FINAL REPORT
ON
RESEARCH ON

END POINT CONTROL OF FLEXIBLE MANIPULATORS

at
STANFORD UNIVERSITY

by

The Stanford University Aerospace Robotics Laboratory
Department of Aeronautics and Astronautics
Stanford University, Stanford, CA 94305

Research Performed Under DARPA Contract MDA
903-86-K-0037
For the period October 1985 through October 1988

Professor Robert H. Cannon Jr.
Principal Investigator

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188
Exp. Date: Jun 30, 1986

1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS None	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Unrestricted APPROVED FOR PUBLIC RELEASE DISTRIBUTION UNLIMITED	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S)		5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION Stanford University	6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS *(City, State, and ZIP Code)* Department of Aeronautics and Astronautics Stanford University, Stanford, CA 94305		7b. ADDRESS *(City, State, and ZIP Code)*	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION DARPA	8b. OFFICE SYMBOL *(If applicable)* ISTO	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER MDA 903-86-K-0037	
8c. ADDRESS *(City, State, and ZIP Code)* 1400 Wilson Blvd. Arlington, VA 22209-2308		10. SOURCE OF FUNDING NUMBERS	

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

DTIC
ELECTE
JAN 19 1990
S E

11. TITLE *(Include Security Classification)*
End Point Control of Flexible Manipulators

12. PERSONAL AUTHOR(S)
Prof. Robert H. Cannon Jr.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
Final	FROM Oct 85 TO Oct 88		97

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB-GROUP	Robotics, manipulators, servo-controls, optimal control (Jk f)

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

This report covers research toward a technology for very quick, precise two-arm cooperative manipulation with the use of end-point control. The research addresses precison force control and dynamic cooperation of two manipulators.

Unavoidable flexibility in the drive trains of robots and the robots' mounts make precise end-point control of flexible robots important. This research is part of a series of experiments requiring increasingly more advanced control capability and working with exaggerated flexibility to force solution of fundamental problems.

The report covers 1) cooperating two-link arms with flexible drive trains, 2) control of a flexible robot arm with a two degree-of-freedom maini-manipulator, and 3) force control of a two-link arm with a flexible drive train. Keywords:

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION	
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Robert L. Rosenfeld	22b. TELEPHONE *(Include Area Code)* (202) 694-4001	22c. OFFICE SYMBOL ISTO

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

Contents

Executive Summary

List of Tables

~

List of Figures

Executive Summary

This document summarizes the work performed under DARPA contract No. MDA 903-86-K-0037 during the period October 1985 through October 1988 at the Stanford Aerospace Robotics Laboratory. The research was carried out by a team of Ph.D. candidate students under the direction of Professor Robert H. Cannon, Jr. The ultimate goal of this research is to develop and test a comprehensive base of new technology for very quick, precise two-arm cooperative manipulation with the use of powerful new end-point control algorithms.

In order to achieve this goal, the research addresses the set of necessary, major advances in robotic technology, including: precision force control and dynamic cooperation of two manipulators.

Robots in industry and in most research laboratories have position control of gross motion, not force control for fine motion of mating parts. Our strategy has been to study different aspects of force control of robots. Some of the tasks that have been achieved include making contact with objects, exerting controlled forces on objects, and making rapid motions of parts for assembly.

The obvious solution to end-point control of flexible manipulators is to measure directly those quantities that are to be controlled; namely, to measure the end-point position and force, and to feed these measurements back to a controller. Such a controller would enable one to use lighterweight, flexible, robots. There is, however, a good reason why this approach has not been widely used in the past: it poses an extremely severe stability problem.

Here, by way of introduction, we describe how we have proceeded. Then we present our research results to date in the formal Report on Research that is the body of this Final Report.

Industrial robots are made stiff enough and strong enough to ignore loads. Unavoidable flexibility in drive trains of robots, and in their mounts, make precise end-point control of flexible robots an issue of central importance. To advance the technology for force control of manipulators, and to also contribute to lowering the cost of using robots by making them much faster and more precise, we have developed a series of experimental systems that require that an increasingly more-advanced control capability be achieved to control them successfully. In each case, flexibility has been greatly exaggerated to force us to solve the control problem in a fundamental way.

Supporting research described in this report is aimed at advancing force control capabilities in the context of a single one-link flexible arm and a single two-link arm having a very flexible drive train. Again, the goal is to develop and test new techniques for effective two-arm cooperative manipulation using end-point control algorithms for manipulators with flexible drive trains.

Several of our tasks have been achieved through the use of a mini-manipulator. The mini-manipulator plays a key role in successful high-performance force control. The rationale for using a mini-manipulator at the tip of an arm is that a mini-manipulator can be used to perform tasks in a localized workspace at a much higher bandwidth than is achievable by the main arm. Two chapters in this report discuss force control experiments conducted with a mini-manipulator attached to the tip of a large manipulator arm.

We have systematically developed the supporting technologies for two-arm cooperation. A crucial capability for two-arm cooperation with flexibility in the drive train is force control at each arm tip, but that is not sufficient. Force sensing at each joint is also required to ensure that the commanded joint torque is actually achieved at the joint, with minimum phase loss. The hardware to implement two-arm cooperation has been designed, built, and is presently in the experimental stage. Larry Pfeffer is leading this research team.

Using the flexible arm, Ray Kraft has recently demonstrated position and force control with a two dimensional mini-manipulator at the end of the very flexible arm. Force control techniques with the mini-manipulator are thus significantly more complex than the force techniques we have previously developed with a single degree-of-freedom force sensor. Control of the extra degrees of freedom is not trivial, since the transformation between joint velocities and end-point velocities requires a computationally intensive Jacobian.

Brian Andersen is currently experimenting with a force control methodology using the mini-manipulator attached to the end of the two-link arm. This will be the first implementation of force control on a non-linear, multiple input-multiple output plant. This particular two-link arm has a flexible drive train, which increases the difficulty of the problem by an order of magnitude.

The main areas that are covered in the report are as follows:

1. Cooperating two-link arms with flexible drive trains.
2. Control of a flexible robot arm with a two-degree-of-freedom mini-manipulator.
3. Force control of a two-link arm with flexible drive train.

Subsequent chapters of this report describe in detail the progress in the above areas.

Chapter 1

Introduction

The Aerospace Robotics Laboratory has conducted a program of experimental research that has categorically responded to all aspects of the DARPA contract No. MDA 903-86-K-0037. The goal of this research is to develop – and demonstrate conclusively – key elements in the base of new technologies necessary for robots to advance to a new, important level of capability, namely, two-arm intimate cooperative manipulation that is swift, highly dextrous, precise, and sure, and that is managed astutely from a high level of task specification.

The power and generality that intimate cooperation and deft movement will give to robots is quite vast; it allows the performance of many new tasks previously entrusted only to humans. We have spent the past several years identifying critical enabling technologies, and have done focused experimental and theoretical work to advance them.

Specifically, if we are to succeed in achieving our ultimate goal, we must develop and demonstrate technology that increases the utility of sets of flexible manipulators by making them work cooperatively to handle tools and objects, especially elongated ones, accurately and with dispatch, and by providing and documenting key extensions to basic control theory that are necessary to accomplishing such a goal.

Our research indicates that the keys to deft and cooperative manipulation are the mastery of three capabilities: (1) end-point sensing and pressure control of force *and* position, (2) specification of desired tasks at the object level, and (3) control of quick, versatile end effectors, including mini-manipulators. For example, directly controlling the endpoint of flexible manipulators provides for a one-to-one matching with desired object motion (which joint-angle control does not).

The following are specific control capabilities, crucial to the achievement of our goals, for which we have developed algorithms and tested them experimentally:

1. Authoritative end-point control, both position and force.
2. Programmable compliance of the end effector.
3. Compensation for nonlinearities in the system's dynamics, and robustness to parameter variation, e.g., payload mass (including adaptive control).

4. Basic strategies for cooperation, including hierarchically structured task partitioning, control mode switching, and the astute management of redundant sensor information.

Our main emphasis is on the design and demonstration of these new control algorithms for the cooperative operation of multiple manipulators. Techniques that are generic to control of multiple flexible manipulators performing complicated tasks must be developed in such a way that others can then use them to achieve similar goals.

In a broader sense, the nature of our goals for a new level of robot capabilities has lead us to identify a number of advances that must be contributed to the body of basic, generic automatic control theory; and we have made a series of such contributions.

In achieving our research objective we intend to solve fundamental problems in the field of flexible manipulator control separately from the (equally important) problems of three dimensional geometry, gravity, complicated sensor systems, and sophisticated end effectors. This will be possible by operating, in these experiments, in a horizontal plane, and by using upgraded versions of the special sensors we have developed as surrogates for more sophisticated ones.

We must be able to take advantage of the flexibility in the manipulators to allow rapid movements without generating excessive contact forces, and we must demonstrate its value to cooperative tasks in which there is serious kinematic overconstraint.

The strategy we have used for achieving the set of enabling technology goals is to pursue a sequence of specific experimental demonstrations. Each demonstration was conceived to require a particular technology advance, and the set of them to enable the graceful, intense cooperation of two flexible arms.

1.1 Background

Current work achieved under this contract is explained in detail in the body of this report. The research centers around the following three major topics:

Creating an advanced test bed for, and achieving control of intimately cooperating two-link arms with flexible drive trains (Chapter 2).

Achieving very quick, precise force and position control of a flexible robot arm with a two-degree-of-freedom mini-manipulator (Chapter 3).

Achieving deft force control of a two-link arm with a flexible drive train and a two-degree-of-freedom mini-manipulator (Chapter 4).

The work which was drawn on to achieve the results in the body of this report follows most directly from the DARPA sponsored groundwork in previous contracts. We have also carried out some other work, sponsored by the AFOSR and NASA, in flexible two-link arm control and on spacecraft manipulators. The direct benefit of carrying out this related work, in the same lab, is the ability to easily drawn upon it. This applies not only to control algorithms, but also to software, computer resources, sensors, and in-house hardware designs.

As a sampling of recent experimental results that are germane, we have achieved:

1. The first end-point control of a very flexible manipulator.
2. The first real-time adaptive control of a very flexible beam.
3. The first experiments with potential-field guidance of manipulation.
4. The first switching, from (optical) position control to force control of a flexible manipulator contacting a moving object.
5. The first impedance-based hierarchical control of two cooperating two-link arms, in which the operator's commands are for object motion directly.
6. The first end-point control of a manipulator operating from a completely freely moving base.

The research in the Stanford Aerospace Robotics Lab has laid the foundation for the investigation of noncolocated control of flexible structures and end-point sensing.

We have addressed the basic question of what must be done to guarantee stability and robustness (insensitivity to large variations in plant parameters) in controllers using noncolocated sensors and actuators: and what are the ultimate limits to what can be achieved on a very flexible manipulator. Experiments on a system of four inertia disks connected by a torsion rod with 0.3% damping demonstrated the ease of achieving colocated control (due to alternating poles and zeros on the imaginary axis) and the difficulty of achieving stable and robust noncolocated control. LQG controllers could always be found to notch out resonant modes and stabilize a noncolocated system; but the controllers were extremely sensitive to variations in those resonant frequencies caused by changes in plant parameters.

