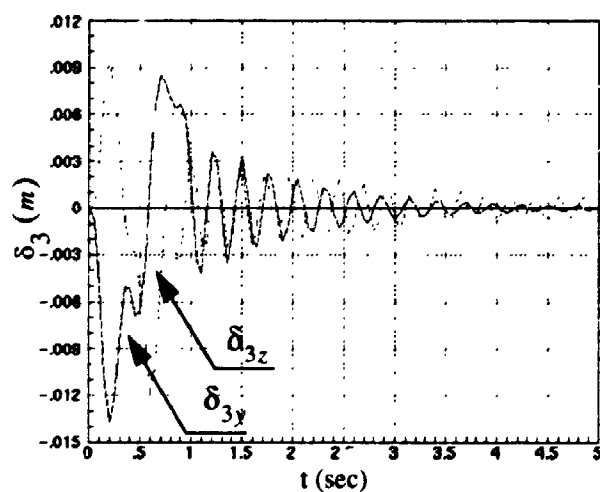
Figure (10.6): Tip deflection of the third link under inverse dynamics control

The desired trajectories are represented by dotted-line and the actual trajectories by the solid lines. It is shown that the robot tracks the desired trajectory accurately with a negligible error compared to the other control strategies.

The link tips vibrations in the horizontal and vertical directions are shown in Figures (10.5, 10.6), where the horizontal deflections are represented by dotted-lines. It is shown that the robot exhibits excessive vibrations at the end motion. The vibrations are damped only due to the existence of small structural damping in the system (damping ratios less than 1% for the vibration modes according to Chapter 5). However, the amplitude of the tip deflections are less than those of the PD control presented in Chapter 7.

The second simulation was carried out on the system assuming unknown parameters case, where the payload mass was set for the controller to be 1.5 kg while for the actual system load was 2.0 kg. The other parameters were left the same. The unknown parameters chosen for the regression were the three joints inertia, elbow mass, and payload.

The controller was based on equations (10.46) and (10.39) where K_D and the outer loop gains were taken similar to the previous run. Γ was chosen to be

$$\Gamma = diag\left(\left[20, 20, 20, 20, 20\right]\right) \tag{10.51}$$

The joints response are shown in Figures (10.7,10.8,10.9). It is shown that although some parameters are unknown, the robot still tracks the desired trajectory accurately. The deflections of the links are shown in Figures (10.11, 10.12). Similar to the previous case, the tips vibration are damped only under the influence of structural damping. However, the amplitude is a little larger than in the previous case.

By observing this controller, it is shown that structural damping plays an important role in this controller. By increasing the structural damping, i.e. building the robot with
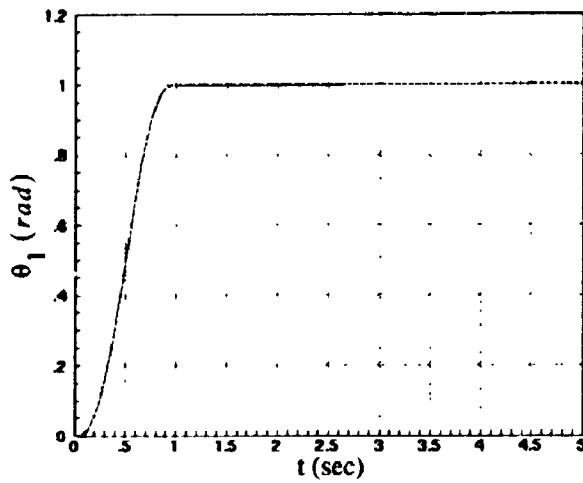
Figure (10.7): Response of the base joint under adaptive inverse dynamics control.
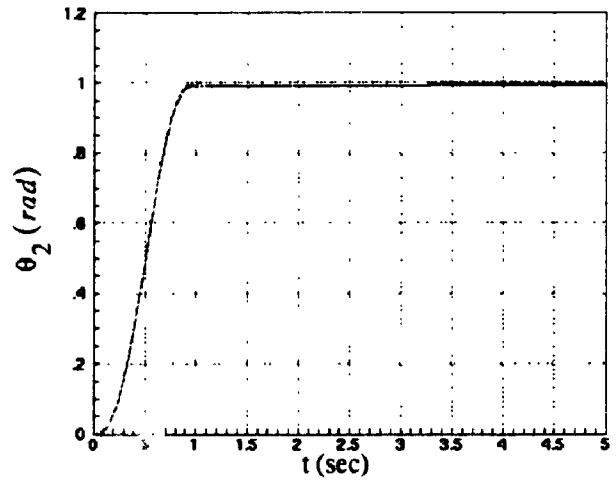
Figure (10.8): Response of the shoulder joint under adaptive inverse dynamics control.
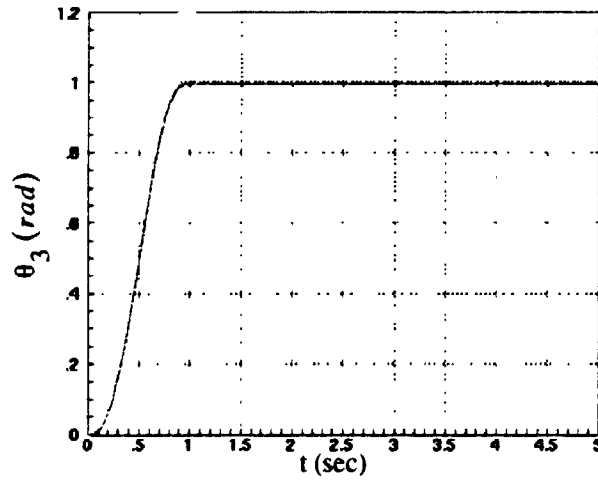
Figure (10.9): Response of the elbow joint under adaptive inverse dynamics control.
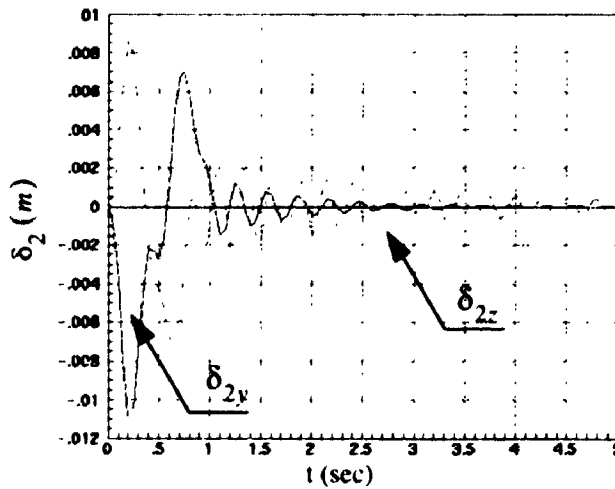
Figure (10.10): Tip deflection of the second link under adaptive inverse dynamics control.
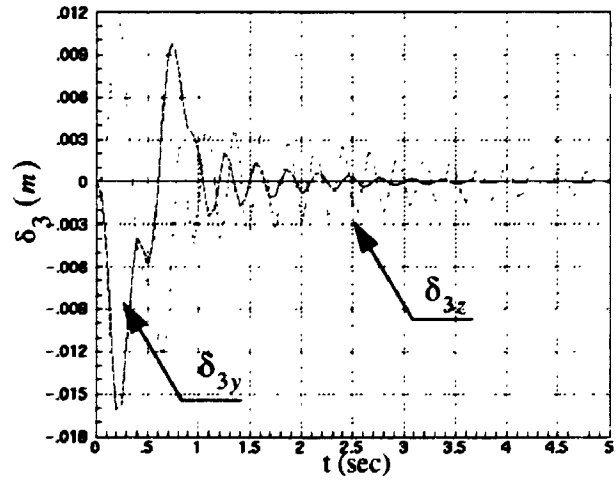
Figure (10.11): Tip deflection of the third link under adaptive inverse dynamics control.
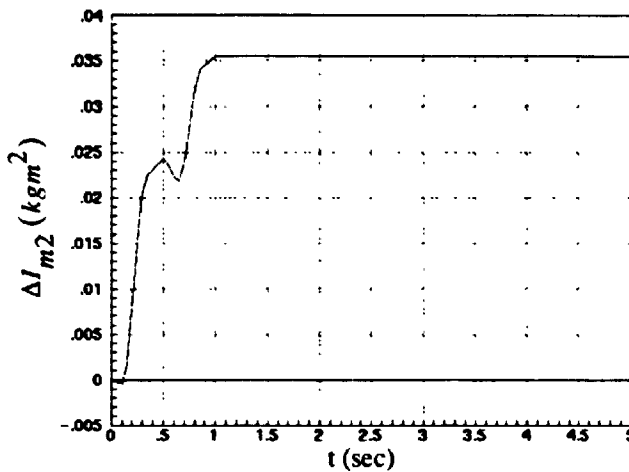
Figure (10.12): Variation of the shoulder inertia under the adaptation law.
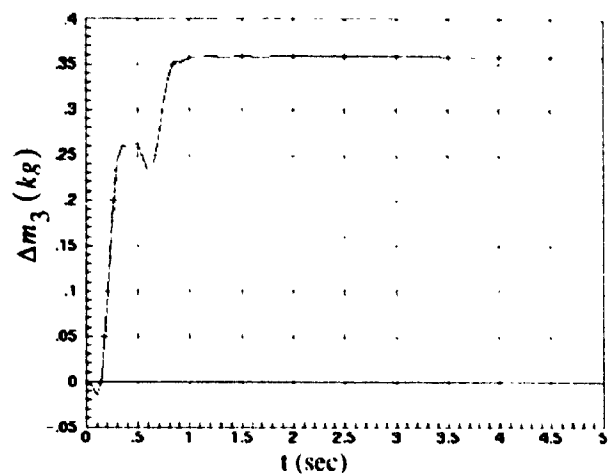
Figure (10.13): Variation of the payload (m_3) under the adaptation law.

composite material [53] or using constraint layer damping [1,2] for the links, the performance of the inverse dynamics controller will be substantially improved.

An example for parameters regression is the change of the payload and shoulder inertia shown in Figure (10.12, 10.13). It is shown that both do not converge to the exact values, which is 0.0 for the shoulder inertia and 0.45 for the payload. This is attributed to the characteristics of the trajectory as it is not persistently exciting. As explained in Slotine and Li [76,77,78], the reference trajectory must sufficiently excite the dynamic response of the system so that the effects of the various parameters can be distinguished. However, the tracking performance is still acceptable.

10.9 Development of a Composite Controllers

It was concluded that the control algorithm, based on the inverse dynamics technique, relies on the existence of links structural damping to decay of the end-points residual vibration.

A composite controller is proposed that consists of implementing a state feedback
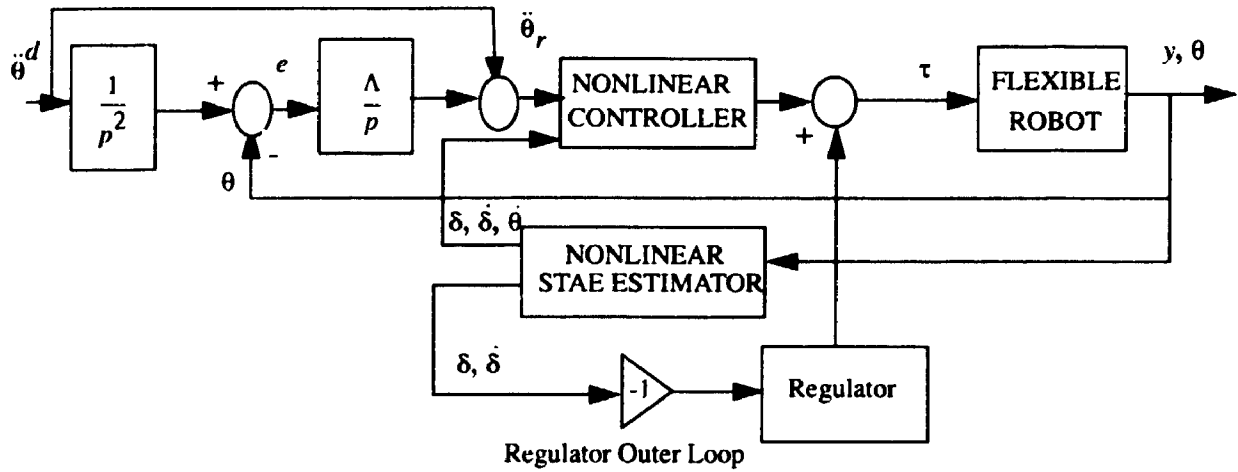
Figure (10.14): Block diagram for the composite controller.

regulator in the feedback loop with the inverse dynamics controller as shown in Figure (10.14). Note that the state-regulator takes the elastic degrees of freedom (δ) and their derivative as input and outputs torque. The calculation of the regulator gains is based on minimizing a certain performance index for the linearized system similar to Chapter 8, while considering the elastic deflection part only.

The primary advantage of using the regulator is to damp out the residual vibration by providing the torques for that assuming perfect model tracking by the adaptive controller. Hence, for the unknown parameters case, the control input equations are

$$\tau = B^{\dagger} \left(D(q)\ddot{q}_r + C(q, \dot{q})\dot{q}_r + G(q) - K_D s \right) - K_\delta [\dot{\delta}, \delta]^T \tag{10.52}$$

where K_δ is the elastic-deflections gains matrix and the negative sign assumes that the desired elastic deflections are zero. Note that for the unknown parameter case, the controller equations are modified similar to equation (10.52).

10.10 Simulation of the Composite Controller

The system was simulated assuming an unknown parameter case similar to that in Section 10.8. The joint angles response are as shown in Figures (10.15, 10.16 and 10.17). It is shown that the robot tracks the desired trajectory perfectly.

The response for the elastic deflections is shown in Figures (10.18, 10.19). It is shown that the residual vibrations are damped by the controller. Notice that the regulator has a negligible effect on the rigid body response of the system since the required torques to damp out the vibration (generated by the regulator) are much less than the torques required for the rigid body motion (generated by the inverse dynamics controller).

10.11 Experimental Results

The composite controller, with friction and gravity compensation, is applied to the system at a sampling rate of 500 Hz. The gravity torques were calculated for the nominal payload of 1.55 kg. The system parameters were taken to be close to the real values. Note that this control strategy takes as reference-input the position, velocity and acceleration.

The values of K_D, Γ and the outer loop PD gains were taken to be the same as given in Section 10.9. Additional integral action was added to the outer loop and the gains for it were 1.0, 2.0 and 2.0 for the base, shoulder and elbow, respectively. K_δ in equation (10.52) was taken to be the scheduled gains developed in Chapter 8. The implementation of the scheduled gains gave better response than that with fixed gains.

The response to a one-second rise-time fifth order step is shown in Figures (10.20 to 10.24), where the reference input is represented by dashed-lines and the response by
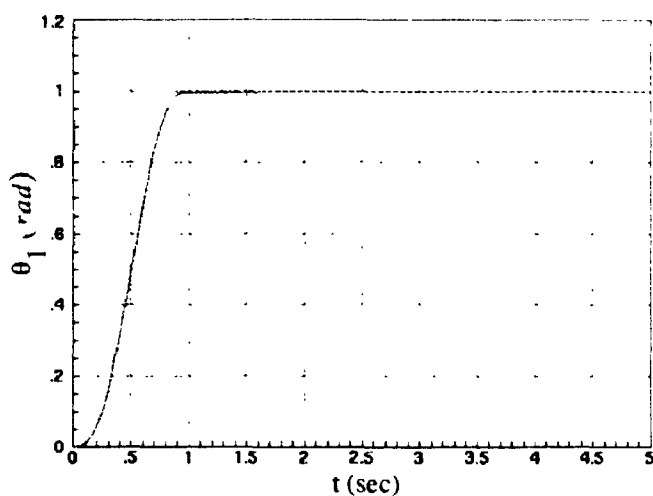
Figure (10.15): Response of the base joint under adaptive inverse dynamics control.
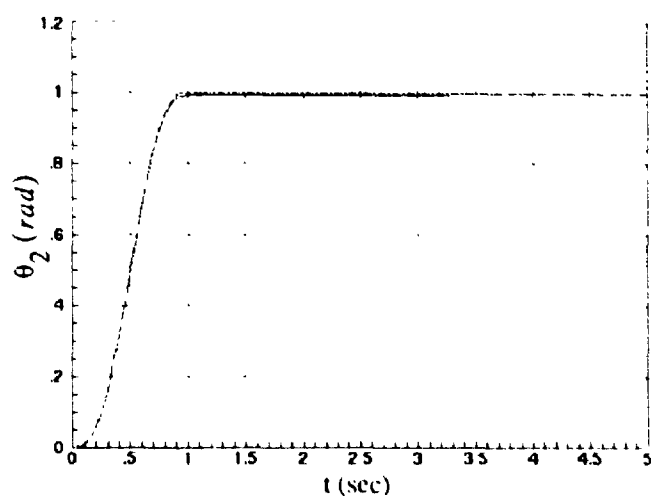
Figure (10.16): Response of the shoulder joint under adaptive inverse dynamics control.
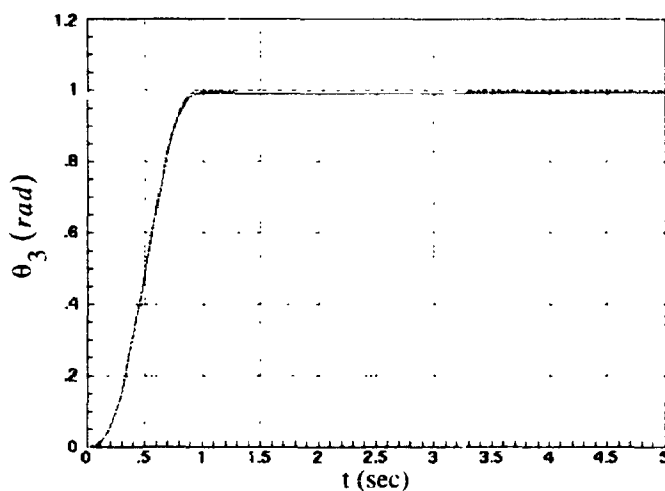
Figure (10.17): Response of the elbow joint under adaptive inverse dynamics control.
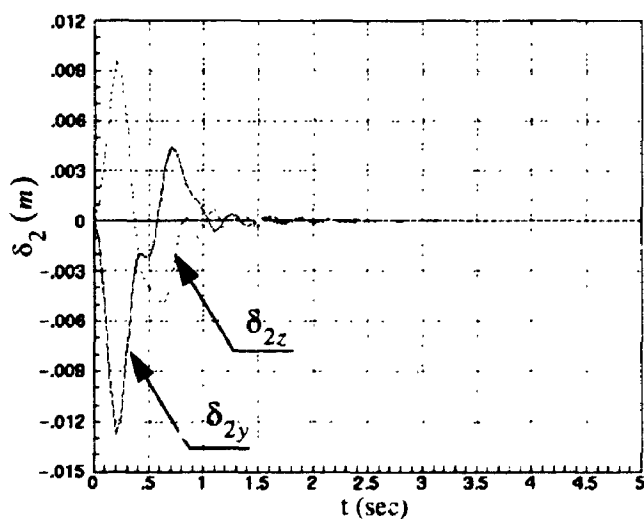
Figure (10.18): Tip deflection of the second link under adaptive inverse dynamics control.
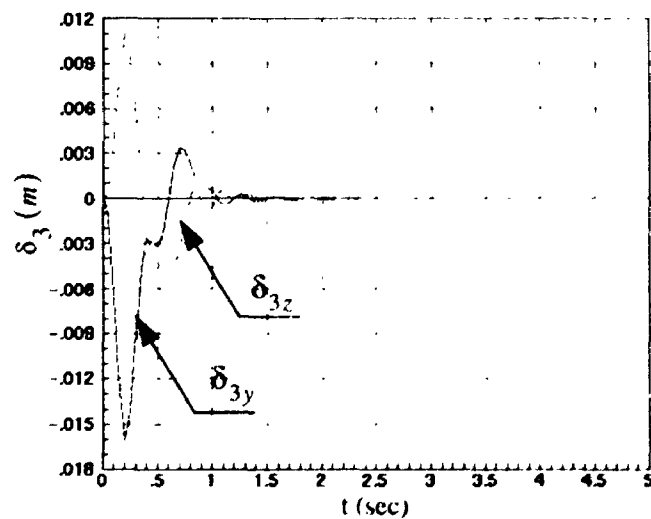
Figure (10.19): Tip deflection of the third link under adaptive inverse dynamics control.
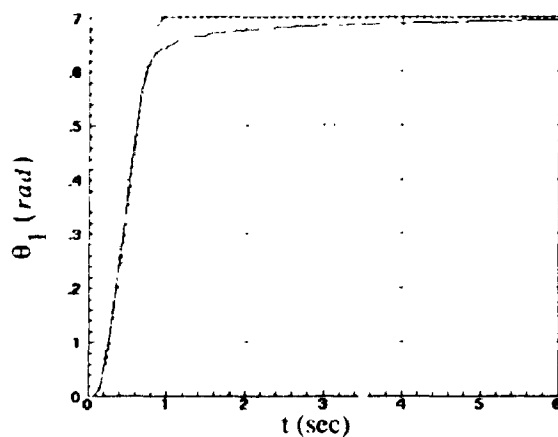
Figure (10.20): Experimental response of the base under the composite controller.
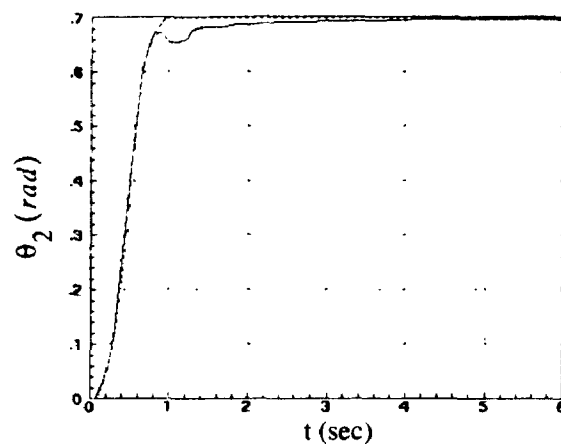
Figure (10.21): Experimental response of the shoulder under the composite controller.
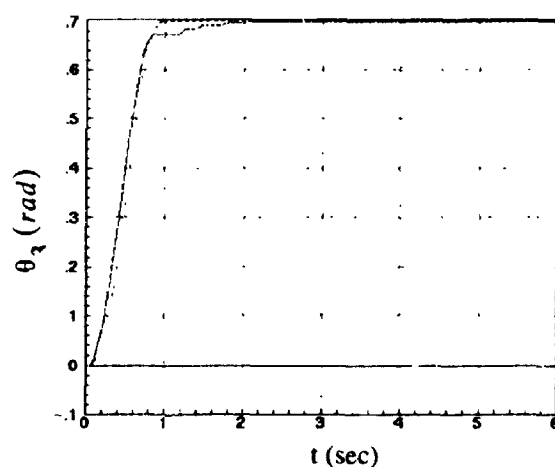
Figure (10.22): Experimental response of the elbow under the composite controller.
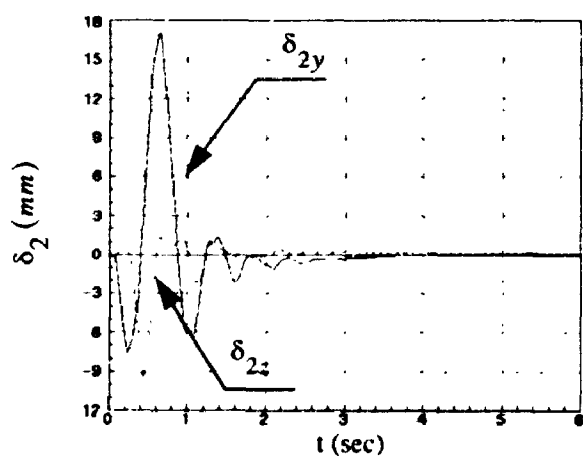
Figure (10.23): Second link tip deflections under the composite controller.
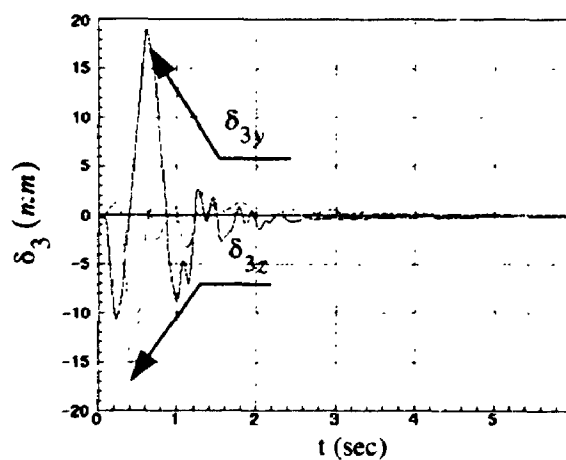
Figure (10.24): Third link tip deflections under the composite controller.