Initial experiments in noncolocated control of very flexible (0.5Hz) manipulators were conducted by Eric Schmitz under NASA funding. Rigorous mathematical models of the flexible manipulator were developed to explain the dynamics of the structure. The fundamental response time limit in his single-link very flexible arm is associated with the bending wave propagation speed. Optical end-point sensing was used to locate a light at the arm's tip, and stable end-point position control was established. The closed-loop bandwidth was twice as high as the arm's first cantilevered frequency, which is the performance limit on the noncolocated control system.

Jim Maples extended this work under DARPA funding to stable force control at the manipulator's tip. A new end-point force sensor was developed to allow for the demonstration of a slew-and-touch maneuver to a moving target without pause in the arm's motion. Furthermore, very fast localized response has been achieved by adding a short rigid wrist to the end of a flexible arm. The rational was that many operations require fast, precise movement within a small workspace and relatively slower gross movement over larger distances. This was the beginning of noncolocated control of truly multiple-input multiple-output systems.

A much more complicated two-link manipulator system has been designed, built, and controlled by Mike Hollars under DARPA and AFOSR funding. This two-link arm has rigid members with lumped flexibility in the form of elastic linear springs. The two-dimensional geometry gives rise to highly nonlinear equations of motion which are more representative of modern industrial robots. Not only has stable colocated closed loop control been experimentally achieved, but also stable noncolocated control. The noncolocated controller uses a constant gain extended Kalman filter to estimate the state of the two-link arm.

Stan Schneider, working under NASA funding, has completed work on a fixed-base cooperative manipulation experiment where the problem of utilizing multiple rigid robotic arms to control a single object was examined. This research had three major goals:

1. Experimental evaluation of fundamental cooperative dynamic control techniques.
2. Investigation of manipulation strategies, especially as they pertain to multiply-armed robotic systems.
3. Achieving a heirarchical control system in which, for the first time, the operator's commands are issued at the object level.

In addition to cooperative dynamic control, the experimental system incorporated real-time vision feedback, a novel programming technique, and a graphical high-level user interface. Not only these subsystems, but also their interfaces and interactions have been studied, and a total, optimized system achieved.

We have systematically developed the supporting technologies for flexible two-arm cooperation. A crucial capability for two-arm cooperation with flexibility in the drive train is force control at each arm tip, but that is not sufficient. Force sensing at each joint is also required to ensure that the commanded joint torque is actually achieved at the joint, with minimum phase loss. The hardware to implement two-arm cooperation has been designed, built, and is presently in the experimental stage.

1.2 Future Plans

Our future research plans include making contributions of three kinds that are central to advancing robot capability from its present level to the much wider realm of use for which it is inherently competent:

1. We will make key new advances in the specific technologies that are critical to achieving true, effective integrated cooperation between compliant manipulators;
2. We will advance the art of motion planning and execution, for both gross and fine maneuvering of awkward objects amidst obstacles, so that users can specify object motions at a high level; and
3. We will unite the path planning and manipulation functions through a major fundamental joint effort – with the Stanford Computer Science Robotics Laboratory – in the foundation and achievement of comprehensive vertical integration.

Specifically, we will demonstrate experimentally the intimate vertical integration of path planning with the manipulative skills of a pair of deft, cooperating robotic arms in the rapid assembly of a complex configuration of awkward objects, all in response to only high-level instructions.

1.3 Summary

Within this report three major topics are addressed.

Chapter 2 discusses cooperating two-link arms with flexible drive trains. The basic objective of this research is to develop and demonstrate cooperative control of flexible robots. The extension of robots to tasks that are not readily accomplished with one (robot) hand has distinct advantages: cooperating robots can handle a broader class of tasks, and they may be accomplished faster, more accurately, and in a more loosely structured environment.

For generality, we wanted a pair of 6 DOF manipulators with a large dexterous workspace. Furthermore, we wanted to be able to change the way the workspaces overlapped for different experiments. However, we wanted to avoid the time and cost of working against gravity in the long primary links and to have a simple means to lessen the chance of collision (as collision avoidance a primary objective for other research). This lead us to the SCARA configuration, with the arms side by side, one left handed and one right handed.

The SCARA configuration keeps the first two links (those with the greatest mass and moment arms) in the horizontal plane. This configuration also allows us to begin cooperative work with less chance of collisions, as the robots generally have their elbows well away from each other. This strategy allows us to avoid arm collisions by monitoring endpoint position primarily. The kinematics are also considerably simplified by the design's four parallel axes. In our design, the actuators for the first two links remain fixed to the base (as in the Adept I robot) so that the mass of those actuators never moves. Similarly, we use brushless motors, and low gear ratios for these unmoving actuators, so that we have low levels of coulomb friction in the degrees of freedom that have the flexibility and the primary responsibility during cooperation. For the rest of the degrees of freedom, we accept the need for higher gearing using small servomotors with brushes.

Chapters 3 and 4 discuss the use of a mini-manipulator. The mini-manipulator plays a key role in successful high-performance control. The rationale for using a mini-manipulator at the tip of an arm is that a mini-manipulator can be used to perform tasks in a localized workspace at a much higher bandwidth than is achievable by the main arm. By using end-point control in conjunction with a mini-manipulator, a flexible robot would be able to achieve high bandwidth, precise control over a large workspace. It is these considerations which motivated the research on position and force control of a flexible robot arms with a two-degree-of-freedom mini-manipulator.

Chapter 3 discusses control of a flexible robot arm with a two-degree-of-freedom mini-manipulator. The flexible arm has distributed flexibility and was intentionally designed to exaggerate structural flexibility in the horizontal plane while remaining stiff in the gravity influenced vertical direction. The purpose of exaggerating the horizontal flexibility in this system was two fold: (1) to make problems associated with flexibility more readily apparent, and (2) to lower the resonant frequencies of the flexible arm into a region where easy laboratory sampling rates, in the neighborhood of 25 to 100 Hz, could be used. A potentiometer is collocated with the hub motor and is used to measure the base angle of the flexible arm. A light emitting diode (LED) array, whose position can be sensed by a photodetector, is located at the end-point of the flexible arm.

The mini-manipulator is attached to the end-point of the flexible arm along one of the side-rails. Essentially, it is a five link, closed kinematic chain — the base link being fixed to the flexible arm. Each of the two inboard links is rigidly attached to one of the two motors that drive the mini-manipulator. Located at the joint between the two outboard links is another LED array which is used to sense the position of the end-point of the mini-manipulator.

Below this LED array is a vertical, force sensing, aluminum beam. It is approximately 9.5 cm long and is equipped with strain gauges which provide a means of measuring end-point force. At the bottom end of the force beam is a circular contact roller, which allows the mini-manipulator to exert only normal forces on objects it comes into contact with.

Finally, Chapter 4 describes work that has been accomplished to date on the two-link arm with a flexible tendon drive and mini-manipulator. The objective of this research is to allow the arm to slew into contact with a target, make a smooth touchdown, and maintain a constant force on the target, using the mini-manipulator for augmentation of precision control. We believe the mini-manipulator is going to play a key role in successful high-performance force control.

During the time covered by this report we have demonstrated initial position and force control with the mini-manipulator on a fixed base. This work will form the basis of the control algorithms when the mini-manipulator is added to the two-link arm.

The capability we wish to develop is very-high-bandwidth, precise control of the force at the tip of a two-link manipulator having a very flexible drive train. We plan to demonstrate this capability by (1) performing rapid slew and touch with a fixed target while having no overshoot in the force! (2) controlling the arm so that its tip moves along a wavy surface while maintaining a constant force on the surface, and (3) slewing into contact with a moving target and maintaining a constant force on it while it continues to move.

The chapters which follow, then, present the specific set of (we submit) crucial contributions that DARPA contract No. MDA 903-86-K-0037 has allowed us to make to a new technology base that will substantially advance the new capabilities and value of the next generation of robotic systems.

Chapter 2

Hardware Development, Modeling, and Experimental Control of Two Cooperating, Flexible-Drive Manipulators

Lawrence E. Pfeffer

Introduction

In this chapter, we report the progress and results from three years of experimental research on the cooperating, flexible-drive manipulator testbed in our laboratory. This testbed provides a base for the development, for the first time, of the following underlying technologies: hierarchical, cooperative control; concurrent mechanism/control design; realtime multiprocessor control of robots; advanced sensing/perception; and graphical meta-teleoperative user interfaces. These are technologies that are essential for the new level of robot capabilities and missions that our country needs for a strong defense system and a vigorous manufacturing economy.

The work reported here was performed during the last year of DARPA contract MDA903-86-K-0037. We preceed it in section 2.1 with an overview of the work reported previously. Then we develop (beginning in section 2.2) the complete kinematic model: rigorous coordinate frame definitions, full forward kinematics, the inverse kinematics. and the Jacobian matrix. The impact of the kinematics' structure on control algorithms is discussed, including computational complexity and effects of singularities. The next section, 2.3 presents the results from the identification and control experiments performed under this contract: mass property identifications, scale factor measurements, and describing function measurements of the testbed. The final section, 2.4 describes the follow-on research that will be performed on this testbed under a new DARPA contract.