solid lines. It is shown that the robot tracks the desired trajectory accurately. However, there are some disturbances as the robot reaches the desired position due to zeroing the friction effect.

It is shown that the residual vibrations are quickly damped out under the controller action although the robot tends to reach the desired position relatively slowly under the integral action.

By attaching an additional 0.5 kg to the tip of the robot, the response of the system is shown in Figures (10.25 to 10.29). It is shown that the robot behavior was similar to the previous case. The end-point deflections are relatively higher than the previous case; however, the vibrations are quickly damped out.

In conclusion, this controller combines the advantages of two controllers, namely, the inverse dynamics controller for the parameter adaptation action and the high tracking performance, and the optimal controller for damping out the vibration. However, it has the disadvantage of the heavy computation involved in its implementation.

Without using a DSP-based real-time computer it would seem to be infeasible to implement this kind of controller in real-time.

Modification of this controller may include on-line identification and compensation of column friction to improve the tracking performance.
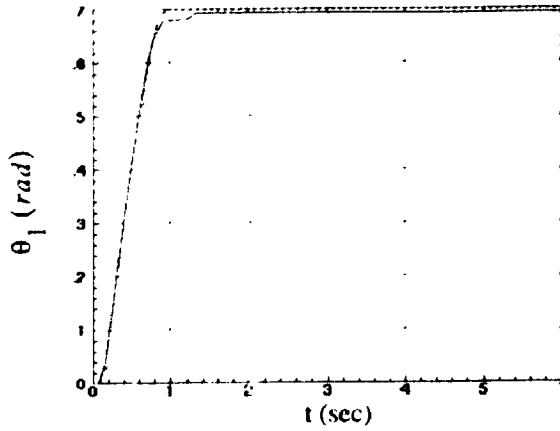
Figure (10.25): Base response with higher payload under the composite controller.
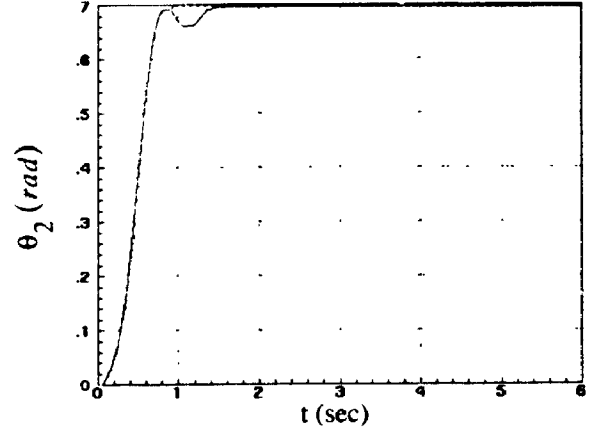
Figure (10.26): Shoulder response with higher payload under the composite controller.
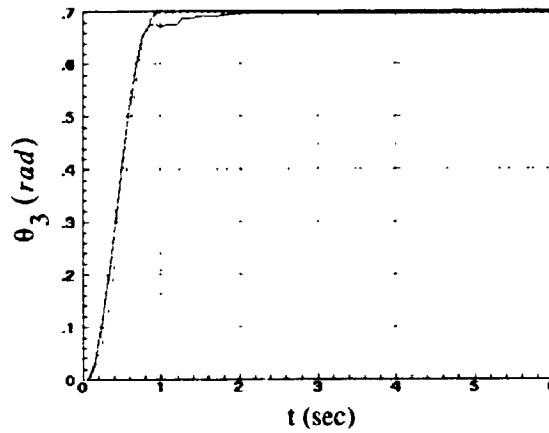
Figure (10.27): Elbow response with higher payload under the composite controller.
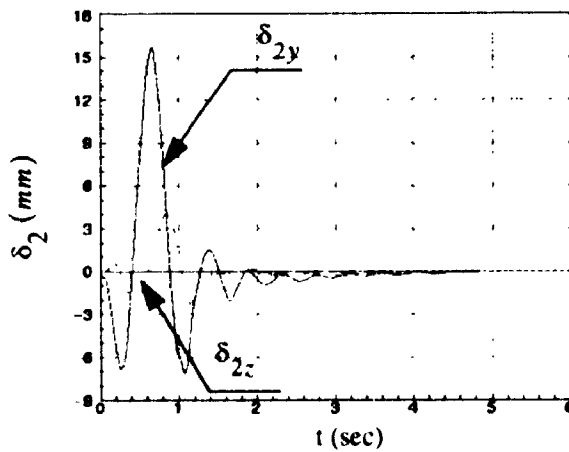
Figure (10.28): Second link tip deflections with higher payload under the composite controller.
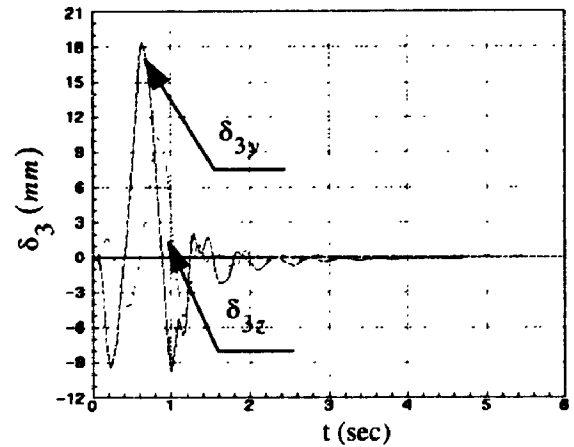
Figure (10.29): Third link tip deflections with higher payload under the composite controller.

- CHAPTER 11 -

PERFORMANCE COMPARISON FOR THE CONTROLLERS

In this chapter, the controllers designed and applied on the robot are compared in a unified way. The performance measures are classified into tracking measures and disturbance rejection measures. Before presenting the results, a summary of the controllers, tests, and performance measures are described.

11.1 Summary of the Designed Controllers

In this thesis, four main controllers are successfully designed and implemented namely: the collocated PD controller, the GS-LQR (gain-scheduling linear quadratic regulator), the MRAC (model reference adaptive controller), the INV-GS (adaptive inverse dynamics with gain-scheduling). A summary of these controllers is shown in Table (11.1).

As a performance measure for the simplicity of implementing these controllers, the maximum allowable sampling rate is defined. It is obtained by increasing the sampling rate till the DSP becomes unable to run the code. A quick indication of the used number of FLOPS (floating point operations per second) can be estimated knowing that the real-time controller runs at 33 MFLOPS.

The PD controller, described in Chapter 7, uses only the joint angles. Hence, it does not have much influence on damping the vibration of the end-point. Because of its

simplicity, it can be implemented at a high sampling rate.

The GS-LQR, described in Chapter 8, represents the first trial toward a non-collocated controller design. This controller uses a look-up table, based on the robot configuration, to find the appropriate gains. Hence, it is configuration adaptable but not payload adaptable. It is relatively simpler than the other two controllers.

The MRAC, described in Chapter 9, tries to get the robot to follow a reference model. As a result, this controller is fully adaptive with respect to the payload and configuration. However, the performance is limited by the choice of the reference model.

The INV-GS, described in Chapter 10, uses the adaptive inverse dynamics to linearize the robot and a secondary loop of GS-LQR to damp out the vibration. This controller is also adaptive. However, it needs substantial computational overhead to linearize the robot which makes it impossible to be used on relatively slow hardware.

Table 11.1: Summary of the controllers.

	PD	GS-LQR	MRAC	INV-GS
Sensors	joint angles	joint angles & accelerometers	joint angles & accelerometers	joint angles & accelerometers
Sampling rate	5 kHz	4 kHz	1 kHz	500 Hz
Max. allowable sampling rate	10 kHz	6 kHz	2.5 kHz	700 Hz

11.2 Definition of the Performance Measures

The performance measures are divided into trajectory tracking performance, and disturbance rejection performance. Trajectory tracking performance measures the ability of the controller to track a desired trajectory with minimum residual vibration. Whereas,

disturbance rejection performance measures the ability of the controller to damp the end-point vibration if it is disturbed from its original position.

In this section, quantitative measures are defined and used to compare various controllers.

11.2.1 Trajectory Tracking Measures

The following definitions are used to assess the ability of the controller to track a desired trajectory with end-point vibration damping.

Joint Tracking Error

It is defined, for each angle, as the maximum deviation from the desired path

$$e_{\theta_i}^{max} = max\left(\left\|e_{\theta_i}(t)\right\|\right) \tag{11.1}$$

where $\|x\|$ is the norm of x.

Joint Steady-State Error

It is defined as the error between the final position of the joint and the desired final position.

$$e_{\theta_i}^{ss} = \lim_{t \to \infty}\left\|e_{\theta_i}(t)\right\| \tag{11.2}$$

Maximum End-Point Deflection

It is defined as the maximum amplitude of the end-point residual vibration after finishing the trajectory.

$$\delta_{y,z}^{max} = max\left(\left\|\delta^{tip}_{y,z}(t)\right\|\right) \qquad\qquad \forall t \geq T_r \tag{11.3}$$

where, T_r is the time taken to execute the reference trajectory (the rise-time for the fifth order step).

End-Point Settling Time

It is a measure of the time elapsed, after the end of the maneuver, for the end-point to reach an acceptable level of deflection or acceleration in the horizontal and vertical directions. Note that the value for the acceptable deflection was set, for simplicity, to 5 mm peak-to-peak. Mathematically, it can be defined as

$$min \ \{ T^s_{(y, z)} \ \big| \ \big| \delta^{tip}_{y, z} \big| \leq \text{Settle\_Threshold} \ \} \qquad \forall t \geq T^s \qquad (11.4)$$

11.2.2 Disturbance Rejection Measures

To examine the ability of the controller in responding to external disturbances, the end-point of the manipulator is disturbed from its regulated configuration and observed while it returns back. The controller disturbance rejection can be judged by the following measures:

Settling Time

Measures how quickly the robot returns to its regulation position. It is defined similar to equation (11.4). The settle threshold is taken to be 5% of the initial value in the horizontal and vertical directions.

End-Point Vibration Energy

It is defined by measuring the energy in the decaying end-point vibration in the horizontal and vertical direction. This can be calculated by

$$E_{y, z} = \int_0^\infty \delta^2_{y, z} (t) \big|_{tip} dt \qquad (11.5)$$

where, for simplicity, the integral is calculated for a period of 10 sec.

It should be noted that for the disturbance rejection, the robot has to be at an arbitrary nominal po:.:tion under the controller action. Since the system is nonlinear, these measures will depend on the robot position.

11.3 Experimental Results

In this section, the performance of the four developed controllers is presented. Also, the torques developed by the controllers are compared.

11.3.1 Trajectory Tracking Performance

As a standard test for all the controllers, the maneuver was set to be a $40°$ fifth-order step in one second for all the joints which ends with an elbow-up posture. The results are shown in Table (11.2) for the case of normal payload and 0.5 kg increase in the payload.

Table 11.2: Trajectory tracking performance.

	PD		PD (high gains)		GS-LQR		MRAC		INV-GS	
Δm (kg)	0.0	0.5	0.0	0.5	0.0	0.5	0.0	0.5	0.0	0.5
θ_1^{ss} (deg.)	2.4	4.5	-0.5	-0.5	-0.4	-0.1	-0.2	-0.2	-1.6	-0.1
θ_2^{ss} (deg.)	0.4	1.9	-0.1	-1.0	0.2	0.9	0.1	0.3	0.9	1.0
θ_3^{ss} (deg.)	1.9	3.5	0.0	0.5	0.9	1.4	-0.1	-0.0	0.1	0.1
θ_1^{max} (deg.)	8.2	10.5	3.6	4.1	5.0	3.0	6.0	6.0	-2.0	-1.0
θ_2^{max} (deg.)	7.4	9.8	1.9	3.2	4.2	7.0	2.0	4.5	-2.7	-2.8
θ_3^{max} (deg.)	4.1	6.5	1.5	1.8	2.3	4.5	1.4	1.0	-2.8	-2.0
δ_y^{max} (mm)	2.6	4.5	3.5	6.5	1.1	2.5	1.2	1.3	1.4	1.5
δ_z^{max} (mm)	2.0	4.0	4.8	5.1	3.2	6.2	1.5	1.7	2.7	2.5
T_y^s (sec.)	5.5	8.1	15.2	17.5	1.3	3.2	3.8	4.0	4.0	5.2
T_z^s (sec.)	5.0	7.2	8.5	9.2	4.2	7.5	4.5	4.8	4.8	4.7

The high-gain PD is shown in the comparative study to demonstrate the effect of increase of the PD gains. It is shown that the PD (with low-gains) has better vibrational performance than the high-gains PD. However, the tracking performance is bad. The high-gain PD has the best tracking performance for the joint angles but the worst for the vibration damping.

The performance of the GS-LQR tends to deteriorate, in the horizontal direction, with the addition of payload. However, it has the best performance in damping the vertical vibration in the regular payload case.

The adaptive controllers performance remains acceptable even with the change of the payload whereas the performance of the other controllers tends to suffer more vibration. From the above results, it is shown that the MRAC and the INV-GS seem to have the best performance.

11.3.2 Disturbance Rejection Performance

In this section, the disturbance rejection performance of the experimental manipulator under various control schemes is demonstrated. Disturbance can enter into the system at many locations and can have many characteristics e.g. noise at the sensors, impulsive loads applied to the robot, sudden release of the payload, etc.

For the robot case, one particular type of disturbance was considered which consists of applying a quick move to the end-point and observing it returning back to its regulated position. This is equivalent to applying an impulsive load at the end-point.

The manipulator response, since it is nonlinear, will depend on the robot position and the amplitude of motion. For this reason, the robot was tested around a regulated

nominal position of (40°, 40°, 40°), measured from the vertical home position, with regular payload and larger payload cases.

Figure (11.1 to 11.5) shows the disturbance rejection performance for the different controllers. The horizontal deflections are represented by dashed lines while the vertical deflections are represented by solid lines and the results are taken for the case of a regular payload (Notice that the vertical scaling is not the same for all figures). The energy of the end-point as well as the settling time in the horizontal and vertical directions are shown in Table (11.3).

Table 11.3: Disturbance rejection performance.

	PD		PD (high gains)		GS-LQR		MRAC		INV-GS	
Δm (kg)	0.0	0.5	0.0	0.5	0.0	0.5	0.0	0.5	0.0	0.5
E_y^{max}(mm^2)	172	194	258	310	78.5	110.	110.	120.	124	140.
E_z^{max} (mm^2)	180	215	264.	298	115.	280	160	180.	137	145.
T_y^s (sec.)	5.0	6.5	8.0	10.	2.0	4.5	3.0	3.3	3.2	3.5
T_z^s (sec.)	5.5	7.0	9.2	9.5	3.5	9.0	4.5	4.8	3.5	4.0

It is shown that both the energy and settling time show the controller effect in damping out the end-point vibration. Lower values of energy and settling time indicate better performance for the controller.

By increasing the gains of the PD, the joint becomes more stiff which results in more settling time and vibration at the end-point. As for the GS-LQR, by increasing the payload, the end-point tends to "wiggle". This means that the system poles tend to be have lower damping and become critically stable in the horizontal direction.

As for the MRAC and the INV-GS, they preserve acceptable performance with the increase of payload.
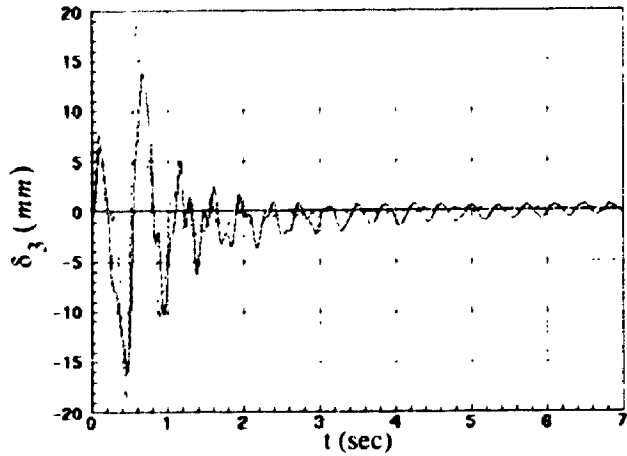
Figure (11.1): End-point disturbance rejection under PD control.
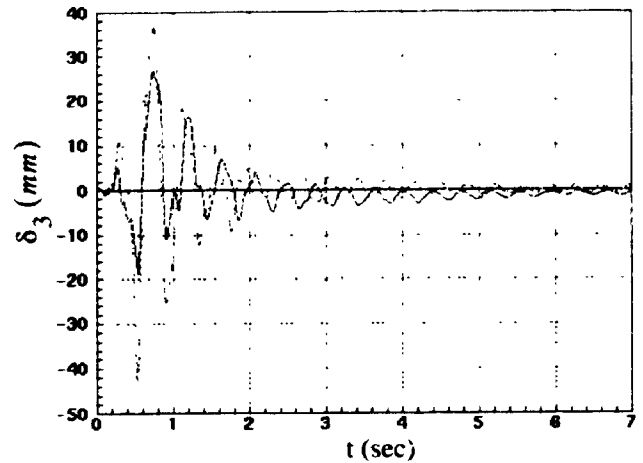
Figure (11.2): End-point disturbance rejection under PD control with high gains.
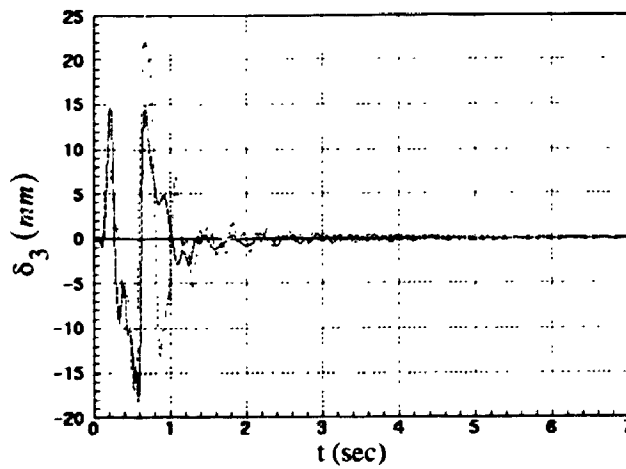
Figure (11.3): End-point disturbance rejection under GS-LQR control.
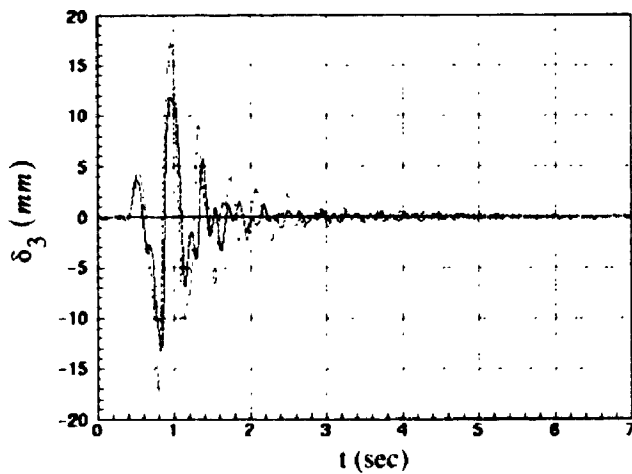
Figure (11.4): End-point disturbance rejection under MRAC control.
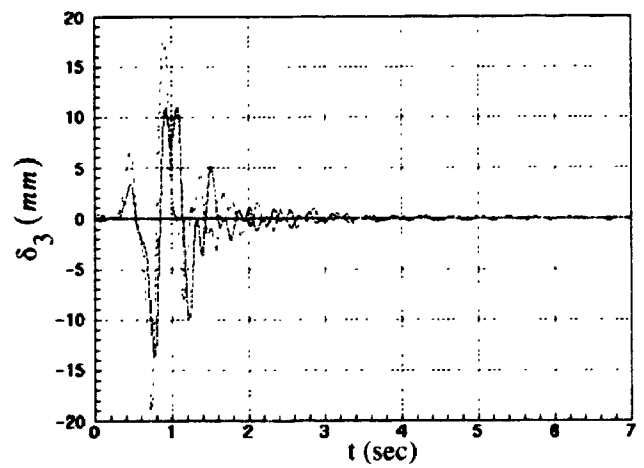
Figure (11.5): End-point disturbance rejection under INV-GS control.

11.3.3 Required Torques

In this section, a comparison of the required torques by the different controllers is presented, where u_1, u_2, u_3 are the input voltages for the base, shoulder and elbow respectively.

For each motor, four components of torques are due to the following: the rigid-body motion and vibration suppression, the gravity compensation, and the friction compensation.

The output torques for the different controllers have chatter due to the friction compensation dither. Notice that the up and down DC values of the torques, after the maneuver, are due to th columb friction compensation which depends on the sign of the vibration suppression torque.

As for the INV-GS, the velocity estimation is smoothed by reducing the cut-off frequency of the low-pass filter. Hence, the torque signals do not undergo excessive chatter.

The level of the torques for the different controllers are close to each other. Note that the INV-GS demands higher level of torques than the rest of controllers.

Trials to run the robot downward at a similar maneuver but with a 0.5 second rise-time did not succeed and resulted in collapsing the robot due to shoulder amplifier saturation as the voltage level exceeded 4 volts. However, moving it upward was successful.
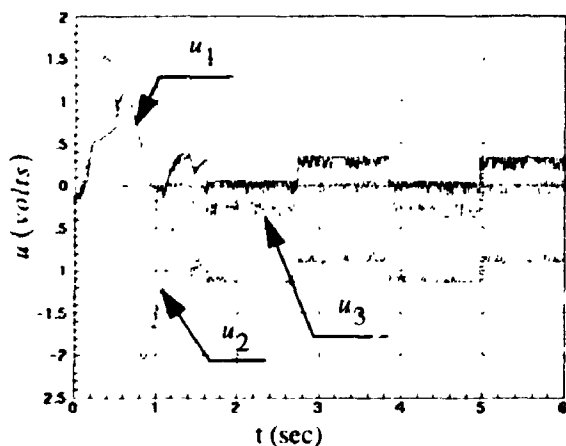
Figure (11.6): Actuators control signals under PD control.
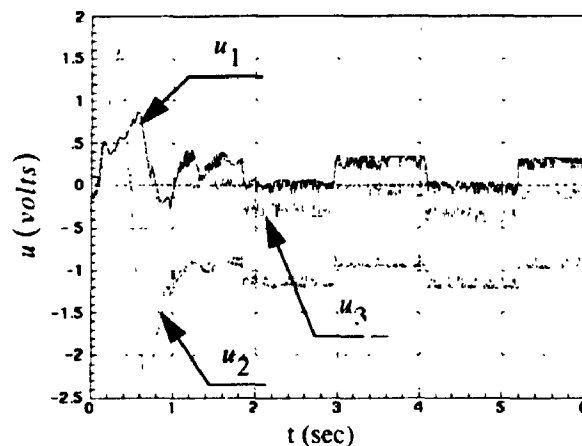
Figure (11.7): Actuators control signals under PD control with high gains.
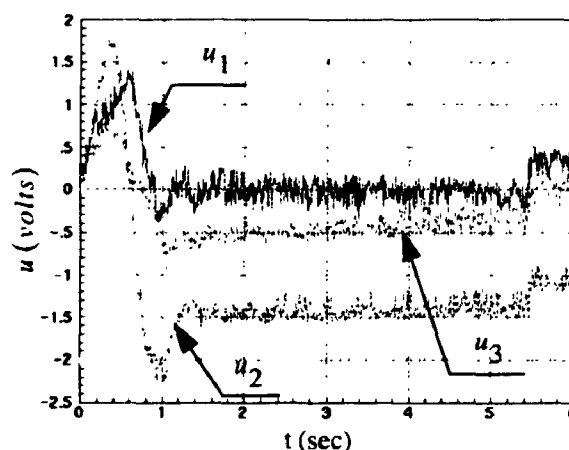
Figure (11.8): Actuators control signals under GS-LQR control.
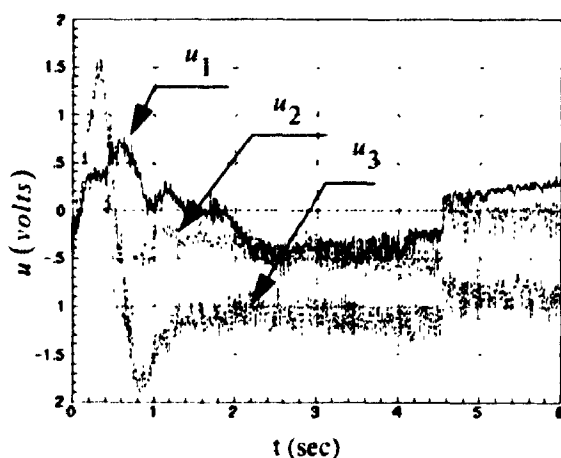
Figure (11.9): Actuator control signals under MRAC control.
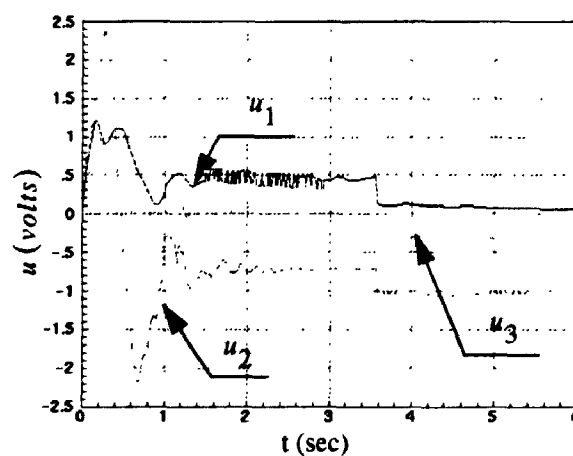
Figure (11.10): Actuator control signals under INV-GS control.

- CHAPTER 12 -

CONCLUSIONS AND RECOMMENDATIONS

This chapter is divided into two sections. The first section is a summary of the research presented in the thesis, conclusions and general observations. The second section includes a discussion of some recommended future research topics that can be considered as an extension to the current thesis work.

12.1 Summary and General Observations

The control of the robot end-point vibration has been achieved. Experimental verification of the simulation was accomplished throughout the thesis context. This would give confidence in implementing the modelling, identification, and control algorithms to a wide range of space and industrial manipulators.

Three issues are significant in this thesis: the 3D configuration of the robot, the modelling and identification, and the sensing techniques as well as adaptive control algorithms.

12.1.1 The Two-Link 3D Configuration

This is one of the most difficult and challenging problems in the area of flexible manipulators since most researchers investigated planar manipulators composed of a single-link or a two-link. The system is complex by nature as it is inherently nonlinear. It

undergoes vibration in the horizontal and vertical directions as well as a complex change of the vibration frequencies with the configuration.

In addition, large space and industrial manipulators which are composed of many links usually have significant flexibility in the first two links. Hence, the techniques developed in the thesis can be applicable to many similar systems. The experimental setup can be easily extended for future research.

12.1.2 Dynamic Modelling and Identification

The modelling and identification approaches used in this thesis are generic. They started by deriving a large order model that includes the flexibility of all the components. This model is used to give a preliminary indication of the system dominant modes, the analysis bandwidth, and the contribution of the joint flexibility to the dominant modes.

Experimental modal analysis was used to identify the system natural frequencies, damping and mode shapes. Based on experimental results, a reduced order model that includes the significant modes is derived to be used for control purposes. Finally, a model tuning technique was used to tune the model to match the experimental results.

Hence, it is shown that the analogy used in the thesis can be applied to any industrial or space manipulator.

12.1.3 Vibration Sensing and Adaptive Control

Accelerometers, attached to the tip of the links, were used to measure the robot vibration. This sensing technique was used due to its availability at relatively low cost and its wide use in the area of vibration. However, the information about the end-point position

can only be estimated using the robot kinematics.

Since the system is highly nonlinear, adaptive controllers and nonlinear state estimators were implemented. Adaptive controllers tend to deal with systems with uncertainty as they incorporate some sort of on-line adaptation. Hence, they have a certain degree of learning ability.

Three controllers were developed for the purpose of vibration control, namely: gain-scheduling regulator (GS-LQR), model reference adaptive control (MRAC) and adaptive inverse dynamics (INV-GS). The first algorithm was not adaptable to the change ol the payload while the other two are adaptable to both payload and configuration changes.

The performance improvement of the adaptive controllers over the PD control algorithm was illustrated in the thesis. It was also shown that the MRAC and INV-GS behave the best. However, MRAC is recommended for most applications rather than INV-GS because it is computationally less expensive.

Two nonlinear state estimators were presented in the thesis, the first was developed specifically for this manipulator in which the flexible states were synthesized using least-squares, while the second was a general observer based on the sliding modes approach.

It should be noted that the controllers and state estimators, derived in the thesis, can be easily applied to any industrial or space manipulator provided that the model order is known and the parameters are partially known.

12.1.4 General Observations

The most general observation that can be drawn from the dissertation is that "Models are made to be used not to be believed". Usually, models are not close to perfect

```
     [0, 0, k3b, 0, -k3e, 0], [0, 0, 0, k3c, 0, 0 ],
     [0, 0, -k3e, 0, k3d, 0], [0, k3e, 0, 0, 0, k3d])$

     delta2 : matrix([ d2x], [ d2y], [ d2z],[ ph2x], [ ph2y], [ ph2z])$

     delta3 : matrix([ d3x], [ d3y],[ d3z],[ ph3x],[ ph3y],[ ph3z])$

  ep1:transpose(delta2).stiff2.delta2+ transpose(delta3).stiff3.delta3 $
  ep2:mass2*ratsimp(transpose(g).posm2) +mass3*ratsimp(transpose(g).posm3) $

  lag: ratsimp(ek1 + ek2 - .5*ep1 - ep2);
  /* ---- Calculating the values for the angles phi's ---- */
   r2e2t: matrix([ 0, -ph2zt, ph2yt],[ ph2zt, 0, -ph2xt],
  [ -ph2yt, ph2xt, 0])$
   ustar: ratsimp(r01.r12.r2e2.re23.def2);
  for i thru 3 do (
        v31[i][1]: ratsimp(diff(ustar[i],ph2xt)),
        v32[i][1]: ratsimp(diff(ustar[i],ph2yt)),
        v33[i][1]: ratsimp(diff(ustar[i],ph2zt)) );

   for i thru 3 do (
        v3[i][1]: v31[i][1],v3[i][2]: v32[i][1],
        v3[i][3]: v33[i][1] );
   k22:matrix([k2c,0,0],[0,k2d,0],[0,0,k2d])$
   k31:matrix([k3a,0,0],[0,k3b,0],[0,0,k3b])$
   k32:matrix([k3c,0,0],[0,k3d,0],[0,0,k3d])$
   k23:matrix([0,0,0],[0,0,k2e],[0,-k2e,0])$
   k33:matrix([0,0,0],[0,0,k3e],[0,-k3e,0])$
   aa:ratsimp(-invert(k32) . transpose(k33))$
   bb2:ratsimp(-invert(k22) . transpose(k23))$
   temp:ratsimp(k31 - k33 . invert(k32) . transpose(k33))$
   t3:ratsimp(r01 . r12 . r2e2 . re23)$
   bb1:ratsimp(invert(k22) . transpose(v3) . t3 . temp)$
   ph3:ratsimp(aa . def3);
   ph2:ratsimp(bb1 . def3)+ratsimp(bb2 . def2);
   ph3t: diff(ph3,t); ph2t: diff(ph2,t);
   kill(k22,k31,k32,k23,k33,aa,bb2,t3,bb1, v3);

   /* substitue and simplify the lagrangian for the analysis */
   lag:subst(ph3t[1][1], ph3xt,lag)$ lag:subst(ph3t[2][1], ph3yt, lag)$
   lag:subst(ph3t[3][1], ph3zt, lag)$ lag:subst(ph2t[1][1][1], ph2xt, lag)$
   lag:subst(ph2t[2][1][1],ph2yt,lag)$lag:subst(ph2t[3][1][1],ph2zt, lag)$

   lag:subst(ph3[1][1], ph3x,lag)$ lag:subst(ph3[2][1], ph3y, lag)$
   lag:subst(ph3[3][1], ph3z, lag)$ lag:subst(ph2[1][1][1], ph2x, lag)$
   lag:subst(ph2[2][1][1], ph2y, lag)$ lag:subst(ph2[3][1][1], ph2z, lag)$

   lag: subst(0, d2xt, lag)$ /* Zeroing the axial deflection */
   lag: subst(0, d3xt, lag)$
   lag: subst(0, d2x, lag)$ lag: subst(0, d3x, lag)$

  dont : [ th1, th2, th3, d2y, d2z, d3y, d3z,
   th1t, th2t, th3t, d2yt, d2zt, d3yt, d3zt,
   th1tt, th2tt, th3tt, d2ytt, d2ztt ,d3ytt, d3ztt];

  zz:[];
  lag:crunch(lag,dont,'tmp)$

  eq1: ratsimp(diff(diff(lag,th1t),t) - diff(lag,th1))$
  eq2: ratsimp(diff(diff(lag,th2t),t) - diff(lag,th2))$
  eq3: ratsimp(diff(diff(lag,th3t),t) - diff(lag,th3))$
  eq4: ratsimp(diff(diff(lag,d2yt),t) - diff(lag,d2y))$
  eq5: ratsimp(diff(diff(lag,d2zt),t) - diff(lag,d2z))$
  eq6: ratsimp(diff(diff(lag,d3yt),t) - diff(lag,d3y))$
  eq7: ratsimp(diff(diff(lag,d3zt),t) - diff(lag,d3z))$

   kill(lag);
   eqns:[eq1,eq2,eq3,eq4,eq5,eq6,eq7]$

  for i thru length(eqns) do (
        eqns[i]:ratsimp(eqns[i]),
        eqns[i]:crunch(eqns[i],dont,'aaa) )$
```

12.2.2 Robot Configurations

The advantages of the manipulator configuration studied in this thesis is that it can be easily extended to study several other configurations. Some of these configurations are discussed below:

Mini-Manipulators

As explained in Section 2.1, the new trend in designing space manipulators is to make a large flexible two-link, 3D, robot to carry out the gross motion. The end of this manipulator will have a mini-manipulator to carry out the fine motion.

Extending the modelling and control algorithms to include mini-manipulators should be investigated. Also, the mini-manipulator motion can be controlled so that the overall vibration may be damped out.

Cooperative Flexible Manipulators

This is a future research area that will have significant applications in space and in industry. However, cooperation of multiple flexible robots is difficult to model successfully since it is a closed-chain flexible mechanism with a lot of interactions. Hence, advanced modelling and control strategies have to be developed for such systems.

Mobile Base Manipulators

For the space-station, the manipulator will be mounted on a mobile base. The interaction of the base motion and the robot motion has to be investigated.