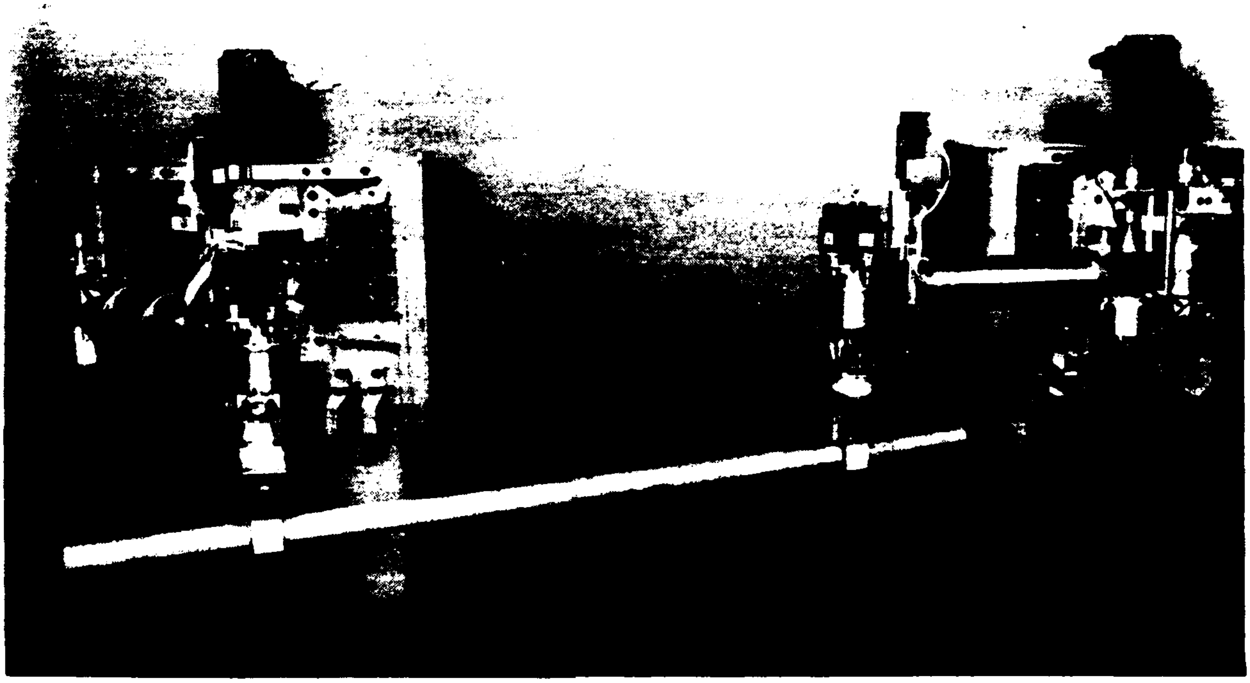

Figure 2.1: The Cooperating Flexible Manipulators Holding a Long Part

2.1 Summary of Previous Research Progress

Over the past three years, we have designed, built, and started experimental research on the cooperating flexible manipulators testbed. This research facility is comprised of a pair of manipulators designed for cooperation; a fast and flexible realtime control computer; a SUN workstation for user interaction and code development; and the sensor/actuator/safety interface subsystems that connect the realtime computer to the robots. Each phase of the design, fabrication, and experimentation has yielded valuable ideas in the theory and practice of robotics. This section is an overview, intended to provide a context for the rest of the report.

Figure 2.1 shows the two cooperating manipulators (posed handling a long part.) There are several aspects of the manipulator design that are worth emphasizing:

- Each arm is an identical six degree-of-freedom manipulator, and therefore can both position and orient its end effector within its dexterous workspace – a necessary geometric freedom for generic experiments in cooperation.
- The manipulators are of the basic SCARA configuration, with their first two links moving in a horizontal plane, and the third link a vertical linear stage. This configuration has two basic advantages: first, it eliminates large gravity torques on

the primary actuators; second, having the first four axes parallel greatly simplifiies the kinematics.

- The SCARA configuration provides a wide, flat dextrous workspace for each arm. This aids in task planning and demonstrations, as it is similar to the sort of two handed manipulation that a person (or two people) might do at a desk or workbench.
- The robots are each attached to individual mounts that are stiff, massive, yet movable. This permits free experimentation with varying degrees of workspace overlap and with different robot orientations, e.g. side by side, across from each other, or at right angles.
- The first two degrees of freedom (D.O.F.) of each arm have lumped (as opposed to distributed) flexibility built into their drive systems. These are the D.O.F. that correspond to the large motions (and large velocities) that span the mutual workspace. This flexibility, properly controlled, permits cooperative control that is precise and gentle without requiring a perfectly known and structured environment.
- There are sensors for each actuator, and more importantly, endpoint and intermediate joint sensors to measure the quantities that matter most in cooperation.

The basic objective of this research is to develop and demonstrate cooperative control of flexible robots. The extension of robots to tasks that are not readily accomplished with one (robot) hand has distinct advantages: cooperating robots can handle a broader class of tasks, and they may be accomplished faster, more accurately, and in a more loosely structured environment. The background, motivation and objectives are detailed in last year's report. The effects of the project objectives on the design are discussed next.

2.1.1 Design Philosophy

The design philosophy for this project has been to develop a versatile testbed that is easy to experiment with, and at the same time to minimize complications not in the mainstream of our research.

For generality, we wanted a pair of 6 DOF manipulators with a large dexterous workspace. Furthermore, we wanted to be able to change the way the workspaces overlapped for different experiments. However, we wanted to avoid the time and cost of working against gravity in the long primary links and to have a simple means to lessen the chance of collision, as collision avoidance is not one of our primary objectives. This lead us to the SCARA configuration, with the arms side by side, one left handed and one right handed.

The SCARA configuration keeps the first two links (those with the greatest mass and moment arms) in the horizontal plane. This configuration also allows us to begin cooperative work with less chance of collisions, as the robots generally have their elbows well away from the other. This strategy allows us to avoid collisions by monitoring endpoint position primarily. The kinematics are also considerably simplified by the design's four parallel axes. In our design, the actuators for the first two links remain fixed to the base (as in the Adept I robot) so that the mass of those actuators never moves. Similarly, we use brushless motors and low gear ratios for these unmoving actuators, so that we have low levels of coulomb friction in the degrees of freedom that have the flexibility and the primary responsibility during cooperation. For the rest of the degrees of freedom, we accept the need for higher gearing and small servomotors with brushes.

The robots are extremely well instrumented. There are position sensors on each actuator (digital shaft encoders), which are supplemented by resolvers on the primary motors (for the first two DOF) that yield analog position and rate. The two primary joints, separated from their actuators by the flexibility, are instrumented with higher resolution encoders for endpoint sensing. Force and torques are measured by joint torque sensors built into the first two joints, as well as by endpoint force/torque sensors (not shown mounted in photograph.) These sensors make it possible to obtain an accurate and timely estimate of both the actuator and endpoint quantities of interest. This is a key to good control in general and cooperation in particular.

2.1.2 Computer system

The realtime computer for the project is a multiple processor system coupled to the rest of the lab's computing resources over the ethernet. The realtime system is capable of rapid computation of complex control algorithms, hierarchical control, modular expansion or upgrading, as well as ease of experimentation and data collection.

The overall computer consists of processors that are tied together through a high-speed communications systems. The development system, or host, is the computer that the user interacts directly with. Ours is a SUN-3/140 workstation. The realtime control system was developed in our laboratory as a platform for controls research. The realtime system is a multiple processor computer, composed of single board processors and peripheral devices for communications with the robots, the host computer, and other experimental equipment. One of the key aspects to the power and utility of this system is that both the host and the realtime system use the same model microprocessor, and the same model of electrical backplane (bus). This allows us to develop and test software for the realtime system using all the tools available on the host.

The processor boards for the realtime system are MVME 147 and MVME133 single board processors manufactured by Motorola Inc., microcomputer division. These cards run at 16 -24 MHz, and each includes a 32 bit microprocessor (68030 or 68020, respectively), a hardware floating point co-processor (68882 or 68881), one to four megabytes of memory, as well as timers, a serial port, and sockets for ROM, etc. In terms of floating point calculation speed, (which is the major bottleneck in robotics/controls work) one of these cards is substantially faster than a VAX-11/780, especially on transcendental functions that are common time sinks (e.g. trig, arctrig, and square root) in robotics.

The current realtime system consists of two processor cards (with the possibility of adding more), an analog to digital converter board (for reading analog signals), digital to analog boards (for commanding actuators), a digital input/output board (for reading/writing digital signals, e.g., reading shaft encoder inputs, or writing enable bits to the safety system.)

This overview has described, in general terms, the experimental testbed that we have built, analysed, and begun experimenting with over the last three years. The next sections describe, in greater detail, the final year's research using the testbed.

2.2 Kinematic Model

In order to build up a usable model of the robot system (beginning here with the robots' kinematics), we need to have a set of well defined coordinate frames. Any ambiguity or mistakes at this stage will ripple through the entire model building process, rendering it useless. Thus, we have to be precise about our model's foundations. There are two types of coordinate systems that we will be interested in, these are:

Cartesian: These are coordinate systems that use three mutually perpendicular (right handed throughout this discussion) basis vectors and an origin to define a coordinate frame from which positions and orientations are measured. Positions are defined by taking components of the position vector parallel to each of the basis vectors. Orientations are described by a series of roll-pitch-yaw rotations (1-2-1 body fixed rotations, true Euler angles) about axes that begin aligned with the reference frame. Our primary interest is in the robot's endpoint/toolpoint position measured in the robot's base frame [2, page 88]. Intermediate frames are introduced, as well, for steps in the analysis, these are attached to each link of the manipulator.

Relative-displacement: These are coordinate systems that describe the configuration of the robot in terms of (linear or angular) displacements of parts of the mechanism relative to one another. We will be primarily interested in joint space and actuator space. The joint space coordinates of the robot are defined as an ordered vector of the robot's joint displacements (our link frames and joint coordinates were described in last year's annual report [1]. The actuator space coordinates are an ordered vector of actuator displacements, in our case, these are all radian measures of the motors relative to the reference configuration.

2.2.0.1 Notation for Coordinates of Interest

Elements in a vector that denotes robot configuration (position and orientation) are called q's. This is in keeping with the nomenclature used in the dynamics methodology that we are using, and avoids the semantic violence of refering to a linear displacement as, for example, θ_3, when in reality the quantity is really a linear displacement. Following the q, is another letter that defines what kind of frame we are dealing with: We use c (for base frame Cartesian), j (for joint space), and m (for motor space). Finally, we append a subscript to denote which coordinate in the vector we are referring to, this subscript can be numeric (1 to 7) or a letter mnemonic that is meaningful in the coordinate space we are using. The definitions following give examples.

For the cartesian coordinate systems, we adopt the following format and nomenclature for vectors and their components. The cartesian space configuration vector is referred to as qc; the individual components of qc are defined below in their order.

X position	qc_1 or qc_X
Y position	qc_2 or qc_Y
Z position	qc_3 or qc_Z
yaw angle	qc_4 or qc_y
pitch angle	qc_5 or qc_p
roll angle	qc_6 or qc_r
Gripper jaw opening	qc_7 or qc_g

where positions are in meters and angles are in radians.

For (robot) joint space, the entire configuration vector is referred to as qj the vector format is:

Joint 1 (shoulder) angle	qj_1 or qj_s
Joint 2 (elbow) angle	qj_2 or qj_e
Joint 3 (vertical slide) position	qj_3 or qj_z
Joint 4 (yaw) angle	qj_4 or qj_y
Joint 5 (pitch) angle	qj_5 or qj_p
Joint 6 (roll) angle	qj_6 or qj_r
Joint 7 (gripper) opening distance	qj_7 or qj_g

where positions are in meters and angles are in radians. The joint coordinates follow the convention used by [2] angles are measured from the positive X axis of the previous frame to the positive X axis of the current one, i.e. qj_1 is the angle between $+X_0$ and $+X_1$. For the linear joints, qj_3 denotes the distance between the origins of frames 3 and 4; qj_7 denotes the distance between the gripper jaws.

For actuator space, the entire configuration vector is referred to as qm the vector format is:

Motor 1 (shoulder) angle	qm_1 or qm_s
Motor 2 (elbow) angle	qm_2 or qm_e
Motor 3 (vertical slide) angle	qm_3 or qm_z
Motor 4 (yaw) angle	qm_4 or qm_y
Motor 5 (pitch) angle	qm_5 or qm_p
Motor 6 (roll) angle	qm_6 or qm_r
Motor 7 (gripper slide) angle	qm_7 or qm_g

where angles are in radians.

The wrist mechanism we are using couples motors 4 through 6 into orientation in a complex (meaning not one to one, except for yaw) manner. Our transmission, in fact, also couples both of the first two motors into the elbow joint's motion, but the identification of shoulder and elbow motors is still clear, as the coupling is one way (only the shoulder motor affects the shoulder joint, the other of the two is called the elbow motor). The wrist coupling is less straightforward. The fourth joint coordinate, qj_4 depends only on one of the motor coordinates, namely qm_4. However, the next two joint coordinates, qj_5 and qj_6 depend on combinations of motor coordinates. Occasionally, we will refer to motors 4, 5, and 6, as the yaw, pitch, and roll motors. This is an arbitrary mnemonic device. It's origin is from visualizing being a pilot sitting in the tool, facing out along the tool's axis (opposite the Z_6 vector). From this point of view, one can think of roll, pitch, and yaw exactly as motions of $qj_6 = qj_r$, $qj_5 = qj_r$, and $qj_4 = qj_y$, respectively. Since qj_y depends only on qm_4, this motor can be referred to as the yaw motor, and we will occasionally refer to motors 5 and 6 as pitch and roll, even though this is arbitrary nomenclature it is occasionally useful.

2.2.0.2 Link Frames and Associated Transforms

We now present the definitions of the coordinate frames used in deriving the forward kinematics of one of our cooperating manipulators. We will follow the convention of [2], and will use homogeneous transforms as defined in his chapter 2 to define the transforms between each adjacent frame, and finally the full kinematic transform for one of our manipulators. The relative simplicity of our robots' kinematics will then be apparent, and will be addressed. Since the possible frame assignments for a manipulator are not unique, a brief description of the origin and orientation of each frame we use is presented.

Frame Definitions First, some comments and definitions that will make the frame descriptions clear and succinct:

All definitions proceed from the assumption that the robot base is resting on its intended supports, and has been properly leveled. Left and right are used as descriptive terms, and assume (unless specified otherwise) that the viewer is head up, and has his back to the front mounting plate of the robot. Similarly, forward is defined as the direction normal to the front mounting plate, and directed from the plate into the interior of the workspace. The wrist point is defined as that point (in the wrist mechanism) where orientation is independent of positioning that point. All coordinate frames are orthogonal and right handed.

Robot frame 0 Robot base frame

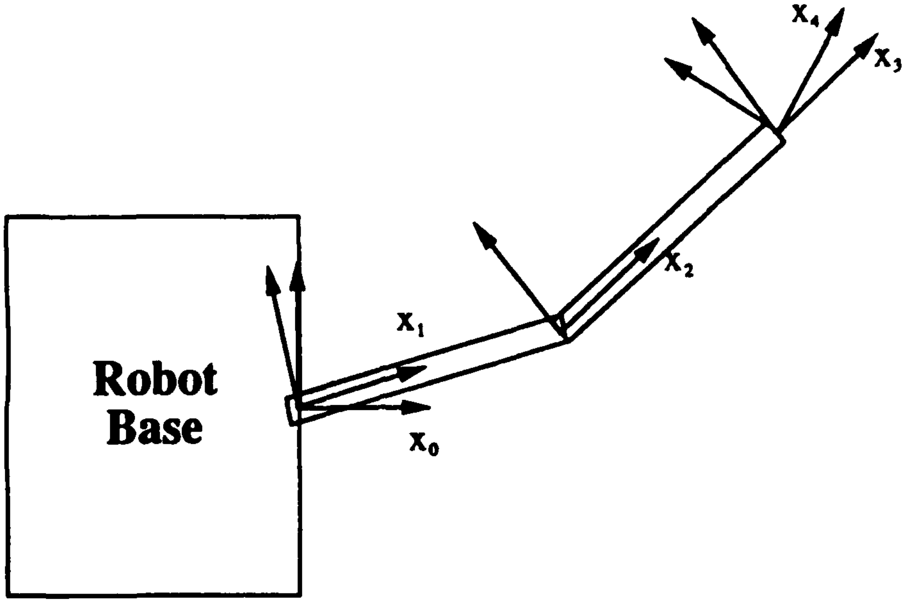

Figure 2.2: Schematic showing Frames 0 through 4

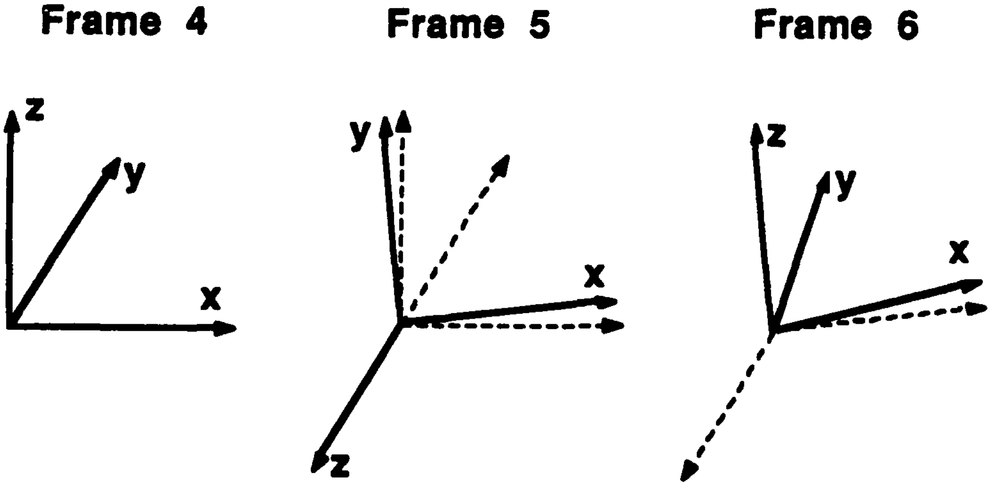

Figure 2.3: Schematic showing Frames 4 through 6 (dotted indicates previous frame)

Frame origin The intersection of the shoulder axis and the horizontal plane that contains the forearm (distil link) centerline.

X+ Forward

Y+ To the left

Z+ Vertical, upward

Robot frame 1 Robot upper arm frame

Frame origin The intersection of the shoulder axis and the horizontal plane that contains the forearm (distil link) centerline.

X+ In horizontal plane, pointing from the frame origin to the elbow joint axis.

Y+ In horizontal plane, pointing to the left from the frame's X+.

Z+ Vertical, upward.

Robot frame 2 Robot forearm frame

Frame origin The intersection of the elbow axis and the horizontal plane that contains the forearm (distil link) centerline.

X+ In horizontal plane, along the forearm centerline, pointing from the origin toward the end of the manipulator.

Y+ In horizontal plane, pointing to the left from the frame's X+.

Z+ Vertical, upward.

Robot frame 3 Robot vertical stage frame

Frame origin The wrist point of the manipulator

X+ Always parallel to X+ of frame 2.

Y+ Always parallel to Y+ of frame 2.

Z+ Always parallel to Z+ of frame 2. (vertical, upward.)

Note that this the vertical stage of the robot is a pure translation, thus the coordinate axes remain parallel to those of the previous frame.

Robot frame 4 Robot wrist barrel frame

Frame origin The wrist point of the manipulator

X+ As marked on the wrist barrel: (X4+ is always in the horizontal plane, and always perpendicular to Z5. It aligns with X3+ when $qj_4 = 0$.)

Y+ As marked on the wrist barrel: (Y4+ is always in the horizontal plane and parallel to Z5; it aligns with Y3+ when $qj_4 = 0$.)

Z+ Vertical, upward.

Robot frame 5 Robot wrist yoke frame

Frame origin The wrist point of the manipulator

X+ As marked on the wrist yoke: always in the X4–Z4 plane, it aligns with X4+ when $qj_5 = 0$

Y+ As marked on the wrist yoke: always in the X4–Z4 plane, it aligns with Z4+ when $qj_5 = 0$

Z+ As marked on the wrist yoke; always opposite of Y4+

Robot frame 6 Robot wrist output frame

Frame origin The wrist point of the manipulator

X+ As marked on the wrist output: always in the X5–Z5 plane, it aligns with X5+ when $qj_6 = 0$

Y+ As marked on the wrist yoke: always in the X5–Z5 plane, opposite Z5+ when $qj_6 = 0$

Z+ As marked on the wrist yoke: always parallel to Y5+

2.2.1 Forward Kinematics from Motor to Joint Space

Now that we have defined the link frames, and through them the joint coordinates of these manipulators, we can present the first part of the forward kinematics, the mapping that takes us from a known set of motor coordinates and gives us the joint coordinates. Note that we have appended a coordinate for the gripper at the end of these vectors, so that the configuration is described by seven coordinates.

$$\begin{bmatrix} q\dot{j}_s \\ q\dot{j}_e \\ q\dot{j}_z \\ q\dot{j}_y \\ q\dot{j}_p \\ q\dot{j}_r \\ q\dot{j}_g \end{bmatrix} = \begin{bmatrix} \frac{1}{ns} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{-1}{ns} & \frac{1}{ne} & 0 & 0 & 0 & 0 & \\ 0 & 0 & \frac{1}{nz} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{n4} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-1}{n5} & \frac{-1}{n6} & 0 \\ 0 & 0 & 0 & \frac{1}{n4} & \frac{2}{n5} & \frac{2}{n6} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{ngrip} \end{bmatrix} \begin{bmatrix} qm_1 \\ qm_2 \\ qm_3 \\ qm_4 \\ qm_5 \\ qm_6 \\ qm_7 \end{bmatrix} \tag{2.1}$$

Where the geometric scale factors (usually gear ratios, except for the linear joints) are:

ns = 9.0
ne = 3.0
nz = 4947.380111 radians/meter
n4 = 250
n5 = 360
n6 = 360
ngrip = 2782.901312 radians/meter

2.2.1.1 Link Transforms

Having defined a series of frames, one for the robot base, and one for each degree of kinematic freedom, we can now derive the homogeneous transform from each one to the next, given the general form of the link transform, and the relevant kinematic data in the form of Denavit–Hartenberg notation, as defined in [2]. Craig defines a standard form of this transform in his equation [3.6, his numbering.]

He also assembles the kinematic parameters into a table, such that the quantities needed to write the transform from frame i to the next frame all appear on line i. We can make the process straightforward and easy to automate by making this table into a matrix with the number of rows equal to the number of degrees of freedom, and four columns for the Denavit–Hartenberg parameters (in the order $\alpha_{i-1}, a_{i-1}, d_i, qj_i$.) We denote this matrix as KP, for kinematic parameters. We can then write the general form of the link transform as follows:

$${}^{i-1}_{\ i}T = \begin{bmatrix} \cos(KP_{i,4}) & -\sin(KP_{i,4}) & 0 & KP_{i,2} \\ \cos(KP_{i,1})\sin(KP_{i,4}) & \cos(KP_{i,1})\cos(KP_{i,4}) & -\sin(KP_{i,1}) & -KP_{i,3}\sin(KP_{i,1}) \\ \sin(KP_{i,1})\sin(KP_{i,4}) & \sin(KP_{i,1})\cos(KP_{i,4}) & \cos(KP_{i,1}) & KP_{i,3}\cos(KP_{i,1}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

We can then substitute in the appropriate entries from KP to get the six transforms we are interested in having. It is desirable to use MACSYMA to do this, both to eliminate the possibility of algebraic error, and to have symbolic help in simplifying the overall transform, which is a product of six such matricies and very complicated looking before simplification.