Payload with Dynamics

Usually, large space manipulators are required to handle large payloads (such as

solar-cell arrays). Such payloads can not be modelled as a simple point mass or even as a rigid body mass with inertia. The control strategies have to be modified to cope with the unknown dynamics of the payload.

Force Control of Flexible Links and Flexible Joints Manipulators

While force control of manipulators with flexible joints has been investigated by several authors [51, 80], introducing the links flexibility was not investigated yet. This is attributed to the large size of the problem and difficulties involved in the modelling and control. However, this problem is of particular interest to the industry as a second step that follows end-point trajectory tracking.

12.2.3 Dynamic Modelling

For flexible manipulators, accurate modelling is required in order to appropriately design the control algorithm.

Throughout the thesis, the modelling was done using Lagrangian dynamics where the flexible vibration of the links was imposed over the rigid body motion using either the assumed modes method or the finite element approach. Symbolic algebra packages were used to carry out the derivation.

It is required to develop a comprehensive, user friendly package that can generate the equations of motion for the manipulator given its configuration. The package should have the options of including several modelling issues such as an arbitrary number of modes per link, the payload dynamics, the links centrifugal stiffening, etc. The package should also be capable of generating an optimized code in FORTRAN or C.

A technique that seems to be promising for such development is Kane's approach

[38,39]. Developing such a package will permit studying the influence of many different modelling issues and whether they should be neglected or kept.

Also, parametric study, as a first step toward optimizing the robot design, can be easily done using such a package, e.g. by introducing non-uniform cross-section and strategic locations for the n-link sensors to achieve desired performance.

Another issue that needs to be investigated is the modelling of links that are made of laminated composite structures. Making links out of composite material can increase the payload to weight ratio for the manipulator up to 40%. Also, inducing significant joint and link damping will alter the system dynamics and improve the performance.

Finally, another area that is important for robotic applications is the modelling of harmonic drives and the dynamics of friction present at the robot joints. Although friction was reduced by feedforward compensation, accurate modelling is required to lead eventually to better end-point tracking performance.

12.2.4 Control and Identification Strategies

The identification done in the thesis was based on experimental modal analysis. It is highly desirable to develop unified identification algorithms for any industrial manipulator to identify its inertias, masses, joints and links flexibility, etc. The degree of flexibility of the manipulator has to be addressed.

The area of suppressing robot vibrations using impulse shaping and command filtering is of particular interest since most industrial robots are controlled by PD controllers [50]. It is required to develop unified methods based on these techniques and incorporate them with an existing industrial robot controller to analyze its performance.

An area of particular interest, especially if any advanced control strategy was used, is reconfigurable control. It was found that if an accelerometer is disconnected, while the robot is under the control action, the system goes dangerously unstable and eventually leads to breaking the robot. Hence, it will be hazardous to implement any of these control algorithms unless redundant sensors were used and handled by a reconfigurable controller to predict and compensate for the failure of one of the sensors.

Using redundant sensors will lead to the area of sensor fusion. An advanced estimation algorithm is required to handle multiple sensors.

Implementing and comparing different controllers is an open-ended research topic for control, in general.

Robust control is easier to implement than adaptive control since it is a fixed controller structure. H-Infinity [111] was used previously in the area of flexible structure control but was never implemented on flexible manipulators.

Another approach is to enhance the control algorithms using Fuzzy-Logic [22] or Neural Networks. Also, the use of neural networks to identify flexible manipulators is an interesting topic of research.

An important robotics control topic is torque control for the robot joints. Ideally, the torque is proportional to the motor current. In fact, compliance and nonlinearties (such as friction and backlash) cause significant deviations from the desired case. Although considerable research was done in this area, there is not any commercial torque controller that is being used in industry. Hence, this can be considered as a future research topic.

As final research topic, the experimental application of the developed modelling and control strategies on an industrial robot is worthily of consideration.

- APPENDIX A -

ROBOT CONSTRUCTION AND SPECIFICATIONS

This appendix contains a detailed description of the manipulator hardware. It is divided into two sections to cover the manipulator construction, and the specifications for the actuators, transmission, and brakes.

A.1 The Manipulator Construction

The robot, as shown in Figure (A.1), consists of four subassemblies, namely: the base, the shoulder, the elbow, and the end-effector. For the thesis work, the end-effector was just a cylindrical mass.

The robot is powered by DC servo motors, shown in Figure (A.2). The motor chosen for the elbow joint was CMC 3505. The motors for the shoulder and base joints were CMC 3515. The specifications for the motors are shown in Section A.2. Note that the motors are over-designed for future implementation of larger and heavier links.

The motors are driven by a PWM Amplifier in the current mode. The amplifier types are Galil ESA-10/75 with switching frequency of 20 KHz. The current limits are 12 Amps continuous, 40 Amps peak which are sufficient for most robotic applications.

Harmonic drives with a ratio 100:1 are attached to each motor to provide maximum torque at minimum joint weight. This was particularly needed for the elbow joint. The harmonic drive components are shown in Figure (A.3). Note that the chosen harmonic drive
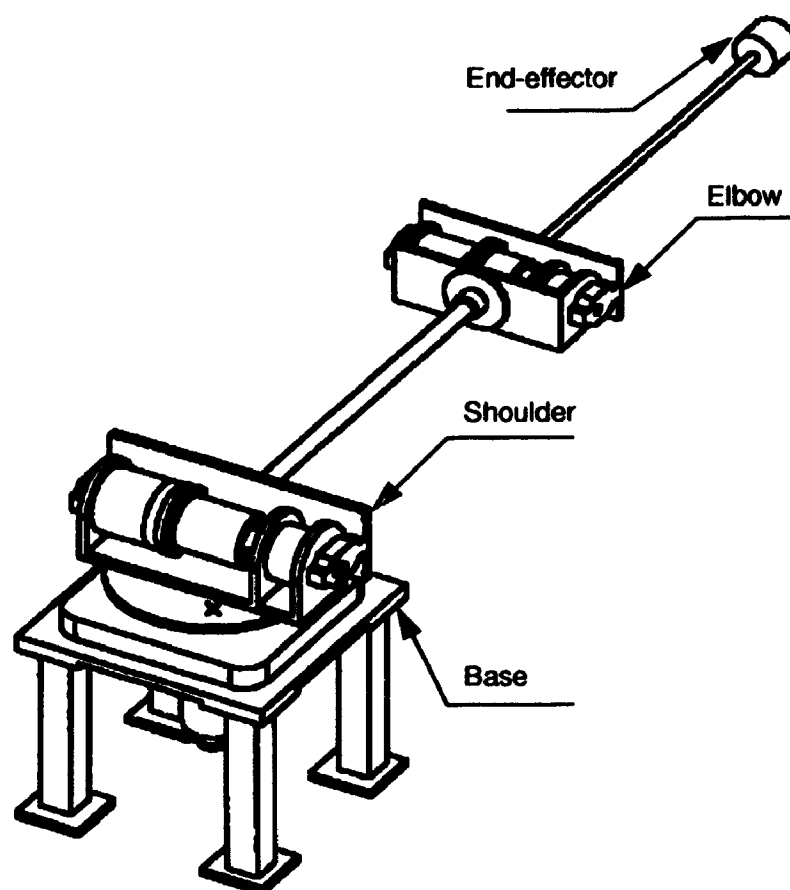
Figure (A.1): Construction drawing of the robot (fully extended).

3500 SERIES Diameter 3.38"

3.38"
SQUARE

L
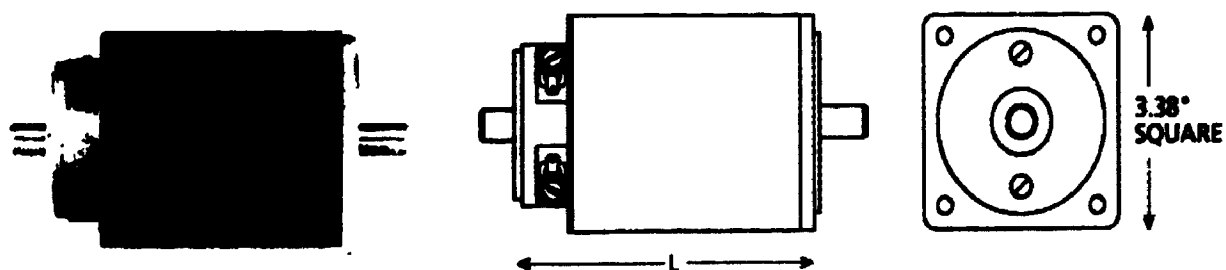
Figure (A.2): Construction drawing of a DC motor.

(Courtesy of CMC Inc.)

HDC CUP COMPONENT GEAR SET

Flexspline
An elliptical,
nonrigid,
external gear

Circular Spline
A round, rigid,
internal gear

Wave Generator
An elliptical
ball bearing assembly
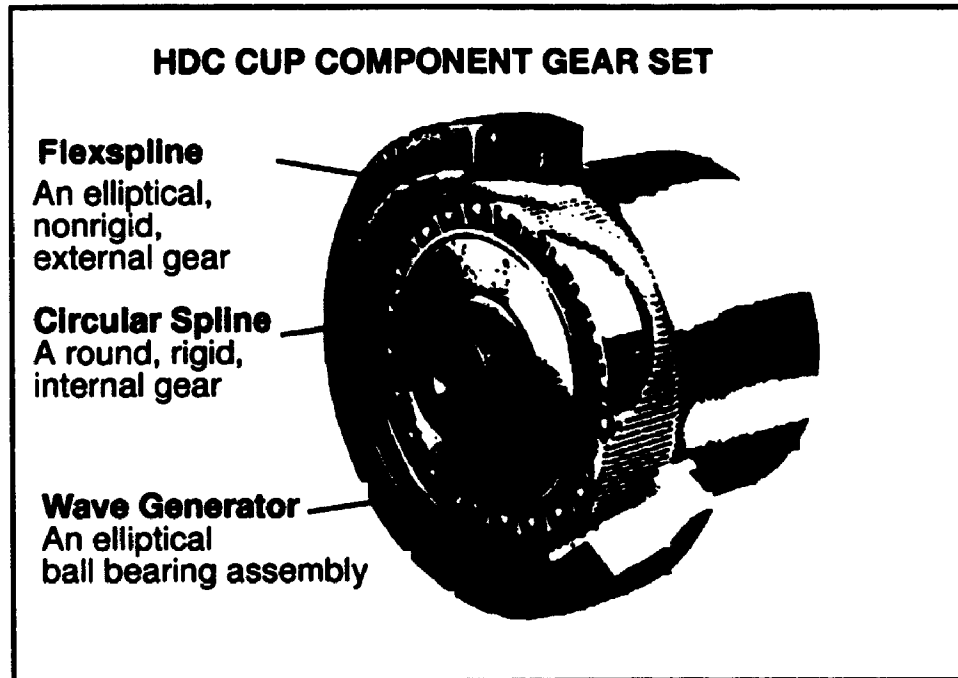
Figure (A.3): The harmonic drive components.

(Courtesy of Harmonic Drive Inc.)

No.	Part No.	DESCRIPTION
1	1M-01	Output shaft
2	1M-02	Retaining ring
3	1M-03	Retaining ring
4	1M-04	Bearing
5	1M-05	Housing
6	1M-06	Harmonic drive
7	1M-07	Cover
8	1M-08	Bush
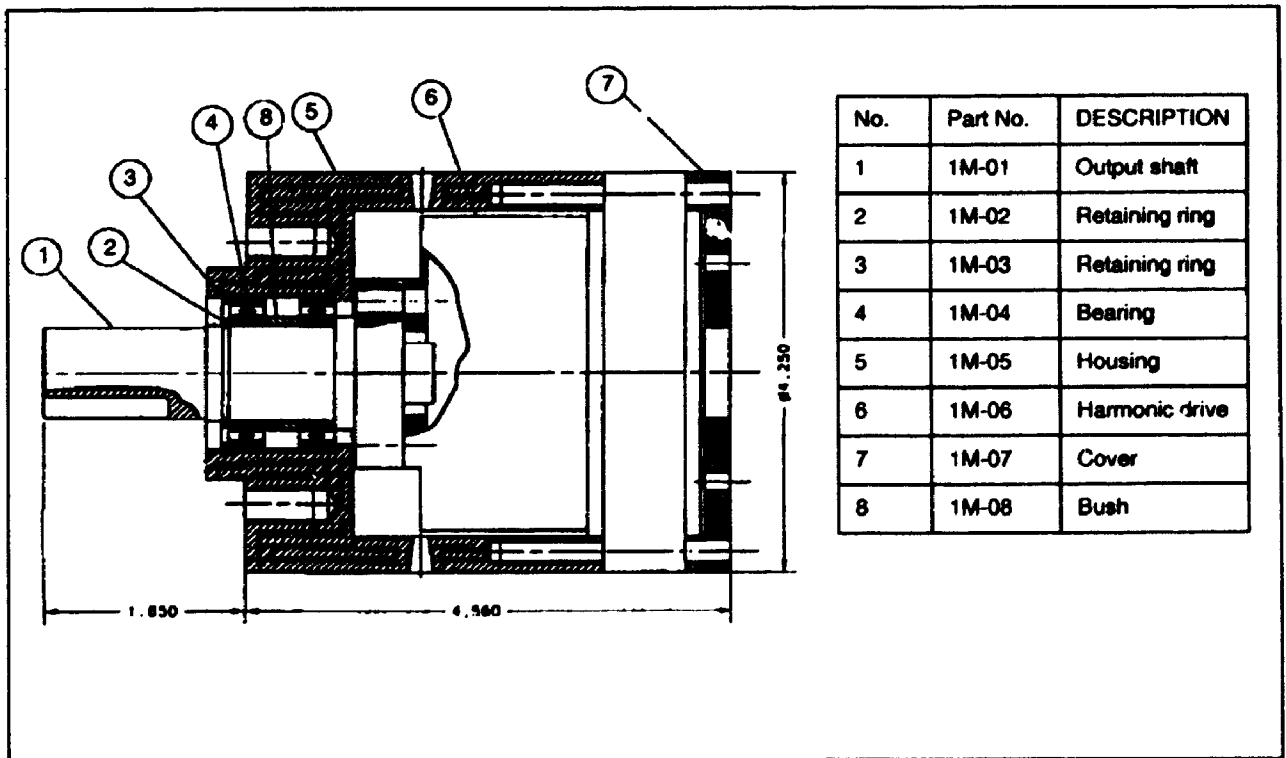
Figure (A.4): Construction drawing of the harmonic drive gearbox.

unit for the elbow was HDC-5C, whereas for the shoulder and base were HDC-1M. The gearbox that contains the harmonic drive is made in-house. The construction drawing for the gearbox assembly is shown in Figure (A.4).

The joints design includes the base, shoulder and elbow joints. The design work was carried out on the solid modelling module of SDRC I-DEAS Ver. 5. The working drawings for all the components, carried out on the drafting module of the package, are available in [105].

The base construction drawing is shown in Figure (A.5) and the exploded assembly is shown in Figure (A.6). Note that the base joint does not have a brake since it is acting in the horizontal direction.

The construction drawing for the shoulder joint is shown in Figure (A.7). The assembly drawing is shown in Figure (A.8). The elbow joint is the same as the shoulder joint but smaller in size.

The motors for both joints had to be extended from the back to allow the mounting of fail-safe brakes. For the elbow joint, the brake type was MFSB 7-6, while for the shoulder the brake type was MFSB 26-8 (from Electroid Inc.). This particular design for the joints was chosen because of its simplicity and low cost.

The links were made of Aluminum tubes (Al 6061-T6) following the pipe standard specifications. The first link was Pipe std. 0.75" Dia Sch. 40 and 0.75 m length, while the second link was Pipe std 0.25" Dia Sch. 80 and 0.75 m length.

It should be noted that the robot design is modular to allow various links configurations. The joints can also be easily modified to include springs to simulate the joints flexibility.

No.	Part No.	DESCRIPTION
1		1M Harmonic Drive
2	B01	Supporting plate
3	B02	Outer retainer
4	B03	Outer retainer (top)
5	B04	Bearing
6	B05	Inner retainer
7	B06	Rotating plate
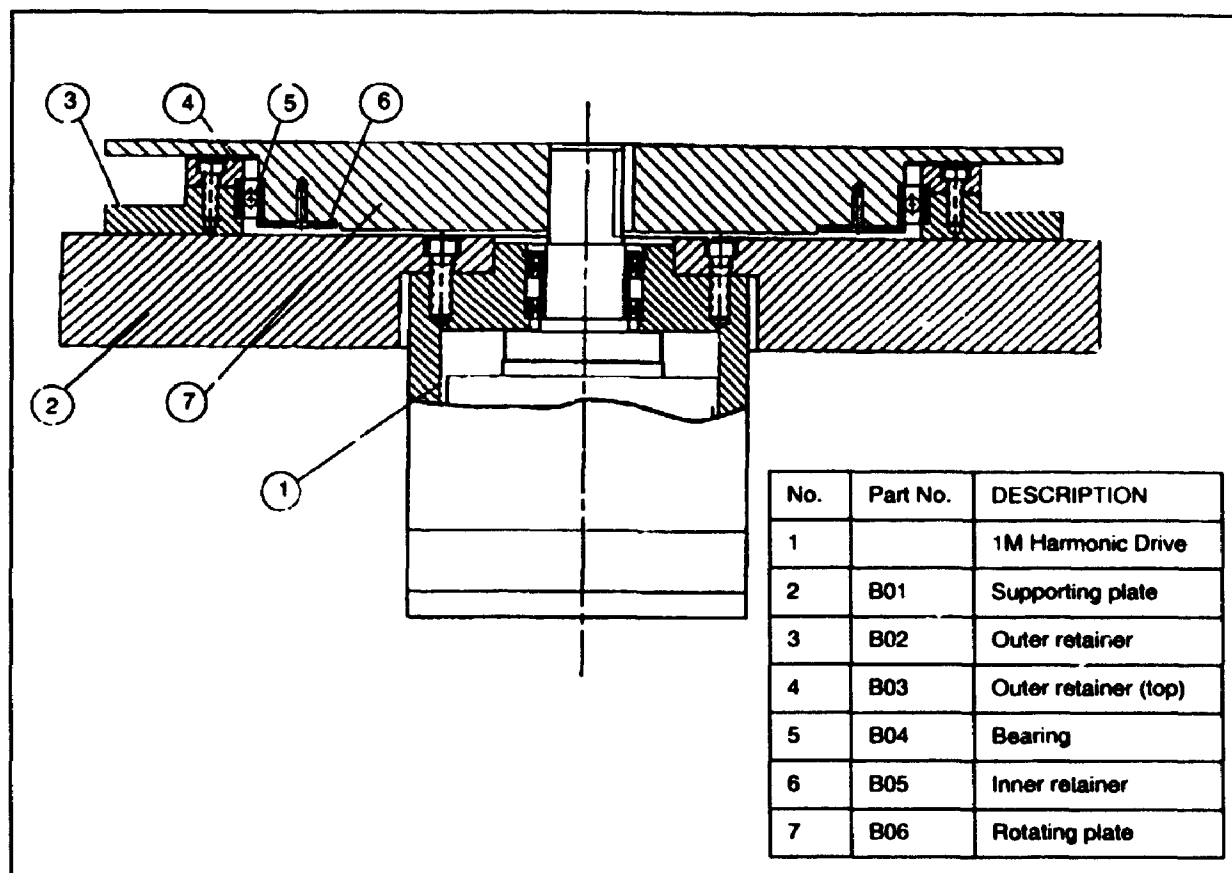
Figure (A.5): Construction drawing of the base joint.
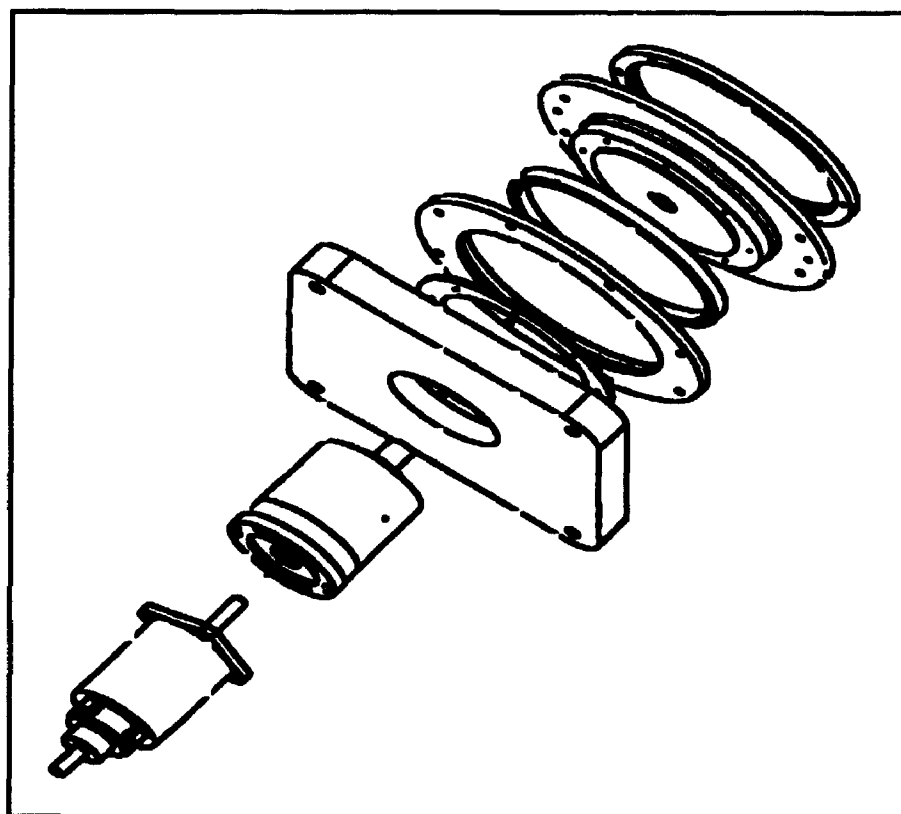
Figure (A.6): Exploded view of the base joint assembly.

No.	Part No.	DESCRIPTION
1	S01	Supporting plate
2	S02	Left end
3	S03	Right end
4	S04	Brake support
5	S05	Drive crank
6	S06	Dummy crank
7	S07	Dummy shaft
8		Fail safe brake
9		Motor 3515
10		Harmonic drive
11		Flange
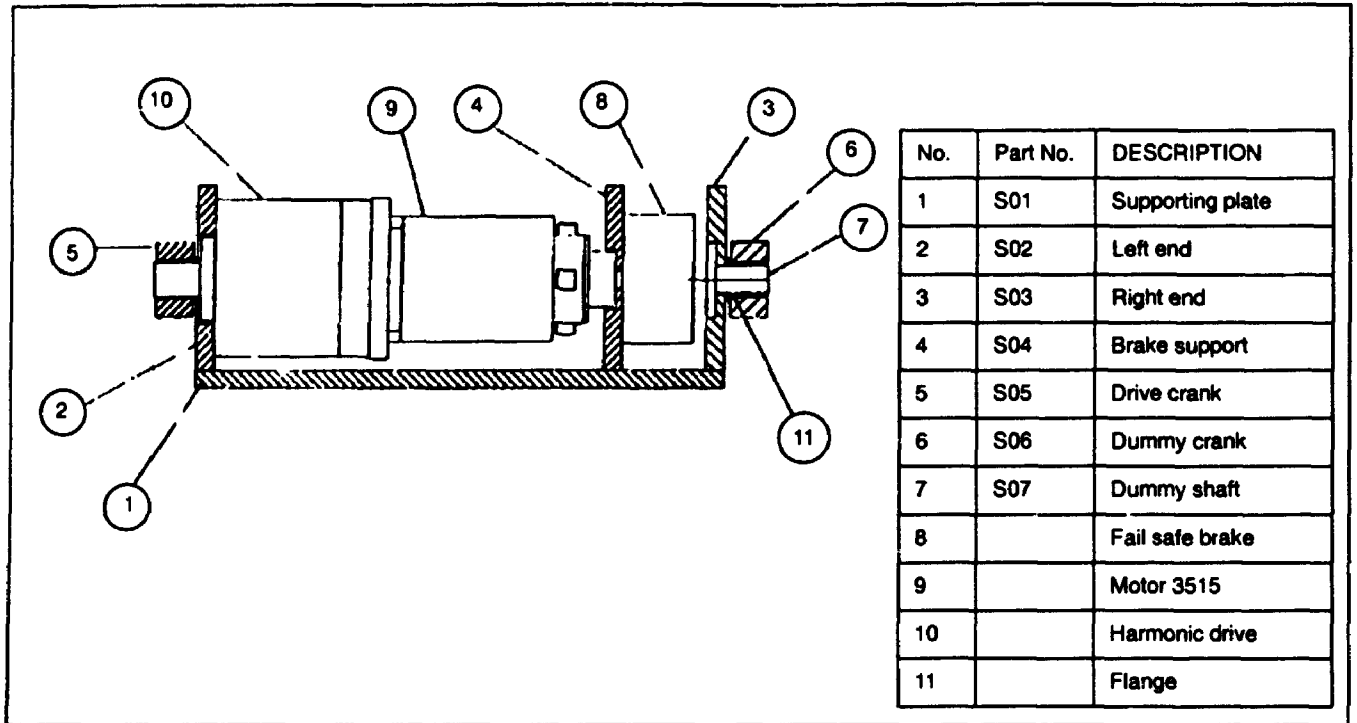
Figure (A.7): Construction drawing of the shoulder joint.
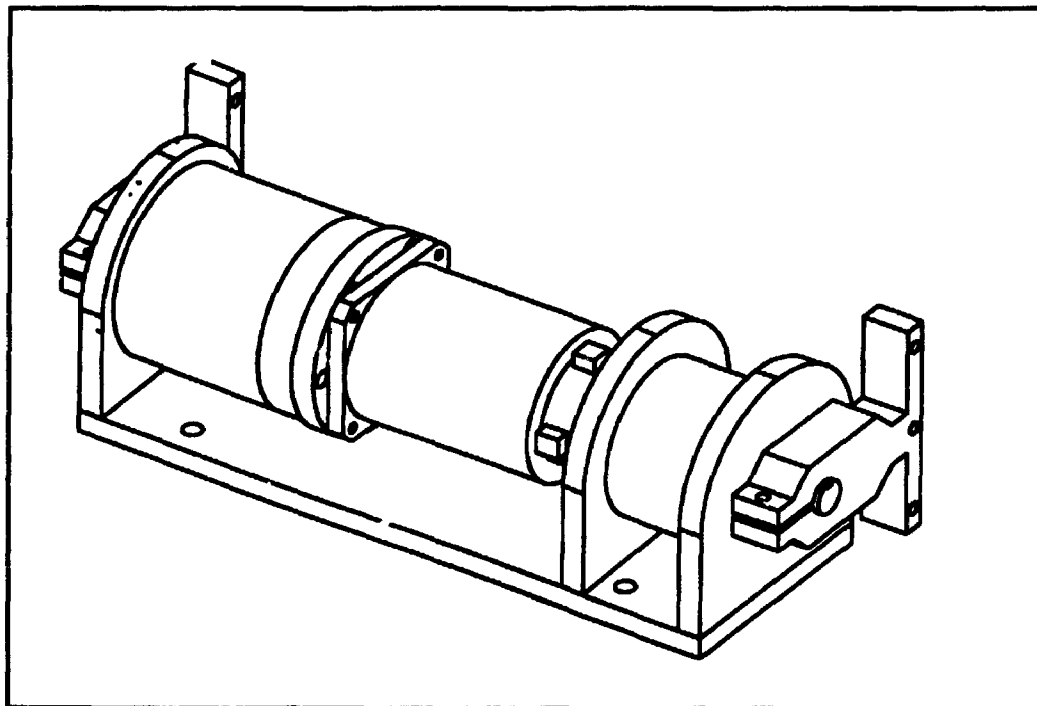
Figure (A.8): Shoulder joint assembly.

A.2 Actuators and Drives Specifications

The specifications for the motors, taken from the manufacturers catalogues, are shown in Table (A.1), while the specifications for the drives and fail-safe brakes are shown in Table (A.2) and Table (A.3), respectively.

Table A.1: Motors specifications.

Motor/Spec.	CMC 3505-G	CMC 3515-C
Cont. Torque (Oz-in)	42	103
Peak Torque (Oz-in)	350	900
Torq. Sens. (Oz-in/Amp)	16	15.9
Arm. Resistance (Ohms)	2.2	0.67
Back EMF (Volts/KRPM)	11.8	11.8
Inertia (Oz-In-sec^2)	0.0067	0.0135
Mech. Time (ms)	8.1	3.9
Elect. Time (ms)	1.5	1.7
Weight (lbs)	3.2	5.0

Table A.2: Harmonic drives specifications.

Drive/Spec.	HDC-5C	HDC-1M
Max. Output Torque (lb-in)	1240	2640
Out. Torq. at 3500 RPM (lb-in)	490	990
No Load Starting Torq. (Oz-in)	5.0	11.0
Weight (lbs)	1.2	2.6

Table A.3: Brakes specifications.

Brake/Spec.	MFSB-7-6	MFSB-26-8
Rated Static Torque (lb-in)	7.0	30.0
Weight (lbs)	0.6	1.4
Current at 90 VDC (Amp)	0.06	0.08

- APPENDIX B -

SELECTED SOFTWARE LISTINGS

This appendix contains some of the important software listings that were used in the process of modelling the manipulator and designing the controllers.

B.1 Modelling Routines

The derivation for the dynamic models as well as the inverse dynamic models for the robot was carried out on MACSYMA. Presented in this section are the listings for the derivation of the large and reduced order models.

B.1.1 The Large Order Model

As explained in Chapter 4, the large order model was derived using Lagrangian dynamics where the elastic deflection was represented using the assumed modes method. The listing for the MACSYMA routine is as follows:

```
/* Macsyma derivation for the equations of motion usinr the assumed modes method
for a two link manipulator in 3-D using Lagrange's method. The shape functions
used are the CANT developed using curve-fitting. Assuming that the ass. modes are
the same for the Y,Z directions. The torsion is present, axial deflection is
neglected */

/* ------ No joint flexibility present in the analysis ------ */
/* dm1, dm2 are the links masses per unit length,
 len1, len2 are the links lengths,
 mass1, mass2 are the tip masses of the links,
 dim1, dim2, dim3 are the inertia of the motors,
 ei is e*i product of the link, gi is the same.
 g is the gravity in the -y direction in frame 0 */

batchload("integ.mac");
batchload("util.mac"); /* simplification routines */
derivabbrev:true;
```

```
matchdeclare(any,true)$
tellsimp(sin(any)^2,1-cos(any)^2)$
linel:100;
writefile("three_ass.deft.dump");

/*q2y and q2z are the link def. p2x is the torsional deflection */

depends([th1,th2,th3,q2y,q2z,p2x,q3y,q3z,p3x],t);

th1t: diff(th1,t)$ th2t: diff(th2,t)$ th3t: diff(th3,t)$
q2yt: diff(q2y,t)$ q2zt: diff(q2z,t)$ p2xt: diff(p2x,t)$
q3yt: diff(q3y,t)$ q3zt: diff(q3z,t)$ p3xt: diff(p3x,t)$

th1tt: diff(th1,t,2)$ th2tt: diff(th2,t,2)$ th3tt: diff(th3,t,2)$
q2ytt: diff(q2y,t,2)$ q2ztt: diff(q2z,t,2)$ p2xtt: diff(p2x,t,2)$
q3ytt: diff(q3y,t,2)$ q3ztt: diff(q3z,t,2)$ p3xtt: diff(p3x,t,2)$

/* ---- the f2 is the displacement, r is the rotation, p is the torsion */

f2: 3*(x2/len2)^2/2 - (x2/len2)^3/2;
f2r: ratsimp(diff(f2,x2))$
f2p: x2/len2;

f3: 3*(x3/len3)^2/2 - (x3/len3)^3/2;
f3r: ratsimp(diff(f3,x3))$
f3p: x3/len3;

/* deflection at the tips of both links */
 d2y: ev(q2y*f2, x2=len2)$ d2z: ev(q2z*f2, x2=len2)$
 ph2y: ev(q2z*f2r, x2=len2)$ ph2z: ev(q2y*f2r, x2=len2)$

 d3y: ev(q3y*f3, x3=len3)$ d3z. ev(q3z*f3, x3=len3)$
 ph3y: ev(q3z*f3r, x3=len3)$ ph3z: ev(q3y*f3r, x3=len3)$

/*-----------------------rotation matrices for the coord. systems */
 r01 : matrix([ cos(th1), -sin(th1), 0],
 [ sin(th1), cos(th1) , 0], [ 0, 0, 1])$

 r12: matrix([ cos(th2), -sin(th2), 0],
 [ 0, 0, -1],[ sin(th2), cos(th2), 0])$

 r2e2: matrix([ 1, -ph2z, ph2y],
 [ ph2z, 1, -p2x], [ -ph2y, p2x, 1])$

re23: matrix([ cos(th3), -sin(th3), 0],
 [ sin(th3), cos(th3), 0 ], [0, 0, 1])$

 r3e3: matrix([ 1, -ph3z, ph3y],
 [ ph3z, 1, -p3x], [ -ph3y, p3x, 1])$

 def2: matrix([ len2], [ d2y], [ d2z])$

 def3: matrix([ len3], [ d3y], [ d3z])$

 g: matrix([ 0], [0], [-g0])$

/*------ rotation matrices for the x's postion on the links */

 def2x: matrix([ x2], [ q2y*f2], [ q2z*f2])$

 def3x: matrix([ x3],
 [ expand(q3y*f3)], [ expand(q3z*f3)])$

 r2e2x: matrix([ 1, -q2y*f2r, q2z*f2r],
 [ q2y*f2r, 1, -p2x*f2p], [ -q2z*f2r, p2x*f2p, 1])$

 r3e3x: matrix([ 1, -q3y*f3r, q3z*f3r],
 [ q3y*f3r, 1, -p3x*f3p], [ -q3z*f3r, p3x*f3p, 1])$

/*------------------------------- evaluating the positions */
posm2: expand(r01.r12.r2e2.def2);
posm3: expand( posm2 + expand(r01.r12.r2e2.re23.r3e3.def3));
```

```
 pos2x: expand(r01.r12.r2e2x.def2x)$
 pos3x: expand( posm2 + expand(r01.r12.r2e2.re23.r3e3x.def3x))$

/*------------------Evaluating the gravitational Pot. energy */
 ep1:mass2*expand(transpose(g).posm3)+mass3*expand(transpose(g).posm2)$
 ep2: mass12/len2*expand(integ(expand(transpose(g).pos2x),x2,0,len2))$
 ep3: mass13/len3*expand(integ(expand(transpose(g).pos3x),x3,0,len3))$

/*-------------------Evaluating the elastic potential energy */
 y2_pp: diff(q2y*f2,x2,2)$
 z2_pp: diff(q2z*f2,x2,2)$
 p2x_p: diff(p2x*f2p,x2)$

 y3_pp: diff(q3y*f3,x3,2)$
 z3_pp: diff(q3z*f3,x3,2)$
 p3x_p: diff(p3x*f3p,x3)$

 ep4: integ(expand(ei2*y2_pp^2 + ei2*z2_pp^2 + gj2*p2x_p^2),x2,0,len2)$
 ep5: integ(expand(ei3*y3_pp^2 + ei3*z3_pp^2 + gj3*p3x_p^2),x3,0,len3)$
 ep: expand(ep1 + ep2 + ep3 + .5*ep4 + .5*ep5)$

 kill(ep1,ep2,ep3,ep4,ep5,y2_pp,z2_pp,p2x_p,y3_pp,z3_pp);
 kill(r01,r12,r2e2,re23,def2,def3,g,r2e2x,r3e3x);

/*----------------------------------- evaluating the velocities */
 vm2: diff(posm2,t)$ vm3: diff(posm3,t)$
 v2x: diff(pos2x,t)$ v3x: expand(diff(pos3x,t))$
 kill(posm2, posm3, pos2x, pos3x);

/*- -------------------------------obtaining the velocity squared */
 vm2_2: expand(transpose(vm2).vm2)$ vm3_2: expand(transpose(vm3).vm3)$
 v2x_2: expand(transpose(v2x).v2x)$ v3x_2: expand(transpose(v3x).v3x)$

/*--------------------------Calculating the KE for links and masses */
 ek1: ratsimp(.5*im1*th1t^2 +.5*mass2*vm2_2 + .5*im2*th2t^2 + .5*mass3*
 vm3_2 + .5*im3*th3t^2);
 ek12: mass12/len2*integ(v2x_2,x2,0,len2)/2;
 ek13: mass13/len3*integ(v3x_2,x3,0,len3)/2;
 ek: expand(ek1 + ek12 + ek13)$


/*-----------------------------------------Analyzing the lagrangian */
 lag: expand(ek - ep);

 kill(ek1,ek12,ek13, ek, ep);
kill(vm2,vm3,v2x,v3x,vm2_2, vm3_2, v2x_2, v3x_2);
 dont:[ th1, th2, th3, q2y, q2z, p2x, q3y, q3z, p3x,
 th1t, th2t, th3t, q2yt, q2zt, p2xt, q3yt, q3zt,p3xt,
 th1tt, th2tt, th3tt, q2ytt, q2ztt, p2xtt, q3ytt, q3ztt, p3xtt];

zz: [];
lag:crunch(lag,dont,'tmp)$ /* extracting time-independent terms */

eq1: ratsimp(expand(diff(diff(lag,th1t),t) - diff(lag,th1)))$
eq2: ratsimp(expand(diff(diff(lag,th2t),t) - diff(lag,th2)))$
eq3: ratsimp(expand(diff(diff(lag,th3t),t) - diff(lag,th3)))$
eq4: ratsimp(expand(diff(diff(lag,q2yt),t) - diff(lag,q2y)))$
eq5: ratsimp(expand(diff(diff(lag,q2zt),t) - diff(lag,q2z)))$
eq6: ratsimp(expand(diff(diff(lag,p2xt),t) - diff(lag,p2x)))$
eq7: ratsimp(expand(diff(diff(lag,q3yt),t) - diff(lag,q3y)))$
eq8: ratsimp(expand(diff(diff(lag,q3zt),t) - diff(lag,q3z)))$
eq9: ratsimp(expand(diff(diff(lag,p3xt),t) - diff(lag,p3x)))$

 kill(lag)$

 eqns:[eq1,eq2,eq3,eq4,eq5,eq6,eq7,eq8,eq9]$

for i thru length(eqns) do (
 eqns[i]:ratsimp(eqns[i]),
 eqns[i]:crunch(eqns[i],dont,'aaa) )$
```

```
kill(eq1, eq2, eq3, eq4, eq5, eq6,eq7.eq8,eq9);
map(fortran,zz); /* coding the time-independent terms */

kill(zz); zz:[];

dont:[th1tt, th2tt, th3tt, q2ytt, q2ztt, p2xtt, q3ytt, q3ztt, p3xtt];
eqns: crunch(eqns,dont,'zd); /* collecting the velocity dep. terms */

a:zeromatrix(9,9)$ /* start the extraction of the mass matrix */

for i thru 9 do (
 tmp:expand((eqns[i])),
a[i][1]: coeff(tmp,th1tt), a[i][2]: coeff(tmp,th2tt),
a[i][3]: coeff(tmp,th3tt), a[i][4]: coeff(tmp,q2ytt),
a[i][5]: coeff(tmp,q2ztt), a[i][6]: coeff(tmp,p2xtt),
a[i][7]: coeff(tmp,q3ytt), a[i][8]: coeff(tmp,q3ztt),
a[i][9]: coeff(tmp,p3xtt) );

fortran(a);/* Generating the code for the inertial matrix */

eqns: subst(0,th1tt,eqns)$ eqns:subst(0, th2tt, eqns)$
eqns: subst(0, th3tt, eqns)$ eqns: subst(0, q2ytt, eqns)$
eqns: subst(0, q2ztt, eqns)$ eqns: subst(0, p2xtt, eqns)$
eqns: subst(0, q3ytt, eqns)$ eqns: subst(0, q3ztt, eqns)$
eqns: subst(0, p3xtt, eqns)$

for i thru 9 do ( /* generating the state equations */
 eqns[i]:ratsimp(-1*eqns[i]) )$

eqns: subst(x[1],th1t,eqns)$ eqns: subst(x[2],th2t,eqns)$
eqns: subst(x[3],th3t,eqns)$ eqns: subst(x[4],q2yt,eqns)$
eqns: subst(x[5],q2zt,eqns)$ eqns: subst(x[6],p2xt,eqns)$
eqns: subst(x[7],q3yt,eqns)$ eqns: subst(x[8],q3zt,eqns)$
eqns: subst(x[9],p3xt,eqns)$ eqns: subst(x[10], th1, eqns)$
eqns: subst(x[11], th2, eqns)$ eqns: subst(x[12], th3, eqns)$
eqns: subst(x[13], q2y, eqns)$ eqns: subst(x[14], q2z, eqns)$
eqns: subst(x[15], p2x, eqns)$ eqns: subst(x[16], q3y, eqns)$
eqns: subst(x[17], q3z, eqns)$ eqns: subst(x[18], p3x, eqns)$

/* generating the code for the state equations */
fortran(eqns[1]); fortran(eqns[2]); fortran(eqns[3]);
fortran(eqns[4]); fortran(eqns[5]); fortran(eqns[6]);
fortran(eqns[7]); fortran(eqns[8]); fortran(eqns[9]);

zz: subst(x[1],th1t,zz)$ zz: subst(x[2],th2t,zz)$
zz: subst(x[3],th3t,zz)$ zz: subst(x[4],q2yt,zz)$
zz: subst(x[5],q2zt,zz)$ zz: subst(x[6],p2xt,zz)$
zz: subst(x[7],q3yt,zz)$ zz: subst(x[8],q3zt,zz)$
zz: subst(x[9],p3xt,zz)$ zz: subst(x[10], th1, zz)$
zz: subst(x[11], th2, zz)$ zz: subst(x[12], th3, zz)$
zz: subst(x[13], q2y, zz)$ zz: subst(x[14], q2z, zz)$
zz: subst(x[15], p2x, zz)$ zz: subst(x[16], q3y, zz)$
zz: subst(x[17], q3z, zz)$ zz: subst(x[18], p3x, zz)$

map(fortran,zz); /* generating the fortran code for the velocity
                    and position dependent terms */
closefile(); quit();
```