Link Parameters and Link Transforms The KP matrix is given as follows. The joint coordinates are $qj_1, qj_2, d_3, qj_4, qj_5$, and qj_6. Angles are given in radians. We have left l_1 and l_2 as symbols for generality, even though they are constants and are equal (16 inches = 40.64 cm .) Note [illegible] number of zero terms; these lead to a simple kinematic solution for the manipulators, even though they have all the geometric freedom that we require.

$$KP = \begin{bmatrix} 0 & 0 & 0 & qj_1 \\ 0 & l_1 & 0 & qj_2 \\ 0 & l_2 & d_3 & 0 \\ 0 & 0 & 0 & qj_4 \\ \pi/2 & 0 & 0 & qj_5 \\ -\pi/2 & 0 & 0 & qj_6 \end{bmatrix} \tag{2.3}$$

Substituting as mentioned above, we can obtain the first link transform, from frame 0 to frame 1:

$${}^0_1T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

Where c_i stands for $\cos(qj_i)$, s_{ij} stands for $\sin(qj_i + qj_j)$, etc. — a convention that we adopt.

In the same manner, we write the transform from frame 1 to frame 2:

$${}^1_2T = \begin{bmatrix} c_2 & -s_2 & 0 & l_1 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

The transform from frame 2 to frame 3:

$${}^2_3T = \begin{bmatrix} 1 & 0 & 0 & l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.6}$$

The transform from frame 3 to frame 4 is:

$${}^3_4T = \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.7}$$

The transform from frame 4 to frame 5:

$$ {}^{4}_{5}T = \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8} $$

The transform from frame 5 to frame 6:

$$ {}^{5}_{6}T = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.9} $$

2.2.2 Forward Kinematics

Now that we have the six individual transforms, we can begin to multiply them together to obtain the forward kinematics of increasingly complete fractions of the total manipulator. We present the results in this fashion in order to show that the geometry we have chosen results in a much simpler form than that of many other robots.

The keys to this are to remember that our geometry maintains the first four joint axes parallel, and that the upper 3 by 3 submatrix of the overall transform specifies the relative orientation between the frames (as direction cosines.) Therefore, the rotations about axes 1, 2, and 4, are all about a parallel axis (joint 3 is a linear translation and doesn't effect orientation.) We expect to be able to simplify the rotation parts of the forward kinematics up to frame 4 to a single rotation about vertical. This is the case, and the kinematics are much simpler because of it.

This is the transform from frame 0 to frame 2, after simplifying using the sum of angles identities:

$$ {}^{0}_{2}T = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1c_1 \\ s_{12} & c_{12} & 0 & l_1s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.10} $$

We can now multiply out the transform from frame 0 to frame 4 and again simplify using the sum of angles identities:

$$ {}^{0}_{4}T = \begin{bmatrix} c_{124} & -s_{124} & 0 & l_2c_{12} + l_1c_1 \\ s_{124} & c_{124} & 0 & l_2s_{12} + l_1s_1 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.11} $$

This is the complete forward kinematic transform from frame 0 to frame 6:

$$
{}^{0}_{6}T = \begin{bmatrix} c_5c_6c_{124} - s_6s_{124} & -c_6s_{124} - c_5s_6c_{124} & -s_5c_{124} & l_2c_{12} + l_1c_1 \\ c_5c_6s_{124} + s_6s_{124} & c_6c_{124} - c_5s_6s_{124} & -s_5s_{124} & l_2s_{12} + l_1s_1 \\ s_5c_6 & -s_5c_6 & c_5 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.12}
$$

It should be noted that the simplification to this form is based on use of knowledge about the geometry. Attempts to use MACSYMA in a brute force manner, without taking advantage of the structure of the system — the command TRIGREDUCE(T1.T2.T3.T4.T5.T6) — results in an expression that is more complicated than this result (in terms of number of trig evaluations) and possibly more complicated than the original. Thus, symbolic methods are not a substitute for insight into the problem, they are a tool whose use is still something of an art.

Comparison to the PUMA 560 A brief comparison to the forward kinematics of a PUMA 560 is instructive. Craig derives these in his book as eq. 3.14.; we do not need to reproduce them here. Even in factored form, the most complex element of the PUMA's forward transform requires the evaluation of 11 different trigonometric terms; the most complex for our arms require only five. This overstates the case , as many of the trig terms are reused in different numbers in the two different kinematics we are comparing, and the reuse of trig terms makes the advantage (measured solely in terms of trigonometry) considerably less. However the complexity in terms of other arithmetic operations still favors our arm substantially. The same PUMA term has 15 other arithmetic operations; ours has four. Again, there is more reuse of common subexpressions in the PUMA solution, which can reduce computation time if these can be stored and later recalled faster than recomputing them. Our purpose is not to criticize the designers of the PUMA (some of whom have given us valuable guidance on this and other research.) But, overall, we have succeeded in developing a pair of manipulators whose geometry is much simpler to deal with in realtime control, enabling us to concentrate our efforts on the dynamics (which are also simplified for the same reasons) and control aspects of cooperation and flexibility.

Solving for the Cartesian Vector Although this is a correct formal representation, we are more interested in solving for the cartesian representation of the configuration vector, given the joint vector, and vice versa. Thus, we do not usually need to represent orientation by the 3 by 3 matrix of direction cosines, except perhaps as an intermediate step in calculations. We can pull out of the transform the terms that enable us to solve for the cartesian description of the arm's configuration given the joint space description. Taking the terms in the last column of transform matrix in 2.12, we have:

$$
qc_X = lc_{12} + lc_1 = l(c_{12} + c_1) \tag{2.13}
$$

$$qc_y = ls_{12} + ls_1 = l(s_{12} + s_1) \tag{2.14}$$

where we have made the previously noted simplification $l = l_1 = l_2$

$$qc_Z = qj_z = qj_3 \tag{2.15}$$

Given our stated definition of orientation in the cartesian vector as a series of rotations that begin aligned with the rererence frame, then we can determine the three roll, pitch, and yaw angles in cartesian coordinates directly, without going through intermediate direction cosine matricies. Yaw in cartesian coordinates is the sum of the revolute joint angles between the base frame and the main wrist barrel. All of these rotations are about a vertical axis, and all have the same sense of sign. So cartesian yaw is the sum of the shoulder joint angle, the elbow joint angle, and the wrist yaw angle (the vertical translation of joint three does not effect yaw.)

$$qc_y = qj_s + qj_e + qj_y = qj_1 + qj_2 + qj_4 \tag{2.16}$$

Given our choice of orientation description in the two coordinate systems, pitch and roll (and gripper opening, as well) translate across identically:

$$qc_p = qj_p \tag{2.17}$$

$$qc_r = qj_r \tag{2.18}$$

$$qc_g = qj_g \tag{2.19}$$

2.2.3 Inverse Kinematics

We now consider the problem of inverse kinematics. The first aspect of the inverse kinematics presented here is the solution for the robot's configuration in joint space, given its configuration in the cartesian base frame (frame zero as defined previously. There is one additional piece of information that we need in order to make this relationship unambiguous: we need to specify whether we want a right-handed or left-handed solution. Left-handed coresponds to $qj_2 < 0$, right handed coresponds to $qj_2 \geq 0$. The inverse kinematic solution is best done, term by term in sequence. We begin by applying the law of cosines to the triangle formed by the first two links (the third side is the line from shoulder joint to the end of the second link.)

$$c_2 = \cos(qj_2) = \frac{qc_1^2 + qc_2^2}{2l^2} - 1 \tag{2.20}$$

The law of cosines calculation is simplified because we have deliberately made $l_1 = l_2 = l$(for notation)- in implementation, $2l^2$ is a precomputed constant. We then calculate qj_2, using the arccosine, and calculate the sine, for use later.

$$qj_2 = \pm \cos^{-1}(c_2) \tag{2.21}$$

where positive means a right handed solution, negative is left handed.

$$s_2 = \sin(qj_2) = \sqrt{1 - c_2^2} \tag{2.22}$$

We must then handle some special cases to get the shoulder angle right under all reachable conditions. when $qc_x = 0$, we would be dividing by 0 when we attempt to calculate the tangent of the shoulder angle, thus we have two special cases:

$$qj_1 = \pi - \frac{qj_2}{2},\ for\ qc_x = 0,\ and\ qc_y \geq 0 \tag{2.23}$$

$$qj_1 = -\pi - \frac{qj_2}{2},\ for\ qc_x = 0,\ and\ qc_y < 0 \tag{2.24}$$

$$qj_1 = \tan^{-1}(\frac{qc_Y}{qc_X}) - \frac{qj_2}{2},\ otherwise \tag{2.25}$$

Since vertical position of the wrist point is the same in cartesian and joint spaces, the third term of the inverse kinematics is available by inspection,

$$qj_z = qj_3 = qc_3. \tag{2.26}$$

The fourth term, (yaw) is simple, because all of the rotations so far have been about the same vertical axis. So, the yaw coordinate is:

$$qj_4 = qj_y = qc_y - (qj_s + qj_e). \tag{2.27}$$

Given our choice of representation for orientation in the cartesian configuration, we have simple term by term equalities between the joint and cartesian vectors for the fifth through seventh terms.

$$qj_5 = qj_p = qc_p \tag{2.28}$$

$$qj_6 = qj_r = qc_r \tag{2.29}$$

$$qj_7 = qj_g = qc_g \tag{2.30}$$

2.2.4 The Jacobian Matrix

The Jacobian matrix can now be calculated from the kinematic relations we have. The Jacobian is required for many control algorithms. It can be thought of as an instantaneous linear mapping from joint velocities to cartesian velocities (or equivalently from joint forces/torques to cartesian forces/torques.) Our approach is to solve for the cartesian velocities on a term by term basis, and then assemble these six equations into matrix form with the variables in the desired order. The first three cartesian quantities, the linear velocities in the x, y, and z directions in the base frame, can be found by taking the derivative of the last column of ${}^0_6T$, which expressed the forward kinematic transform. The velocities (or rates) are referred to as u's, with the rest of the nomenclature for the three principal coordinate systems similar to that for the q's except that the cartesian orientation rates are defined and named slightly differently: The motor space velocity vector is defined to be the time derivative of the motor space configuration vector, and the joint space velocity vector is defined analogously:

$$\mathbf{um} = \dot{\mathbf{qm}} \tag{2.31}$$

$$\mathbf{uj} = \dot{\mathbf{qj}} \tag{2.32}$$

The cartesian velocity vector is *not* simply the time derivative of the (base frame) cartesian configuration vector. Following the convention of [2], we define the (base frame) cartesian velocity vector as follows.

$$\mathbf{uc} = \begin{bmatrix} {}^0v \\ {}^0\omega \\ q\dot{c}_g \end{bmatrix} = \begin{bmatrix} uc_X \\ uc_Y \\ uc_Z \\ uc_{\omega X} \\ uc_{\omega Y} \\ uc_{\omega Z} \\ uc_g \end{bmatrix} \tag{2.33}$$

where ${}^0v$ is the cartesian velocity (in the base frame) of frame six's origin and ${}^0\omega$ is the angular velocity of frame six, expressed in the base frame and uc_g is the gripper separation velocity. This follows [2] equation 5.36. The first three components in uc *are* the time derivatives of the coresponding terms in qc, so the first three components can be found by straightforward differentiation of the term by term forward kinematics for position, 2.13 through 2.15. Performing these differentiations, we get the following three equations:

$$\begin{aligned} uc_X &= -L(s_1 uj1 + s_{12}(uj1 + uj2)) \\ uc_Y &= +L(c_1 uj1 + c_{12}(uj1 + uj2)) \\ uc_Z &= uj3 \end{aligned} \tag{2.34}$$

In matrix form, the first three DOF are independent of the latter three, leaving us with:

$$\begin{bmatrix} uc_X \\ uc_Y \\ uc_Z \end{bmatrix} = \begin{bmatrix} -L(s_1 + s_2) & -L(s_{12}) & 0 \\ L(c_1 + c_1 2) & -L(c_{12}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} uj_1 \\ uj_2 \\ uj_3 \end{bmatrix} \tag{2.35}$$

Obtaining the equations for the last three cartesian u's, namely $uc_{\omega X}$, $uc_{\omega Y}$, and $uc_{\omega Z}$, are more complex. These quantities are the last link's angular velocity, resolved into components about the unit vectors of the base frame. Note that these are *not* simply the derivatives of the last three coordinates. To get the desired expressions, we must first compute the angular velocity of the last link (and this calculation is best done in that body's attached frame), and then transform that back into the base frame. The angular velocity is not merely the derivatives of the joint rates, instead, it can be computed by propagating the angular velocity from link to link by applying Craig 5.16 once for each link. The result is the angular velocity of the last (sixth) link, resolved into components of its attached frame:

$${}^{6}\omega^{6} = \begin{bmatrix} s_5 c_6 uj_{124} - s_6 uj_5 \\ -c_6 uj_5 - s_5 s_6 uj_{124} \\ uj_6 + c_5 uj_{124} \end{bmatrix} \tag{2.36}$$

Now we want to express this quantity in the base frame, so we multiply by ${}^{0}_{6}R$, which is the upper 3 by 3 matrix in ${}^{0}_{6}T$. After expanding all of the terms and simplifying using trig identities, we get

$${}^{0}\omega^{6} = \begin{bmatrix} s_{124} uj5 - s_5 c_{124} uj6 \\ -s_5 s_{124} uj6 - c_{124} uj5 \\ c_5 uj6 + uj_{124} \end{bmatrix} \tag{2.37}$$

where $uj_{124} = uj_1 + uj_2 + uj_4$.

We can now write the equation for cartesian velocity as a function of joint velocity in matrix form by combining equations 2.35 and 2.37.

$$\mathrm{uc} = {}^{\mathrm{c}}_{\mathrm{j}}\mathrm{J}\,\mathrm{uj} \tag{2.38}$$

where

$${}^{c}_{j}J = \begin{bmatrix} -L(s_1 + s_{12}) & -L(s_{12}) & 0 & 0 & 0 & 0 & 0 \\ +L(c_1 + c_{12}) & -L(c_{12}) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_{124} & -c_{124}s_5 & 0 \\ 0 & 0 & 0 & 0 & -c_{124} & -s_{124}s_5 & 0 \\ 1 & 1 & 0 & 1 & 0 & c_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.39}$$

The Jacobian has some important structural features that have bearing on the complexity and robustness of control algorithms. The matrix is almost block diagonal, and the blocks are small (one 3 × 3, one 2 × 2, and two 1 × 1 blocks). The only terms that are not block diagonal are the first two ones in the next to the bottom row. These terms express the fact that the shoulder and elbow joint rates add directly to the last link's angular velocity about the base frame Z axis. This is a simple one way coupling. The important results of this structure are twofold. The Jacobian, being fairly sparse and simple, can be computed rapidly. Secondly, we can easily see when the Jacobian will go singular (lose rank). First we consider the uppermost 2 × 2 block, namely

$$\begin{bmatrix} -L(s_1 + s_{12}) & -L(s_{12}) \\ +L(c_1 + c_{12}) & -L(c_{12}) \end{bmatrix}$$

, if we let $qj_2 = 0$ or $qj_2 = \pm\pi$, then this submatrix becomes singular. Physically, we can understand these cases as follows: When $qj_2 = 0$, the arm is straight out, with the wrist point as far out from the base frame origin as it can go; in this configuration, incremental motions of either the shoulder joint or the elbow joint will move the wrist point from side to side, but will cause no motion normal to the workspace boundary (in or out). Similarly, when $qj_2 = \pm\pi$, the manipulator is folded back on itself, and incremental motions of the shoulder or elbow cannot cause motion normal to the workspace boundary. In practice, the elbow is prevented from reaching $\pm\pi$, by mechanical limit stops, in order to protect the wrist mechanism, so this case does not occur physically. These cases are called workspace boundary singularities. Next we consider the 3 × 3 block, namely

$$\begin{bmatrix} 0 & s_{124} & -c_{124}s_5 \\ 0 & -c_{124} & -s_{124}s_5 \\ 1 & 0 & c_5 \end{bmatrix}$$

, when $qj_5 = 0$ or $qj_5 = \pm\pi$, then this submatrix becomes singular. Again, the $\pm\pi$ is outside the limit stops on the mechanism, and is not a problem. However, the case of $qj_5 = 0$ is *precisely* the reference configuration for the fifth joint. When $qj_5 = 0$, the fourth and sixth joint axes are aligned with each other (both vertical); this condition is called a workspace interior singularity (also known as gimbal lock, a term from gyroscopes, which also have the same mechanism geometry as our wrist). In this configuration, the robot loses one degree of orientation freedom (cannot rotate about the ${}^4X$ axis), and gains redundancy in another (joints 4 and 6 are both vertical). Special exception handling software must take this limitation of the mechanism into account, and the human or AI system that performs the task planning must be aware of, and properly plan for, this situation. This problem could be sidestepped by limiting the range of motion of the fifth axis to $0 < qj_5 < \pi$ or $0 > qj_5 > -\pi$, but this would be artificially discarding half of the mechanism's reachable orientations, a gross limitation of the dexterous workspace. This problem is fundamental to this class of wrists, and has been a problem for many 6 degree-of-freedom manipulators.

2.3 Identification and Control Experiments

In this section, we present the principal results from the identification and control experiments performed using the cooperating flexible-drive manipulators. The kinematic parameters (link parameters and gear ratios) have been presented in the previous section. In the sections below, we present the results of experiments designed to identify important aspects of the dynamic model, namely mass properties and friction-flexibility properties of the manipulators.

2.3.1 Mass Property Identification Experiments

Knowledge of the mass properties is essential to an accurate dynamic model of the manipulator. The mass properties, together with the kinematic model, are the heart of the ideal mechanism dynamics – the dynamics of a manipulator with absolutely no friction or flexibility. The properties that we need are the mass, location of the center of mass, and the inertia properties (about the mass center) for each link. Often, due to the geometry of a manipulator, some of this information is not used in the model, and is thus not needed. For example, the first four links of our manipulator rotate *only* about vertical axes; thus they never experience angular accelerations about any other axis, and the only relevant inertia properties are their inertias (with respect to their respective mass centers) about their vertical axes. A table of the most important parameters follows:

Assembly name	Mass Center Location (M.)	Mass (Kg.)	Inertia Term	Inertia (KgM^2)
Upper Arm	X_1 +0.150	4.902	I_{zz}	0.141
Forearm	X_2 +0.225	6.345	I_{zz}	0.222
Gripper	n/a	0.6105	I_{zz}	0.001749

The actuator scale factors (in our case, these are all motor torque constants) together with the geometric scale factors, and the scale factors associated with the interface electronics, define the corespondence between commands from the computer to torques generated at the motors (exclusive of friction). The scale factors for both the actuators and the associated electronics are presented here. The interface electronics also impose current (and thus torque) limits on the actuators; for the shoulder and elbow actuators, these have two settings: a low power setting for safe testing of controllers and a full power setting. These values are tabulated below.

Actuator	Torque const. (N-m/Amp)	Amplifier const. (Amps/volt)	Max. Torque (N-m)	Low Power Torque (N-m)
Shoulder	0.941	1.00	9.41	1.88
Elbow	0.941	1.00	9.41	1.88
Z Axis	0.0224	0.50	0.112	0.112
Yaw	0.0224	0.20	0.0447	0.0447
Roll (motor 5)	0.0224	0.20	0.0447	0.0447
Pitch (motor 6)	0.0224	0.20	0.0447	0.0447
Gripper	0.0224	0.20	0.0447	0.0447

Actuator scale factors are the nominal values provided by the manufacturers; the amplifier scale factors were measured experimentally, as were the peak torques at both maximum and the low power settings.

2.3.2 Flexibility and Friction Identification Experiments

Identification and control experiments have been performed on the manipulator, with particular emphasis on understanding the effects of joint flexibility on the system's performance. One of the foundations of our approach to cooperative control of *flexible drive* manipulators is our ability to model and control the torque delivered to each link through the flexible drive system. The subsystem of actuator, flexible drivetrain, and link is modeled as a free, free, mass-spring-mass system (while not in contact with anything in the environment [6]). Thus, in our identification and initial control experiments, we look for evidence of response dominated by second order, lightly damped behavior. In the experimental data shown in figure 2.3.2, we see clear evidence that the elbow subsystem's behavior coresponds to our model's prediction. Additionally, we see the evidence of some nonlinearities – friction (and/or cogging) – having a substantial effect on the amplitude response. Thus we see clearly the need for more research.

2.4 Continuing Research

Research on cooperating, flexible drive manipulators will continue under DARPA contract DAAA21-89-C-0002, with principal investigators Professor Robert H. Cannon, Jr. in the department of Aeronautics and Astronautics and Professor Jean-Claude LaTombe in the department of Computer Science. Under this new contract, research into the dynamics and control of cooperating, flexible manipulators will continue, and will be integrated with research on motion and task planning. The continuing work in dynamics and control(s) will emphasize achieving levels of dynamic control that will permit the demonstration of good solutions to previously intractable problems for automation.