B.1.2 The Reduced Order Model

As explained in Chapter 5, a reduced order model was derived using Lagrangian

dynamics where the elastic deflection was represented using the finite element

approximation. The listing for MACSYMA routine is as follows:

```
/* Macsyma input file for TWO LINK 3D manipulator. The elastic deflection is
represented by the FE method. Let zero condition be horizontal position, gravity
points down */

/* mass1, mass2, are the lumped masses,
 len1, len2 , and lenc are link lengths
 ei is the e*i product for the flexible beam crossection
 g is gravity, points in the -y direction in frame 0 */

batchload("util.mac");
writefile("three_lk.app.dump");
derivabbrev:true;

depends([th1,th2,th3,d2x,d2y,d2z,d3x,d3y,d3z,ph2x,ph2y,ph2z,ph3x,ph3y,ph3z],t);

th1t: diff(th1,t)$ th2t: diff(th2,t)$ th3t: diff(th3,t)$
d2xt: diff(d2x,t)$ d2yt: diff(d2y,t)$ d2zt: diff(d2z,t)$
d3xt: diff(d3x,t)$ d3yt: diff(d3y,t)$ d3zt: diff(d3z,t)$
ph2xt: diff(ph2x,t)$ ph2yt: diff(ph2y,t)$ ph2zt: diff(ph2z,t)$
ph3xt: diff(ph3x,t)$ ph3yt: diff(ph3y,t)$ ph3zt: diff(ph3z,t)$

th1tt: diff(th1,t,2)$ th2tt: diff(th2,t,2)$ th3tt: diff(th3,t,2)$
d2xtt: diff(d2x,t,2)$ d2ytt: diff(d2y,t,2)$ d2ztt: diff(d2z,t,2)$
d3xtt: diff(d3x,t,2)$ d3ytt: diff(d3y,t,2)$ d3ztt: diff(d3z,t,2)$
ph2xtt: diff(ph2xt,t,2)$ ph2ytt: diff(ph2yt,t,2)$ph2ztt: diff(ph2zt,t,2)$
ph3xtt: diff(ph3xt,t,2)$ ph3ytt: diff(ph3yt,t,2)$ph3ztt: diff(ph3zt,t,2)$

/*--------------------------rotation matrices for the coord. systems */
 r01 : matrix([ cos(th1), -sin(th1), 0],
 [ sin(th1), cos(th1) , 0], [ 0, 0, 1])$

 r12: matrix([ cos(th2), -sin(th2), 0],
 [ 0, 0, -1], [ sin(th2), cos(th2), 0])$

 r2e2: matrix([ 1, -ph2z, ph2y],
 [ ph2z, 1, -ph2x], [ -ph2y, ph2x, 1])$

 re23: matrix([ cos(th3), -sin(th3), 0],
 [ sin(th3), cos(th3), 0 ], [0, 0, 1])$

 def2: matrix([ len2 + d2x],
 [ d2y], [ d2z])$

 rot2: matrix([ ph2x],
 [ ph2y], [ ph2z])$

 def3: matrix([ len3 + d3x],
 [ d3y], [ d3z])$

 rot3: matrix([ ph3x],
 [ ph3y], [ ph3z])$

g: matrix([ 0],
 [0], [-g0])$

posm2: ratsimp(r01.r12.def2);
posm3: ratsimp(posm2 + ratsimp(r01.r12.r2e2.re23.def3));
posmc: ratsimp(r01.r12.defc);

vm2: diff(posm2,t)$ vm3: diff(posm3,t)$ vmc: diff(posmc,t)$
vm2_2: ratsimp(transpose(vm2).vm2)$
vm3_2: ratsimp(transpose(vm3).vm3)$
vmc_2: ratsimp(transpose(vmc).vmc)$

ek1:ratsimp(.5*im1*th1t^2+.5*mass2*vm2_2 +.5*im2*th2t^2)$
ek2: ratsimp(.5*mass3*vm3_2 + .5*im3*th3t^2)$

 stiff2: matrix([k2a, 0, 0, 0, 0, 0], [0, k2b, 0 , 0, 0, k2e],
 [0, 0, k2b, 0, -k2e, 0], [0, 0, 0, k2c, 0, 0 ],
 [0, 0, -k2e, 0, k2d, 0], [0, k2e, 0, 0, 0, k2d])$

 stiff3: matrix([k3a, 0, 0, 0, 0, 0], [0, k3b, 0 , 0, 0, k3e],
```

```
[0, 0, k3b, 0, -k3e, 0], [0, 0, 0, k3c, 0, 0 ],
[0, 0, -k3e, 0, k3d, 0], [0, k3e, 0, 0, 0, k3d])$

delta2 : matrix([ d2x], [ d2y], [ d2z],[ ph2x], [ ph2y], [ ph2z])$

delta3 : matrix([ d3x], [ d3y],[ d3z],[ ph3x],[ ph3y],[ ph3z])$

ep1:transpose(delta2).stiff2.delta2+ transpose(delta3).stiff3.delta3 $
ep2:mass2*ratsimp(transpose(g).posm2) +mass3*ratsimp(transpose(g).posm3) $

lag: ratsimp(ek1 + ek2 - .5*ep1 - ep2);
/* ---- Calculating the values for the angles phi's ---- */
 r2e2t: matrix([ 0, -ph2zt, ph2yt],[ ph2zt, 0, -ph2xt],
[ -ph2yt, ph2xt, 0])$
 ustar: ratsimp(r01.r12.r2e2.re23.def2);
for i thru 3 do (
        v31[i][1]: ratsimp(diff(ustar[i],ph2xt)),
        v32[i][1]: ratsimp(diff(ustar[i],ph2yt)),
        v33[i][1]: ratsimp(diff(ustar[i],ph2zt)) );

 for i thru 3 do (
        v3[i][1]: v31[i][1],v3[i][2]: v32[i][1],
        v3[i][3]: v33[i][1] );
 k22:matrix([k2c,0,0],[0,k2d,0],[0,0,k2d])$
 k31:matrix([k3a,0,0],[0,k3b,0],[0,0,k3b])$
 k32:matrix([k3c,0,0],[0,k3d,0],[0,0,k3d])$
 k23:matrix([0,0,0],[0,0,k2e],[0,-k2e,0])$
 k33:matrix([0,0,0],[0,0,k3e],[0,-k3e,0])$
 aa:ratsimp(-invert(k32) . transpose(k33))$
 bb2:ratsimp(-invert(k22) . transpose(k23))$
 temp:ratsimp(k31 - k33 . invert(k32) . transpose(k33))$
 t3:ratsimp(r01 . r12 . r2e2 . re23)$
 bb1:ratsimp(invert(k22) . transpose(v3) . t3 . temp)$
 ph3:ratsimp(aa . def3);
 ph2:ratsimp(bb1 . def3)+ratsimp(bb2 . def2);
 ph3t: diff(ph3,t); ph2t: diff(ph2,t);
 kill(k22,k31,k32,k23,k33,aa,bb2,t3,bb1, v3);

 /* substitue and simplify the lagrangian for the analysis */
 lag:subst(ph3t[1][1], ph3xt,lag)$ lag:subst(ph3t[2][1], ph3yt, lag)$
 lag:subst(ph3t[3][1], ph3zt, lag)$ lag:subst(ph2t[1][1][1], ph2xt, lag)$
 lag:subst(ph2t[2][1][1],ph2yt,lag)$lag:subst(ph2t[3][1][1],ph2zt, lag)$

 lag:subst(ph3[1][1], ph3x,lag)$ lag:subst(ph3[2][1], ph3y, lag)$
 lag:subst(ph3[3][1], ph3z, lag)$ lag:subst(ph2[1][1][1], ph2x, lag)$
 lag:subst(ph2[2][1][1], ph2y, lag)$ lag:subst(ph2[3][1][1], ph2z, lag)$

 lag: subst(0, d2xt, lag)$ /* Zeroing the axial deflection */
 lag: subst(0, d3xt, lag)$
 lag: subst(0, d2x, lag)$ lag: subst(0, d3x, lag)$

 dont : [ th1, th2, th3, d2y, d2z, d3y, d3z,
  th1t, th2t, th3t, d2yt, d2zt, d3yt, d3zt,
  th1tt, th2tt, th3tt, d2ytt, d2ztt ,d3ytt, d3ztt];

 zz:[];
 lag:crunch(lag,dont,'tmp)$

 eq1: ratsimp(diff(diff(lag,th1t),t) - diff(lag,th1))$
 eq2: ratsimp(diff(diff(lag,th2t),t) - diff(lag,th2))$
 eq3: ratsimp(diff(diff(lag,th3t),t) - diff(lag,th3))$
 eq4: ratsimp(diff(diff(lag,d2yt),t) - diff(lag,d2y))$
 eq5: ratsimp(diff(diff(lag,d2zt),t) - diff(lag,d2z))$
 eq6: ratsimp(diff(diff(lag,d3yt),t) - diff(lag,d3y))$
 eq7: ratsimp(diff(diff(lag,d3zt),t) - diff(lag,d3z))$

 kill(lag);
 eqns:[eq1,eq2,eq3,eq4,eq5,eq6,eq7]$

 for i thru length(eqns) do (
        eqns[i]:ratsimp(eqns[i]),
        eqns[i]:crunch(eqns[i],dont,'aaa) )$
```

```
map(fortran,zz);

a:zeromatrix(7,7)$

for i thru length(eqns) do (
        tmp:expand((eqns[i])),
        a[i][1]: coeff(tmp,th1tt), a[i][2]: coeff(tmp,th2tt),
        a[i][3]: coeff(tmp,th3tt), a[i][4]: coeff(tmp,d2ytt),
        a[i][5]: coeff(tmp,d2ztt), a[i][6]: coeff(tmp,d3ytt),
        a[i][7]: coeff(tmp,d3ztt) );

fortran(a); /* generating the code for the mass matrix */

 eqns: subst(0, th1tt, eqns)$ eqns: subst(0, th2tt, eqns)$
 eqns: subst(0, th3tt, eqns)$ eqns: subst(0, d2ytt, eqns)$
 eqns: subst(0, d2ztt, eqns)$ eqns: subst(0, d3ytt, eqns)$
 eqns: subst(0, d3ztt, eqns)$

for i thru length(eqns) do (
        eqns[i]:ratsimp(-1*eqns[i]) )$

eqns: subst(x[1],th1t,eqns)$ eqns: subst(x[2],th2t,eqns)$
eqns:subst(x[3],th3t,eqns)$eqns:subst(x[4],d2yt,eqns)$
eqns:subst(x[5],d2zt,eqns)$ eqns: subst(x[6],d3yt,eqns)$
eqns: subst(x[7],d3zt,eqns)$ eqns: subst(x[8], th1, eqns)$
eqns: subst(x[9], th2, eqns)$ eqns: subst(x[10], th3, eqns)$
eqns: subst(x[11], d2y, eqns)$ eqns: subst(x[12], d2z, eqns)$
eqns: subst(x[13], d3y, eqns)$ eqns: subst(x[14], d3z, eqns)$

 fortran(eqns);

closefile(); quit();
```

B.2 Control Design Routines

In this section, two routines that were used in the process of designing controllers are described and coded in MATRIX$_X$ [37] command language. The first routine calculates the equivalent discrete regulator gains, while the second routine solves the Lyapunov's equation.

B.2.1 The Discrete-Equivalent-Gains Function

The regulator gains, designed in the continuous-time, have to be converted to the discrete-time since the implementation is done on a computer. The method used to calculate the equivalent discrete gains, implemented on MATRIX$_X$ [37], is presented as follows: The linear continuous-time model is given by

$$\dot{x} = Ax + Bu \tag{B.1}$$

$$y = Cx \tag{B.2}$$

The discrete-time model is given by

$$x(k+1) = A_d x(k) + B_d u(k) \tag{B.3}$$

$$y(k) = C_d x(k) \tag{B.4}$$

where, A_d and B_d are a function of the sampling rate T_s

$$A_d = e^{AT_s} \tag{B.5}$$

$$B_d = \int_0^\infty e^{AT_s} B \, dt \tag{B.6}$$

$$C_d = C \tag{B.7}$$

The continuous-cost function is

$$J = \frac{1}{2} \int_0^\infty (x^T Q x + u^T R u) \, dt \tag{B.8}$$

By minimizing this equation, the solution leads to the continuous regulator gains k.

The discrete equivalent cost function is

$$J_d = \frac{1}{2} \sum_{k=0}^\infty (x^T(k) Q_d x(k) + 2x^T(k) N_d u(k) + u^T(k) R_d u(k)) \tag{B.9}$$

where

$$Q_d = \int_0^{T_s} A_d^T Q A_d \, dt \tag{B.10}$$

$$N_d = \int_0^{T_s} A_d^T Q B_d \, dt \tag{B.11}$$

$$R_d = \int_0^{T_s} (B_d^T Q B_d + R) \, dt \tag{B.12}$$

Note that a cross term N_d exists in the transformation. By minimizing this equation (equation B.9), the discrete equivalent gains can be calculated. The MATRIX$_X$ user-

defined-function used to carry out these calculations is as follows

```
// [ Qd, Rd, Nd,dkr] = dreg_gains(S, Ns, Q, R, Ts)
//
// Function to calculate the equivalent discrete regulator gains
//
// inputs: S - system matrix
// Ns - number of states
// Q - continous state weighting matrix
// R - continous control weighting matrix
// Ts - sample time
//
// Outputs: Qd - discretized state weighting matrix
// Rd - discretized control weighting matrix
// Nd - discretized cross term of state & control weighting matrices
// dkr - regulator discrete gain
//
// split out system matrix
 [F, G, H, J] = split(S, Ns);
//
// dimensions
 [ Ns, M] = size(F);
 [ M, Nc] = size(G);
//
// calculate discretized weighting matrix
 Abar = [ F, G; 0.0*ones(Nc, Nc+Ns)];
 Bbar = [Q, 0.0*ones( Ns, Nc); 0.0*ones(Nc, Ns), R];
 Cbar = [ -Abar', Bbar; 0.0*ones(Ns + Nc, Ns + Nc), Abar];
 Dbar = exp(Cbar*Ts, 'pade');
 D1 = Dbar( 1:Ns+Nc, 1:Ns+Nc);
 E1 = Dbar( 1:Ns+Nc, Ns+Nc+1:2*(Ns+Nc) );
 D2 = Dbar( Ns+Nc+1:2*(Ns+Nc), Ns+Nc+1:2*(Ns+Nc) );
//
 Wt = D2'*E1;
 Qd = Wt( 1:Ns, 1:Ns);
 Rd = Wt(Ns+1: Ns+Nc, Ns+1:Ns+Nc);
 Nd = Wt( 1:Ns, Ns+1:Ns+Nc);
//
// calculating the discrete gains
 sd = disc(S, Ns, Ts);
 [ad, bd, cd, dd] = split(sd, Ns);
 [eval, dkr] = dregu(ad, bd, Qd, Rd, Nd);
//
 RETF
```

B.2.2 The Solution of Lyapunov Equation

In designing a controller using the model reference approach, the Lyapunov equation has to be solved. The equation is given by

$$A^T P + PA = -Q \qquad (B.13)$$

Where Q is a symmetric, positive definite matrix. The solution given by Jameson [82] is the commonly required for the stability of continuous-time system. Note that the solution available by MATRIX$_X$ will give different results. The method is as follows

Defining λ_i (i=1,...,n) be the eigenvalues of A, obtained by solving the characteristic equation of A which is given by

$$\lambda^n + \sum_{i=0}^{n-1} a_i \lambda^i = 0 \qquad \text{(B.14)}$$

Let

$$\Gamma_k = -(A^T)^{k-1} Q - \Gamma_{k-1} A \qquad \text{(B.15)}$$

where $\Gamma_0 = 0_n$ and (k=1,2,...,n). Defining

$$\theta = \prod_{i=1}^{n} (A^T + \lambda_i I_n) \qquad \text{(B.16)}$$

and

$$\psi = \Gamma_n + \sum_{i=1}^{n-1} (-1)^{n-i} a_i \Gamma_i \qquad \text{(B.17)}$$

Then, the solution of the Lyapunov's equation is

$$P = \theta^{-1} \psi \qquad \text{(B.18)}$$

The MATRIX$_X$ function, used to carry out the above calculations, is as follows

```
//  p = lyapj(a,q)
//
//  Lyapunov function solution using Jameson method
//  INPUTS: a - the state matrix
//  q - a symmetric positive definite matrix
//
//  Outputs: p - The solution for Lyapunov's Equation
//
 [nx,ny] = size(a);
 ai = poly(a);
 lamda = eig(a);
 theta = eye(nx);
 epsi = 0*eye(nx);
 gama = 0*eye(nx);
 for i=1 : nx-1; ...
      gama = -((a')**(i-1))*q - gama*a; ...
      theta = theta*(a'+ lamda(i)*eye(nx)); ...
      epsi = epsi + (-1)**(nx-i)*ai(nx-i+1)*gama;
      END;
 gama = -((a')**(nx-1))*q - gama*a;
 theta = theta*(a'+ lamda(nx)*eye(nx));
 epsi = gama + epsi;
 p = inv(theta)*epsi;
 retf
```

- APPENDIX C -

REAL-TIME SYSTEM AND ROBOT INTERFACE

This appendix contains a brief description of the real-time system boards, the operator manual control to safely drive the robot, the robot inverse kinematic equations, and the PC-based graphical interface program used to operate the robot.