The new contract emphasizes research in three main areas: (1) the specific technologies that are critical to achieving true, effective integrated cooperation between

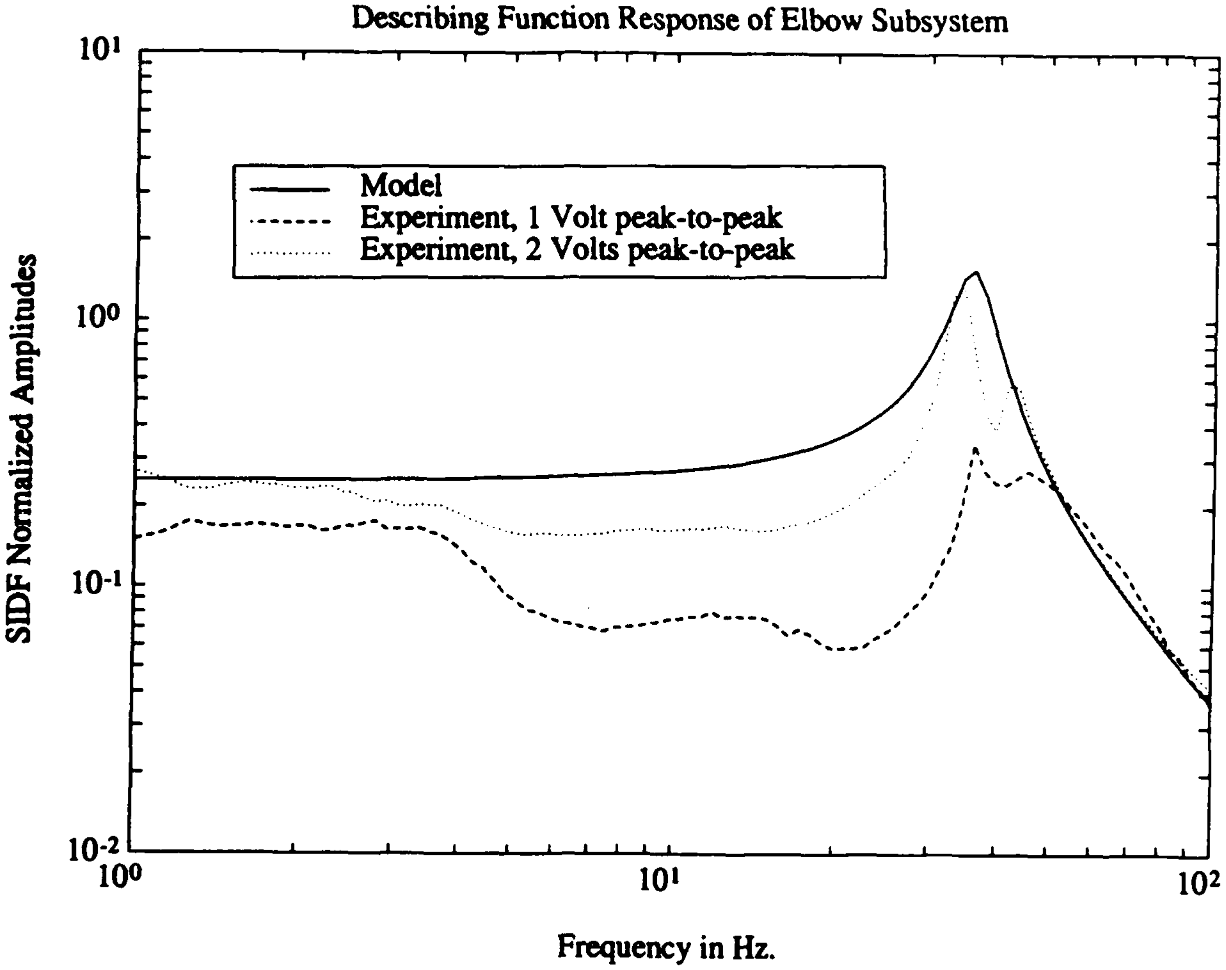

Figure 2.4: Model and experimental response to a sinusoidal input: describing function data for the elbow subsystem excited by a 2.0 and 1.0 volts peak-to-peak input. The solid line is the model, the dotted line the data at 2.0 V., and the dashed line the data at 1.0 V.

compliant manipulators; (2) motion planning and execution, for both gross and fine maneuvering of awkward objects amidst obstacles, so that users can specify object motions *per se* at a high level; and (3) integration of the path planning and manipulation functions. Specifically, we will demonstrate experimentally the intimate vertical integration of path planning with the manipulative skills of a pair of deft, cooperating robotic arms in the rapid assembly of a complex configuration of awkward objects, all in response to only high-level instructions.

Chapter 3

Control of a Flexible Robot with a Mini-Manipulator

Raymond Kraft

3.1 Introduction

Present day industrial robots are slow and massive. Their motions are controlled by inferring end-point position from joint angle measurements, and end-point force from joint torque measurements. This is known as collocated control; that is, sensing and actuation take place at the same physical location. Naturally, if there is any flexibility in the system — something present in all real physical systems — the actual end-point position and force will not agree with what is predicted by the joint angles and torques. Consequently, robots have been designed to be very rigid, and hence very heavy, so as to minimize errors introduced by flexibility.

The obvious solution to this problem is to directly measure those quantities that are to be controlled. Namely, to measure the end-point position and force, and to feed these measurements back to a controller that drives the actuators. Given such a controller, this scheme would eliminate flexibility induced errors, and would hence enable one to use lighter-weight, flexible, robots. There is, however, a good reason why this approach has not been used in the past: it poses a much more difficult stability problem. Where it is easy to achieve stable, albeit slow, control through the use of collocated control, it is non-trivial to achieve good, stable control using end-point feedback.

While end-point feedback increases the speed and precision of manipulation, there is still a fundamental problem inherent in controlling robots that are designed to operate over a large workspace. Such a robot must obviously be large in size, and consequently it becomes difficult to achieve rapid, precise end-point control using an actuator that is far removed from the end-point. This difficulty may be attributed to two factors: (1) fine end-point motions of a large manipulator translate into yet finer joint angle motions, and (2) the flexibility in the manipulator limits end-point control bandwidth

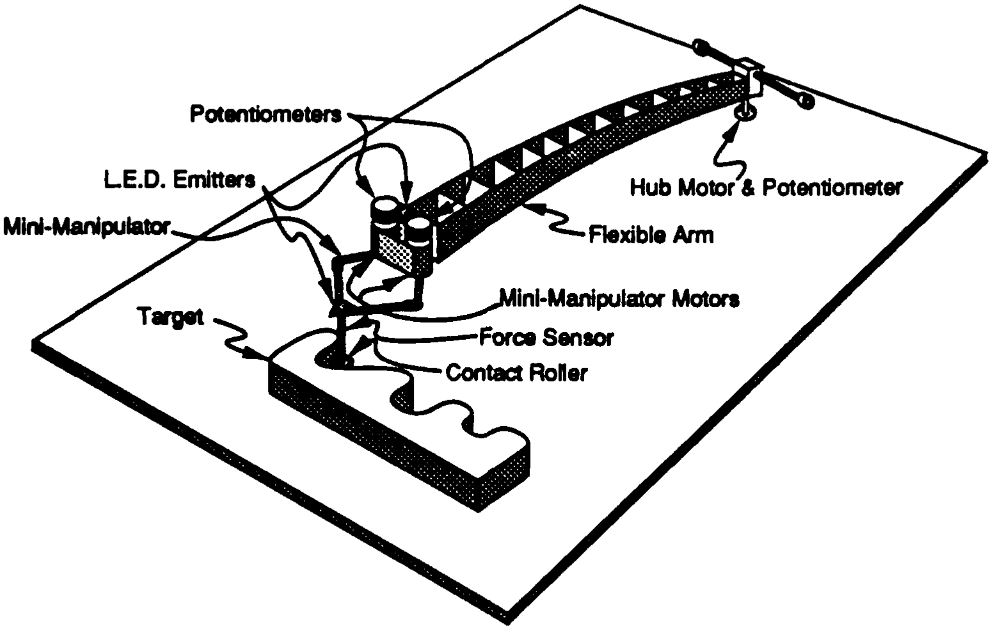

Figure 3.1: The Experimental System

to the time it takes a bending wave to travel the length of the manipulator.

A means of sidestepping this problem is to equip the manipulator with a small, lightweight, rigid mini-manipulator located at the end-point of the main manipulator. The mini-manipulator is then used for rapid, precise control within a small workspace, while the main manipulator is used to ferry the mini-manipulator between widely separated workstations.

Thus, by using end-point control in conjunction with a mini-manipulator, a lightweight robot would be able to achieve high bandwidth, precise control over a large workspace. It is these considerations which motivated the research on position and force control of a flexible robot arm with a two-degree-of-freedom mini-manipulator.

3.2 Experimental Apparatus

Figure 3.1 depicts the experimental system. It is comprised of the following major subsystems: the flexible arm, the mini-manipulator, actuators, sensors, and a target. The entire manipulator system operates in the horizontal plane with the end-point of the flexible arm following a circular arc.

3.2.1 The Flexible Arm

The flexible arm has distributed flexibility and was intentionally designed to exaggerate structural flexibility in the horizontal plane while remaining stiff in the gravity influenced vertical direction. The purpose of exaggerating the horizontal flexibility in this system was two fold: (1) to make problems associated with flexibility more readily apparent, and (2) to lower the resonant frequencies of the flexible arm into a region where reasonable sampling rates, in the neighborhood of 25 to 100 Hz, could be used. The flexible arm consists of a hub inertia block, two thin aluminum plates as side-rails, and fourteen bridges which separate the side-rails and are connected to them by beryllium copper springs. The flexible arm and the hub inertia block are 1.04 meters long, weigh 3.1 kg, and have an inertia about the hub of 0.45 kg-m^2. The two side-rails are 1 mm thick, 6.6 cm wide, and are made of 2024-T3 grade aluminum. The main actuator for the flexible beam is the hub motor located at its base and below the table. A potentiometer is collocated with the hub motor and is used to measure the base angle of the flexible arm. A light emitting diode (LED) array, whose position can be sensed by a photodetector, is located at the end-point of the flexible arm.

3.2.2 The Mini-Manipulator

The mini-manipulator is attached to the end-point of the flexible arm along one of the side-rails. Essentially, it is a five link, closed kinematic chain — the base link being fixed to the flexible arm (see Fig 3.10 for a close-up view of the mini-manipulator). The base link measures 7.6 cm in length. Each of the two inboard links are 7.0 cm long and are rigidly attached to the two motors that drive the mini-manipulator. The two outboard links are 10.8 cm long. Located at the joint between the two outboard links is another LED array which is used to sense the position of the end-point of the mini-manipulator.

Below this LED array is a vertical, force sensing, aluminum beam. It is approximately 9.5 cm long and is equipped with strain gauges which provide a means of measuring end-point force. At the bottom end of the force beam is a circular contact roller, which allows the mini-manipulator to exert only normal forces on objects it comes into contact with. The entire five link chain, along with the force beam and contact roller, weighs approximately 0.14 kg.