C.1 The dSPACE System

The "DSP-CIT pro" hardware line, developed by dSPACE Inc., is composed of a set of boards specifically designed for high speed multivariable control systems based on Texas Instruments TMS 320 DSP family and hosted by an industry standard PC-AT. Compatibility, among the boards, is maintained by using a 32 bit wide peripheral high speed bus (PHS-bus). The PHS-bus provides a fast and very flexible I/O system with large addressing space. Figure (C.1) illustrates how the boards are used for general robotic applications. A brief description of the boards is as follows:

DS1002 Floating Point Processor Board

The DS1002 is the processor board. It is built around the Texas Instrument TMS 320C30 floating-point digital signal processor, capable of a computing performance of 33 MFLOPS. It has 60 ns cycle time. Up to 512 KBytes of fast RAM are available on the board, suitable to run most of the robotic control algorithms. Peripherals can be attached through the PHS-bus allowing 16 MB/sec transfer speed.
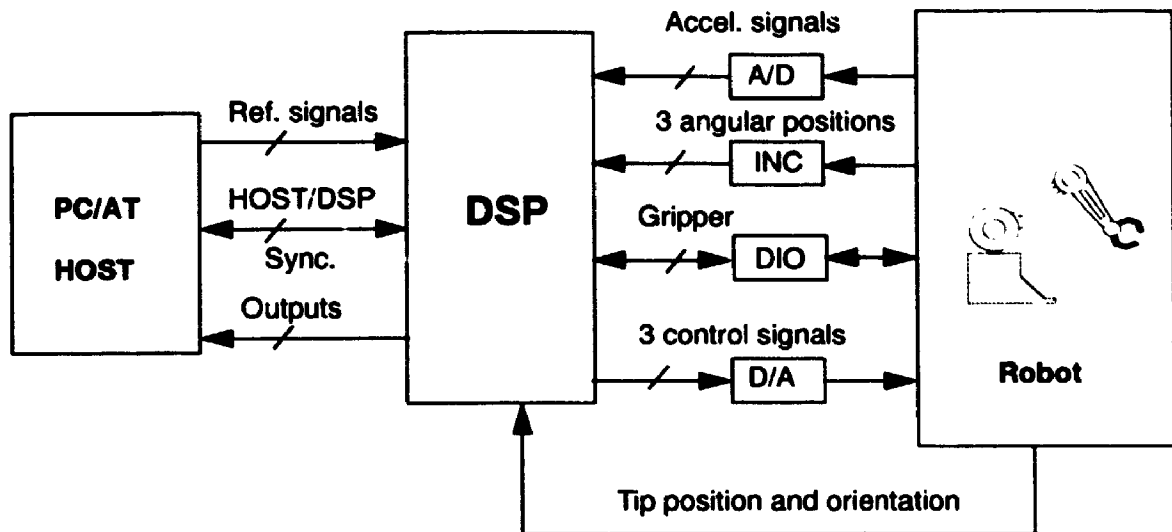
Figure (C.1): The DSP-CIT hardware and its use for robot control.

DS3001 Incremental Encoder Board

The DS3001 comprises 5 fully parallel input encoder channels. This board has a special noise rejection circuit, a fourfold pulse multiplication with direction sensing logic to give high resolution 2 bit counter for each channel, cable failure and encoder power supply failure detection.

DS2002 Multichannel A/D Board

32 high resolution analog to digital (A/D) channels are provided by this board. This is based on two 12 bit, 5 μs D/A converters. Each of them with 16-channel multiplexer and a fast sample/hold (S/H) circuit, and 4,8,12 bit short cycling. High linearity of 14 bits and selectable input voltage ranges accommodate precision control and measurements tasks. Various input ranges can be easily selected for precision measurement.

DS2101 Multichannel D/A Board

The DS2101 comprises 5 fully parallel analog outputs channels, equipped with 12 bit, 3μs digital to analog converters (D/A's). Flexible output capabilities, independent

unlatched output (minimum time delay). It also has a special D/A reset mechanism and PHS-bus interface to transfer data. The output ranges available for the L A are -5V to +5V, 0 to +10V and -10 to +10 V.

DS4001 Digital-I/O and Timer Board

This board was designed for applications with sensors providing a parallel digital output signal, with switch polling, frequency input and PWM output signals, and handling of external interrupts. The DS 4001 comprises 32 digital I/O lines which can be configured as inputs or outputs in groups of 8 lines for in the latched or unlatched mode. Five separate timers can be used for timer interrupts, frequency measure or PWM. Also, up to 8 external interrupt inputs for communication are available.

C.2 The Operator Control Panel

The operator control panel comprises a main power panel and three motor drive panels to control each joint independently. The description of these panels and the method of operation are as follows:

The Main Power Panel

As shown in Figure (C.2). It has a main power switch, start switch, stop switch and brake release switch. The brakes are normally engaged in case of power failure. The start button will release the brakes, the stop will hold the robot with the brakes engaged.

The Joint Drive Module (once per axis):

As shown in Figure (C.2), each joint drive module has a watch-dog enable switch and a watch-dog input channel. This circuit expects a 100 Hz, 1V, sine input and it will

disable the motor and engage the brake in case of not receiving this signal. This enables the operator to run each joint independently and to stop the robot in case of computer crash.

The joint drive module comprises a three state button switch and a drive signal input channel. The switch is normally in the middle state which the joint is driven by the input signal. If the switch is pressed up or down the input drive signal will be disabled and the motor will be driven by internal signal to move it in a tele-operation mode.

The Sequence of Operation:

1. The operator presses the power button to activate the modules.

2. The operator has to position the robot in the home position by releasing the brakes and position it manually.

3. When the drive signal is ready, the operator presses the start button to drive the motors or certain modules depending on the watch-dog signal or enable switch.

4. If the operator wants to run it in a tele-operation mode, the operator can press the CCW or CW drive. The robot will be in the stop state after this i.e. the brakes will be engaged and the drive signal will be disabled. The operator has to press start again to drive the motors using the external drive signal again.
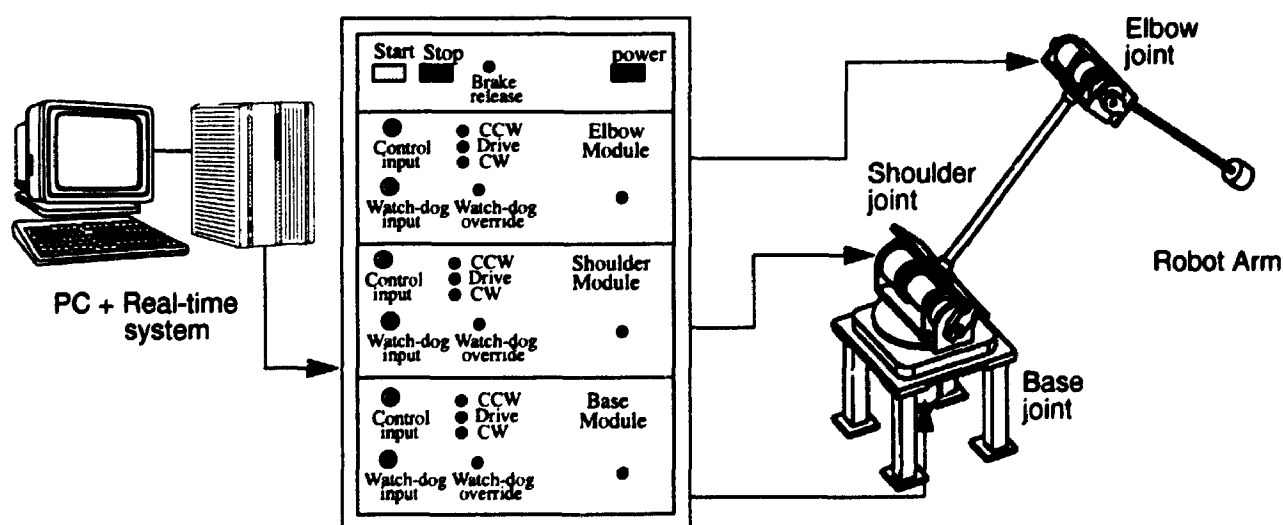
Figure (C.2): Robot drive panel.

C.3 The Robot Inverse Kinematic Equations

The inverse kinematic equations are used to drive the robot end-point along a line, circle, etc. The home position, which corresponds to zero values for the robot joint angles, is the position when the robot arm is straight up. This will make it easy to position the robot. By choosing the coordinate system to be as shown in Figure (C.3), where the z-axis points in-ward.

The position of the payload is measured in terms of thc distance of the payload from the intersection of centres of the base and shoulder. The desired end-point position is given in x, y, z, the lengths of the arms are L_2, L_3, the angles are θ_1, θ_2, θ_3. The equations for the inverse kinematics are as follows:

$$\rho = \sqrt{x^2 + y^2 + z^2} \tag{C.1}$$

$$\theta_1 = \text{atan}\left(\frac{z}{x}\right) \tag{C.2}$$

$$\theta_2 = \text{acos}\left(\frac{y}{\rho}\right) - \text{acos}\left(\frac{\rho^2 + L_2^2 - L_3^2}{2L_2\rho}\right) \tag{C.3}$$

$$\theta_3 = \text{acos}\left(\frac{\rho^2 - L_2^2 - L_3^3}{2L_3L_2}\right) \tag{C.4}$$

The inverse trigonometric functions are defined so that the inverse tangent returns an angle between -90 and +90 and the inverse cosine returns an angle between 0 - 180.
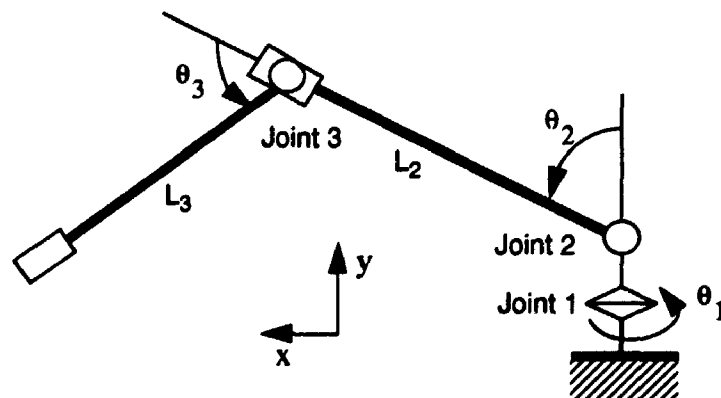
Figure (C.3): The robot coordinate system for inverse kinematics calculations.

C.4 The Robot Interface Program

The main objective of the robot interface is to provide a link between the control algorithm, running on the DSP, and the user commands issued on the PC.

For this purpose, two interface programs were developed. The first was a simple interface that was written using Microsoft Quick C. This interface allowed driving the robot joint along a series fifth-order step trajectories of given rise-time, collecting the data from the sensors and control outputs and writing them to a file.

However, the second interface was more sophisticated. It was based on Microsoft Windows and written using Borland C++. Figure (C.4) shows a typical screen display the robot controller program. This interface allows the generation of complex arm trajectories
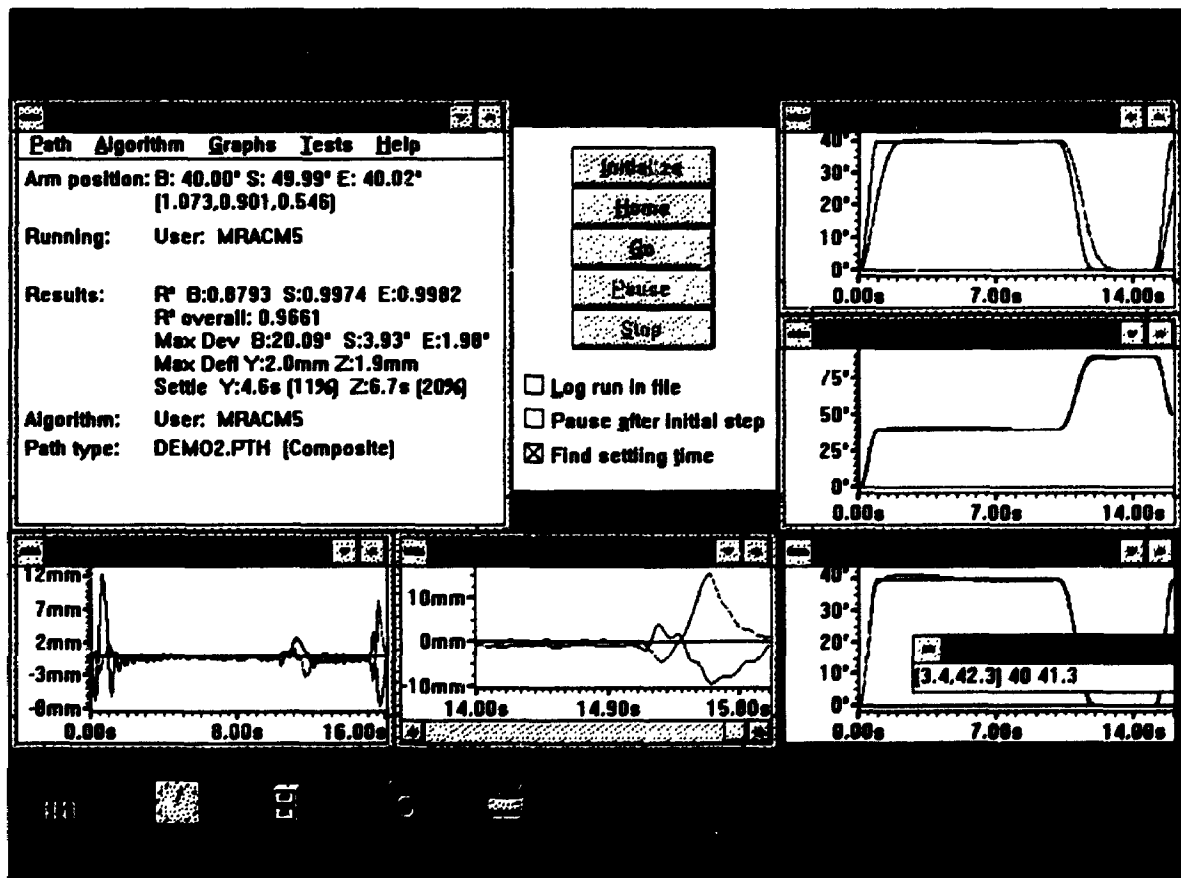
Figure (C.4): Sample screen display of the robot interface program.

from a number of simple primitives, the real-time execution of the calculated path and displaying of different sensors outputs, the capability to print, save, and compare different runs, the switching among different control algorithms, and finally, the ability to carry out statistical tests for the performance measures given in Chapter 11.

A detailed description of the program and how to use it are available in [81]. Note that the program is user-friendly and has an extensive on-line help system for ease of use. A sample of the program output is shown in Figure (C.5).
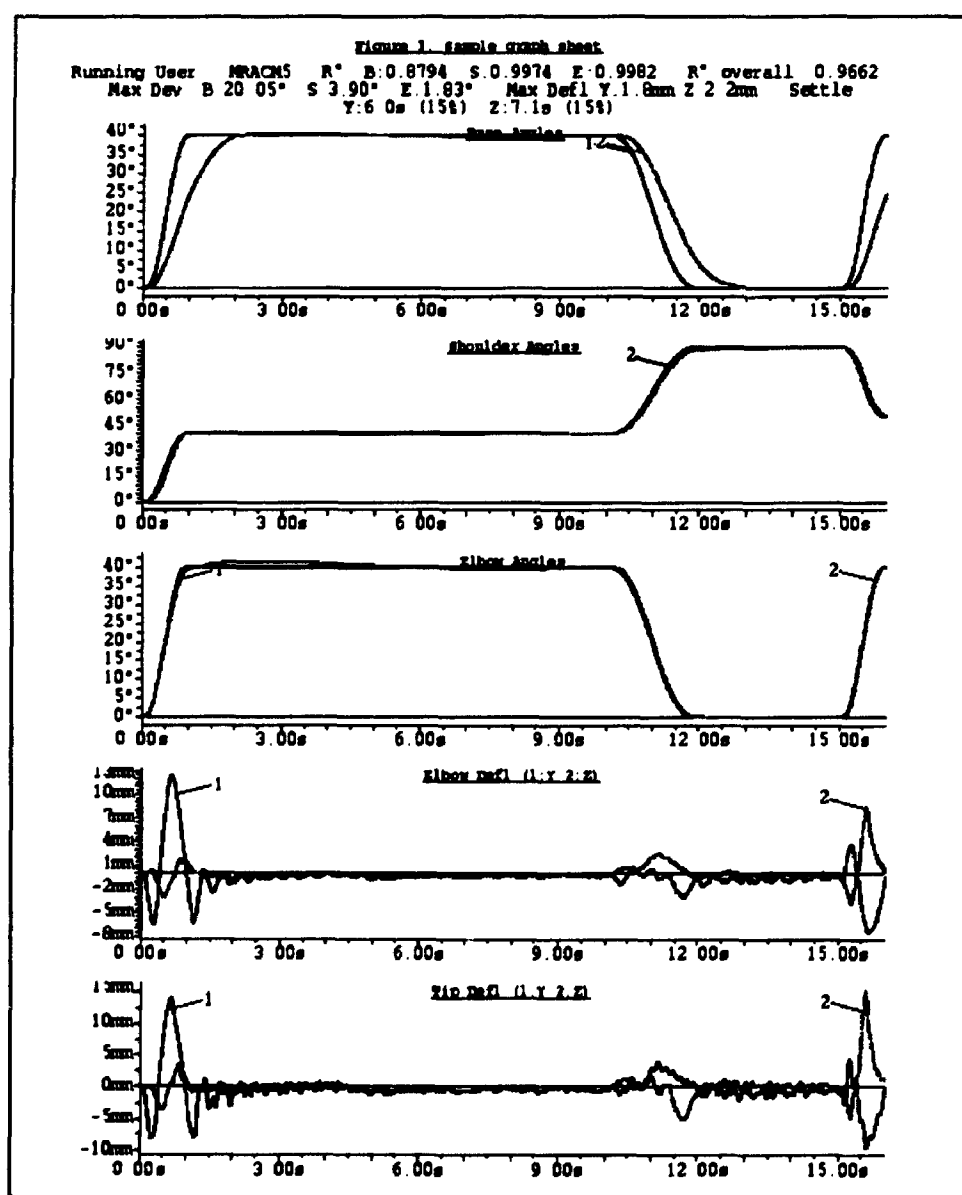
Figure (C.5): On-line print-out of the robot interface program.

REFERENCES

[1]   Alberts T. E., Dickerson S. L., and Book W. J., "Modelling and Control of Flexible Manipulators", Proceedings of Robot 9 Conference on Advancing Applications, Vol. 1, pp. 59-73, 1985.

[2]   Alberts T. E., Love L. J., Bayo E., and Moulin H., "Experiments with End-Point Control of a Flexible Link Using the Inverse Dynamics Approach and Passive Damping", Proceedings of the American Control Conference, pp. 350-355, 1990.

[3]   Armstrong B., "Friction Experimental Determination, Modelling and Compensation", Proceedings of the IEEE Conference on Robotics and Automation, pp. 1422-1427, 1988.

[4]   Bayo E., and Moulin H., "An Efficient Computation of the Inverse Dynamics of Flexible Manipulators in the Time Domain", Proceedings of the IEEE Conference on Robotics and Automation, pp. 710-715, 1989.

[5]   Bayo E., "Timoshenko Versus Bernoulli-Euler Beam Theories for the Inverse Dynamics of Flexible Robots", IEEE Transactions on Robotics and Automation, Vol. 5, No. 4, pp. 53-58, 1989.

[6]   Book W. J., "Modeling, Design, and Control of Flexible Manipulator Arms: A Tutorial Review", Proceedings of the IEEE Conference on Decision and Control, pp. 500-506, 1990.

[7]   Book W.J., "Recursive Lagrangian Dynamics of Flexible Manipulator Arms", The International Journal of Robotics Research, Vol. 3, No. 3, pp. 87-101, 1984.

[8]   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, pp. 62-75, 1984.

[9]   Canudas De Wit C., Astrom K. J., and Braun K., "Adaptive Friction Compensation in DC-Motor Drives", IEEE Transactions on Robotics and Automation, Vol. 3, No. 8, pp. 681-685, 1987.

[10]   Canudas De Wit C., and Slotine J.-J. E, "Sliding Observers for Robot Manipulators", Automatica Vol. 27, No. 5, pp. 859-864, 1991.

[11] Carusone J., and d'Eleuterio G., "Experiments in the Control of Structurally Flexible Manipulators with the *Radius* Facility ", Proceedings of the Second Joint Japan-USA Conference on Adaptive Structures, 1991.

[12] Centinkunt S. and Ittoop B., "Computer-Automated Symbolic Modelling of Dynamics of Robotic Manipulators with Flexible Links", IEEE Transactions on Robotics and Automation, Vol. 8, No. 1, pp. 94-105, 1992.

[13] Centinkunt S., and Book W. J., "Performance Limitations of Joint Variable-Feedback Controllers due to Manipulator Structural Flexibility", IEEE Transactions on Robotics and Automation, Vol. 6, No. 2, pp. 219-231, 1990.

[14] Cetinkunt S., and Book W.J., "Symbolic Modelling and Dynamic Simulation of Robotic Manipulators with Compliant Links and Joints", Journal of Robotics & Computer-Integrated Manufacturing, Vol. 5, No. 4, pp. 301-310, 1989.

[15] Chen J. S., and Menq C. H., "Experiments on the Payload-Adaptation of a Flexible One-Link Manipulator with Unknown Payload", Proceedings of the IEEE Conference on Robotics and Automation, pp. 1614-1619, 1990.

[16] Christian A.D., "Design and Implementation of a flexible robot", Technical Report No. AI-TR 1153, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1989.

[17] Computer Aided Design Software Inc. (CADSI), Oakdale, Iowa, "DADS User's Manual Version 6", 1992.

[18] Craig J. J., Hsu P., and Sastry S. S., "Adaptive Control of Mechanical Manipulators", The International Journal of Robotics Research, Vol. 6, No. 2, pp. 16-28, 1987.

[19] Craig J.J., "Adaptive Control of Mechanical Manipulators", Addison-Wesley, Reading, Massachusetts, 1989.

[20] De Luca A. and Siciliano B., "Trajectory Control of a Nonlinear One-link Flexible Arm", The International Journal of Control, Vol. 50, No. 5, pp. 1699-1715, 1989.

[21] Dubowsky S. and DesForges D.T., "The Application of Model Reference Adaptive Control to Robot Manipulators", ASME Transactions on Dynamic Systems Measurements and Control, vol. 101, pp. 193-200, 1979.

[22] ElDeeb Y., Zaki A., and ElMaraghy, W.H., "A comparison between fuzzy logic and model reference adaptive control for robotic manipulators", Proceedings of the IASTED International Conference on Robotics and Manufacturing, pp. 198-201, 1993.

[23] Ewins, D. J., "Modal Testing: Theory and Practice", Research Studies Press Ltd., Hertfordshire, England, 1986.

[24] Franklin G., Powell J.D., and Workman M.L., "Digital Control of Dynamic Systems", Addison-Wesley, 1990.

[25] Ghorbel F., Fitzmorris A., and Spong M.W., "Robustness of Adaptive Control of Robots: Theory and Experiment", Proceedings of the International Workshop on Nonlinear and Adaptive Control: Issues in Robotics, pp. 1-29. 1990.

[26] Gomes S.C. P., and Chretien J. P., "Dynamic Modelling and Friction Compensated Control of a Robot Manipulator Joint", Proceedings of the IEEE Conference on Robotics and Automation, pp. 1429-1435, 1992.

[27] Grossman W., Khorrami F., and Friedland B., "An Observer-Based Design for Robust Control of Robot Manipulators", Proceedings of the American Control Conference, pp. 731-736, 1990.

[28] Harris C.M., "Shock and Vibration Handbook", McGraw-Hill Inc., Third Edition, 1988.

[29] Hastings G. G., "Controlling Flexible Manipulators, An Experimental Investigation", PhD thesis, Georgia Institute of Technology, Department of Mechanical Engineering, 1986.

[30] He. S. X., "Multibond Graph Method For Flexible Multibody System Dynamics", PhD thesis, The University of Western Ontario, Department of Mechanical Engineering, 1992.

[31] Henrichfreise H., "The Control of an Elastic Manipulation Device Using DSP", Proceedings of the American Control Conference, pp. 1029-1035, 1988.

[32] Hirschorn R. M., "Invertability of Multivariable Nonlinear Control Systems", IEEE Transactions on Automatic Control, Vol. 24, pp. 855-865, 1979.

[33] Hogan N., and Flash T., "The Coordination of Arm Movements, An Experimental-

ly Confirmed Mathematical Model", AI Memo 786, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1984.

[34] Hollars M.G., "Experiments in End-Point Control of Manipulators with Elastic Drives", PhD thesis, Stanford University, Department of Aeronautics and Astronautics, 1988.

[35] Hughes P.C, "Space Structures Vibration Modes: How Many Exist? Which Ones are Important", IEEE Control Systems Magazine, pp. 22-28, 1987.

[36] Hughes P.C., "Dynamics of a Chain of Flexible Bodies", The Journal of the Astronautical Sciences, Vol. 27, No. 4, 1979.

[37] Integrated Systems Inc., Santa Clara, CA, "MATRIX$_X$ Version 3.0 User's Manual", 1992.

[38] Kane T. R. and Levinson D. A., "Dynamics: Theory and Application", McGraw-Hill Series in Mechanical Engineering, McGraw-Hill, 1985.

[39] Kane T. R., Rayan R. R., and Banerjee A. K., "Dynamics of a Cantilever Beam Attached to a Moving Base", AIAA Journal of Guidance, Control and Dynamics, Vol. 10, No. 2, pp. 139-151, 1987.

[40] Khosla P. K., and Kanade T., "Experimental Evaluation of Nonlinear Feedback and Feedforward Control Schemes for Manipulators", The International Journal of Robotics Research, Vol. 7, No. 1, pp. 18-28, 1988.

[41] King J. O., "Recursive Models and Controllers of Flexible Manipulators", PhD thesis, The University of Alberta, Department of Electrical Engineering, 1988.

[42] Kraft R. H., "Experiments in End-Point Control of a Flexible Robot with a Mini-Manipulator", PhD thesis, Stanford University, Department of Aeronautics and Astronautics, 1989.

[43] Kubo T., Anwar G. and Tomizuka M., "Application of Nonlinear Friction compensation to robot arm control", Proceedings of the IEEE Conference on Robotics and Automation, pp. 722-727, 1986.

[44] Kwon D., and Book W. J., "An inverse Dynamic Method Yielding Flexible Manipulator State Trajectories", Proceedings of the American Control Conference, pp. 186-193, 1990.

[45] Labinaz G., Belanger P., and Dobrovolny P., "Modelling and Frequency Domain Design of a Single Joint Torque Controller", Presented at the second IRIS workshop on Flexible Manipulators, Vancouver B.C., 1993.

[46] Landau, Y.D., "Adaptive Control: The Model Reference Approach", Marcel Dekker, Inc. NY. 1979.

[47] 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, pp. 116-120, 1987.

[48] Luh J. Y. S., "Conventional Controller Design for Industrial Robots-A Tutorial", IEEE Transactions on Systems, Man and Cybernetics, Vol. 13, No. 3, pp. 298-316, 1983.

[49] Luh J. Y. S., Fisher W. D., and Paul R. P. C., "Joint Torque Control by a Direct Feedback for Industrial Robots", IEEE Transactions on Automatic Control, Vol. 28, No. 2, pp. 153-161, 1983.

[50] Magee D.P., and Book. W.J., "Implementing Modified Command Filtering to Eliminate Multiple Vibration Modes", Proceedings of the American Control Conference, pp. 2700 - 2704, 1993.

[51] Massoud A. T., and ElMaraghy H.A., "Design, Dynamics, and Identification of a Flexible Joint Robot Manipulator", Proceedings of the IASTED International Conference on Robotics and Manufacturing, pp. 72-75, 1993.

[52] Meirovitch L., "Analytical Methods in Vibration", Macmillan Series in Applied Mechanics, Macmillan Co., N.Y., 1967.

[53] Mir H., Zaki A. and ElMaraghy W.H., "An Experimental Study to Determine the Dynamic Characteristics of a Two-Link 3 DOF Manipulator Constructed from Composite Materials", submitted to the 1994 SEM Spring Conference on Experimental Mechanics.

[54] Misawa E. A., and Hedrick J. K., "Nonlinear Observers - A State-of-the-Art Survey", Transactions of the ASME Journal of Dynamic Systems, Measurements and Control, Vol. 111, No. 3, pp. 344-352, 1989.

[55] Mitchell and Gauthier Assoc., Concord, Mass., "Advanced Continuous Simulation

Language (ACSL)", 1990.

[56]   Modi V. J., Suleman A., and Ng A. C.,"On the Modal Representation and Dynamics of the Evolving Space Station Freedom", Proceedings of the European Space Agency, 1992.

[57]   Modi V.J., Karray F. and Chan J.K., "On the Control of a Class of Flexible Manipulators Using Feedback Linearization Approach", Proceedings 42nd Congress of the International Astronautical Federation, 1991.

[58]   Morris K. A., and Vidyasagar M., "Modelling of Beam Vibrations for the Purpose of Controller Design", Proceedings of the ASME Winter Annual Meeting, pp. 17-26, 1988.

[59]   Mujtaba M.S., "Exploratory Study of Computer Integrated Assembly Systems", Technical Report No. STAN-CS-76-568, Stanford University, Computer Science Department, 1977.

[60]   Naganathan G., and Soni A. H., "An Analytical and Experimental Investigation of Flexible Manipulators Performance", Proceedings of the IEEE Conference on Robotics and Automation, pp. 767-773, 1987.

[61]   Oakley C. M., and Cannon R. H., "Anatomy of an Experimental Two-Link Flexible Manipulator under End-Point Control", Proceedings of the IEEE Conference on Decision and Control, 1990.

[62]   Oakley C. M., and Cannon R. H., "Equations of Motion for an Experimental Planar Two-Link Flexible Manipulator", Proceedings of the ASME Winter Annual Meeting, pp. 267-278, 1989.

[63]   Oakley C. M., and Cannon R. H.,"Theory and Experiments in Selecting Mode Shapes for Two-Link Flexible Manipulators", Proceedings of the First International Symposium of Experimental Robotics, pp. 1-19, 1989.

[64]   Ortega R. and Spong M.W., "Adaptive motion control of rigid robots: a tutorial", Automatica, Vol. 25, No. 6, pp. 877-888, 1989.

[65]   Panzieri S., and Ulivi G., "Design and Implementation of a State Observer for a Flexible Robot", Proceedings of the IEEE Conference on Robotics and Automation, pp. 204-209, 1993.

[66] Pfeiffer F., and Gebler B., "A Multistage Approach on the Dynamics and Control of Elastic Robots ", Proceedings of the IEEE Conference on Robotics and Automation, 1988.

[67] Rovner D. M., and Cannon R. H., "Experiments Toward On-Line Identification and Control of a Very Flexible One-Link Manipulator", The International Journal of Robotics Research, Vol. 6, No. 4, pp. 3-19, 1987.

[68] Rugh W., "Analytical Framework for Gain Scheduling", IEEE Control Systems Magazine, pp. 79-84, 1991.

[69] Sadegh N., and Horowitz R., "Stability Analysis of an Adaptive Controller for Robotic Manipulators", Proceedings of the IEEE Conference on Robotics and Automation, pp. 1223-1229, 1987.

[70] Sakawa K., Matsung F., Ohsawa Y., Kiyohara M., and Abe T., "Modelling and Vibration Control of a Flexible Manipulator with Three Axes by Using Accelerometers", Japan Journal of Advanced Robotics, Vol. 4, No. 2, pp. 119-137, 1990.

[71] Schmitz E., "Experiments on the End-Point Position Control of a Very Flexible One-Link Manipulator", PhD thesis, Stanford University, Department of Aeronautics and Astronautics, 1985.

[72] Shigley J. E., and Mischke C. R., "Standard Handbook of Machine Design", McGraw Hill Inc., 1986.

[73] Siciliano B. and Book W.J., "A Singular Perturbation Approach to Control a Lightweight Flexible Manipulator", The International Journal of Robotics Research, Vol. 7, No. 4, pp. 79-90, 1990.

[74] Siciliano B., Yuan B., and Book W. J., "Model Reference Adaptive Control of a One Link Flexible Arm", Proceedings of the IEEE Conference on Decision and Control, pp. 91-95, 1986.

[75] Slotine J. -J. E., Hedrick J. K., and Misawa E. A., "On Sliding Observers for Nonlinear Systems", Transactions of the ASME Journal of Dynamic Systems, Measurements and Control, Vol. 109, pp. 245-252, 1987.

[76] Slotine J.-J. E, and Li W., "Adaptive Robot Control: a New Perspective", Proceedings of the IEEE Conference on Robotics and Automation, 1987.

[77] Slotine, J.-J.E. and Li, W., "Applied Nonlinear Control", Printce Hall, New Jersey, 1991.

[78] Slotine, J.-J.E. and Li, W., "On the Adaptive Control of Robot Manipulators", International Journal of Robotics Research, Vol. 6, No. 3, pp. 49-59, 1987.

[79] Spong M. W., Khorasani K., and Kokotovic P.V., "An Integral Manifold Approach to the Feedback Control of Flexible Joint Robots", IEEE Transactions on Robotics and Automation, Vol. 3, No. 4, pp. 291-300, 1988.

[80] Spong M.W., Vidyasagar M., "Robot Dynamics and Control" by John Wiley & Sons, Inc., 1989.

[81] Stokes P., "A Graphical Interface for the DAMRL Flexible Manipulator", DA&M-RL Summer Research Report, Sept. 1993.

[82] Stoten, D.P., "Model Reference Adaptive Control of Manipulators", Research Studies Press Ltd., Somerset England, 1990.

[83] Stoten, D.P., and Benchoubane H., "Empirical Studies of an MRAC algorithm with Minimal Controller Synthesis", International Journal of Control, Vol. 51, No. 4, pp. 823-849, 1990.

[84] Structural Dynamics Research Inc. (SDRC), Milford, Ohio, "I-DEAS Ver. 6.0 User's Guide", 1992.

[85] Sunada W. and Dubowsky S., "On the Dynamic Analysis and Behavior of Industrial Robotic Manipulators with Elastic Members", Trans. of the ASME Journal of Mechanisms, Transmissions and Automation in Design, Vol. 105, pp. 42-50, 1983.

[86] Swanson Analysis Systems Inc., Houston PS, "ANSYS User's Manual Version 4.4", 1990.

[87] Symbolic Dynamics Inc., 928 Wright Ave., Mountain View, CA., "SD/EXACT and SD/FAST User's Manual", 1989.

[88] Symbolics Inc., Arlington MA, "MACSYMA User's Guide", 1988.

[89] Toogood R. W., Kermack I., Mackay D. J., and El-Rayyes L., "Using Symbolic Generation for Dynamic Analysis of Robotic Manipulators", IASTED International Symposium on Robotics and Manufacturing, pp. 76-79, 1990.

[90] Tsujio S., "A New Approach to Inverse Dynamics of Flexible Manipulator Arms",

Proceedings of the USA-Japan symposium on Flexible Automation, pp. 375-382, 1988.

[91] Tsujisawa T., and Book W. J., "A Reduced Order Model Derivation for Lightweight Arms with a Parallel Mechanism", Proceedings of the IEEE Conference on Robotics and Automation, pp. 728-735, 1989.

[92] Tzou H. S., Wan G. C., and Tseng C. I., "Dynamics and Distributed Vibration Controls of Flexible Manipulators", Proceedings of the IEEE Conference on Robotics and Automation, pp. 1716-1721, 1989.

[93] Uhlik C.R., "Experiments in High-Performance Nonlinear and Adaptive Control of a Two-link, Flexible-Drive-Train Manipulator", PhD thesis, Stanford University, Department of Aeronautics and Astronautics, 1990.

[94] Wang D., "Modelling and Control of Multi-Link Manipulators with One Flexible Link", PhD thesis, University of Waterloo, Department of Electrical Engineering, 1989.

[95] Wang D., and Vidyasagar M., "Modelling a Class of Multilink Manipulators with the Last Link Flexible", IEEE Transactions on Robotics and Automation, Vol. 8, No. 1, pp. 33-41, 1992.

[96] Wang D., and Vidyasagar M., "Passive Control of a Single Flexible Link", Proceedings of the IEEE Conference on Robotics and Automation, pp. 1432-1436, 1990.

[97] White A.S., Gleeson P.T., Wong Y., "Robot Vibrations", Environmental Engineering, Vol. 3, No. 1, pp. 26-29, 1990.

[98] Yigit A. S., and Ulsoy A. G., "Controller Design for Rigid-Flexible Multibody Systems", Proceedings of the IEEE Conference on Decision and Control, pp. 665-670, 1989.

[99] Yoshikawa T., Murakami H., and Y. Hosoda K, "Modelling and Control of a Three Degree of Freedom Manipulator with Two Flexible Links", Proceedings of the Second International Symposium on Experimental Robotics, 1991.

[100] Yuan B., "Adaptive Strategies for Controls of Flexible Arms", PhD thesis, Georgia Institute of Technology, Department of Mechanical Engineering, 1989.

[101] Zaki A, and ElMaraghy W. H., "Model Reference Adaptive Control for a Three-Degree of Freedom Manipulator with Flexible Links ", Proceedings of the American Control Conference, pp. 1400- 1406, 1993.

[102] Zaki A. and ElMaraghy W. H., "Modelling and Cor'rol of a Two-Link Flexible Manipulator", Transactions of the CSME, Vol. 16, No. 3/4, pp. 311-327, 1992.

[103] Zaki A., Mir H. and ElMaraghy W.H., "Flexibility Modelling and Control of Industrial Robots", Proceedings of the IASTED Internatio.'·al Conference on Robotics and Control, pp. 83-86, 1992.

[104] Zaki A., Mir H. and ElMaraghy, W.H. "Initial Experiments on the UWO Flexible Manipulators", Proceedings the IASTED International Conference on Robotics and Manufacturing, pp. 81-83, 1993.

[105] Zaki A., Mir H. and ElMaraghy W.H., "Design of a Two-Link Flexible Manipulator", DA&MRL Lab. Report No. 91-09-01, Sept. 1991.

[106] Zaki A., Mir H., and ElMaraghy W.H., "Experimental Manipulator: Design and Performance" DAMRL Technical Report No. 92-09-01, 1992.

[107] Zaki A., and ElMaraghy W.H., "An Approach for the Adaptive Inverse Dynamics for Manipulators with Flexible links", Proceedings of the 6th International Conference on Advanced Robotics, Kyoto, Japan, 1993.

[108] Zaki A., Mir H., and ElMaraghy, W.H. "Design, testing and simulation of a flexible robot for the study of dynamics and Control at UWO", Proceedings of the IASTED International Conference on Robotics and Manufacturing, pp. 195-197, 1993.

[109] Zaki A., ElMaraghy W.H., "Bond graph modelling of two-link manipulator with flexible links and joints", Proc. of the 10th IASTED Int. Conf. on Modelling and Simulation, Feb. 18-21, Innsbruck, Austria, 1991.

[110] Zaki A., and ElMaraghy W.H." An Overview of the Experimental Facility for the Study of Flexible Manipulators at UWO", Submitted to the 1994 CSME conference.

[111] Zames G., "Feedback and Optimal Sensitivity: Model Reference Transformations, Multiplicative Seminorms, ·~·" Approximate Inverses", IEEE Trans. on Automatic Control, Vol. AC-26, No. 2, pp. 301-320, 1981.