3.2.3 Actuators

The hub motor is an Aeroflex TQ-52W limited angle, brushless DC torquer with a peak torque of 1.7 N-m and a linear range of ±45 degrees. The mini-manipulator motors are small, rare earth magnet, DC torquers with 0.15 N-m of peak torque.

Each of the motors is driven by a linear power amplifier that produces an output current proportional to an input voltage. These amplifiers have a bandwidth of about 1kHz, which allows them to be modeled as a pure gain for the flexible arm and mini-manipulator system.

3.2.4 Sensors

The hub angle is measured by a conductive plastic potentiometer mounted on the shaft of the hub motor. The hub potentiometer has a sensitivity of about 3.4 Volts per radian.

Angular rate is obtained by passing the signal of the potentiometer through an analog pseudo-differentiator whose transfer function is $\frac{64.7s}{s+64.7}$.

Position sensing is achieved through the use of an overhead photodetector which measures the Cartesian coordinates of a single light source. To enable the tracking of multiple targets, a time multiplexing scheme is used, where multiple LED arrays are sequentially flashed on and off at 1kHz. LED's mounted at the end-point of the flexible arm and at the end-point of the mini-manipulator provide a means of measuring these two end-point coordinates. A finely tuned analog notch filter is used to filter out 60 Hz noise present in the system. Additionally, these position measurements are passed through analog, single pole, low-pass filters to remove other high frequency noise. The cut-off frequencies for measurements of the end-point of the flexible arm and the mini-manipulator are 10 Hz and 40 Hz respectively. Overall, position sensing accuracy is roughly ±0.5 mm at a gain of about 0.1 Volts per centimeter.

The angular positions of the two inboard mini-manipulator links are measured by two conductive plastic potentiometers mounted on the shafts of the mini-manipulator motors. The potentiometers have a sensitivity of about 3.4 Volts per radian. Angular rates are obtained by passing the potentiometer signals through analog pseudo-differentiators with transfer function $\frac{30s}{s+300}$.

The forces applied to the end-point of the mini-manipulator are measured by an aluminum force beam. It is 9.5 cm long and has a 5mm square cross section. Semiconductor strain gauges are mounted on all four sides at the root of the force beam. Thus Cartesian components of force in a local reference frame can be measured by connecting opposing pairs of strain gauges in a bridge configuration, and amplifying the bridge output. The force signals are filtered by a third order Bessel type filter with poles at $s = -151.6$ and $s = -334.0 \pm 530.2j$. Overall, the force sensor has a gain of roughly 14 grams force per Volt, with a noise level of about 1.5 mV.

3.2.5 The Target

A target was used to demonstrate force control and surface following. It was cut from plexiglas on a Matsuura CNC machine. It is 11 inches long, 2.5 inches wide, and 1.5 inches thick. It offers a variety of complex contours from flat surfaces, to curved surfaces, to sharp corners.

3.2.6 The Real Time Computer

Real time control was achieved using a DEC PDP 11/83 mini-computer with an RT-11 operating system. The system has a floating point math board, 16 twelve-bit A/D converters, and 12 twelve-bit D/A converters. All control programs were written in FORTRAN 77 and PDP assembly language.

3.2.7 Identification Equipment

A Schlumberger Solartron model 1254 frequency response analyzer was used to identify the dynamics of the experimental system. The frequency response analyzer generates a sinusoidal signal which is used to drive the system's actuators. The resulting motions and forces are then measured by the analyzer, and their magnitudes and phases relative to the input are calculated.

3.2.8 Other Computing Equipment

Control law synthesis and simulation was done on both VAX and SUN workstations. The commercially available control packages MATRIX_X and PRO-MATLAB were used.

3.3 Modifying the Plant for Favorable Dynamics

One problem that was encountered in doing end-point control of the mini-manipulator was the presence of torsion in the flexible arm. The flexible arm had been designed to ideally be flexible only in the horizontal direction, and with earlier experiments that used similar flexible arms, no torsion related problems were encountered. However, the current two-degree-of-freedom mini-manipulator mounted at the end of the flexible arm resulted in the flexible arm having a much higher end-point mass and inertia than previous experimental arms. This in turn resulted in a much lower frequency for torsional oscillations. The frequency of the first torsion mode was approximately 10 Hz — just within the desired 10 Hz mini-manipulator control bandwidth.

Further complicating matters, it was discovered that the transfer function from mini-manipulator end-point force to mini-manipulator end-point position had a first resonance pole at a lower frequency than the first resonance zero — an unfavorable pole-zero configuration.

3.3.1 Favorable and Unfavorable Pole-Zero Patterns

Basically, the transfer function from mini-manipulator end-point force to mini-manipulator end-point position in the lateral direction is that of a double integrator plant with an added lightly damped torsional resonance. This, in addition to a simple lead compensator, constitutes the original, unfavorable configuration of the system. Figure 3.2 shows the root-locus for the "Unfavorable Configuration". Notice that the poles associated with the torsional resonance immediately move into the unstable right half-plane. The fact that the pole frequency is lower than the zero frequency make this system inherently difficult to control.

On the other hand, if the pole frequency is higher than the zero frequency, then one has the system shown in Fig. 3.2 labeled "Favorable Configuration". Notice that here the resonance pole moves into the stable left half-plane.

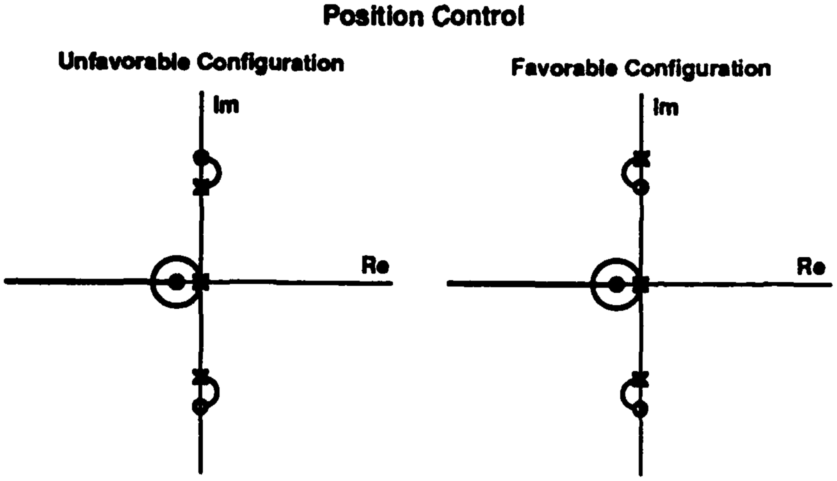

Figure 3.2: The Torsion Problem: Position Control Root-Locus

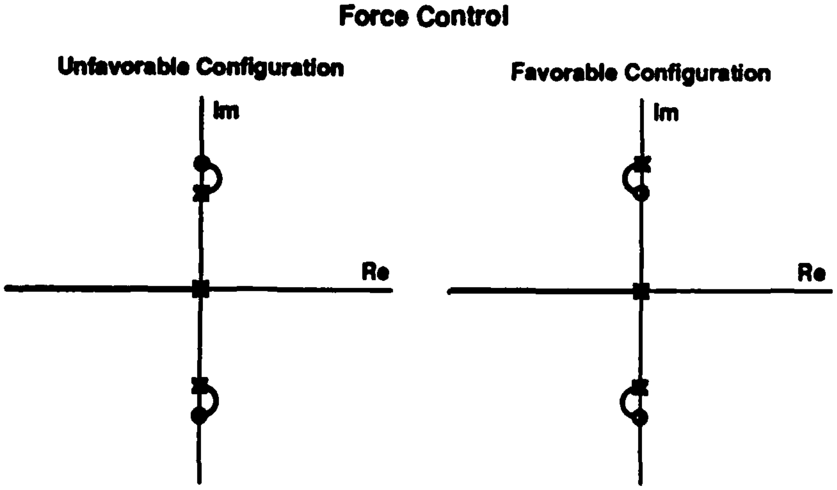

Figure 3.3: The Torsion Problem: Force Control Root-Locus

Similarly, in Fig. 3.3 we see the "Favorable" and "Unfavorable" pole-zero configurations for end-point force control of the mini-manipulator using integral control. When in force control, the plant no longer has a rigid body mode. The pole at the origin is due to integral control, and the two pairs of poles and zeros on the imaginary axis are due to the torsional resonance.

If the experimental system could be modified to cause the pole frequency to be greater than the zero frequency, the tasks of both position and force control would be greatly simplified. This possibility will now be investigated.

3.3.2 A Simplified Model of the Torsion Mode

A simplified model of the flexible arm and mini-manipulator system will be used to analyze the torsion mode. Figure 3.4 depicts this simplified model. It is essentially a view of the flexible arm looking from the end-point towards the hub. A dextral set of mutually perpendicular unit vectors $\boldsymbol{n}_1$, $\boldsymbol{n}_2$, and $\boldsymbol{n}_3$ are fixed in inertial reference frame N. Rigid body A, whose center of mass is located at C, has mass M and central inertia scalar I about an axis parallel to $\boldsymbol{n}_3$. Point C is located a distance h along the $\boldsymbol{n}_2$ direction from an inertially fixed point O. A dextral set of mutually perpendicular unit vectors $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, and $\boldsymbol{a}_3$ are fixed in a reference frame attached to A. A linear torsion spring with spring constant k is attached to A at C. This is intended to approximate the torsional characteristics of the flexible arm. A point mass located at P with mass m is constrained to move along groove G which is parallel to $\boldsymbol{a}_1$ and lies a distance d along the $\boldsymbol{a}_2$ direction from C. This is intended to approximate the mass of the end-point of the mini-manipulator. An LED located at L lies a distance l in the $\boldsymbol{a}_2$ direction from C. It is connected to a massless rigid link which is connected to point P. A contact point T lies a distance b in the $-\boldsymbol{a}_2$ direction from C. It is connected to a massless rigid link which is connected to point P.

This simplified system has three degrees of freedom described by three generalized coordinates: q_1, q_2, and q_3. Generalized coordinate q_1 is the angle between unit vectors $\boldsymbol{n}_1$ and $\boldsymbol{a}_1$. Generalized coordinate q_2 is the distance from O to C along the $\boldsymbol{n}_1$ direction. Finally, q_3 is the distance from C to T along the $\boldsymbol{a}_1$ direction. The generalized speeds are defined simply as $u_i \triangleq \dot{q}_i$ $(i = 1,2,3)$.

Three forces contribute to the motions of the simplified system. An internal force $F\boldsymbol{a}_1$ is exerted on P by body A. At the contact point T, an external force $R\boldsymbol{n}_1$ is exerted. Finally, an external torque $-kq_1\boldsymbol{a}_3$ is exerted on A by the linear torsion spring.

Given this system, the linearized equations of motion may be written as

$$\begin{aligned} -(I+md^2)\dot{u}_1 + md^2\dot{u}_2 + md\dot{u}_3 + Rb - kq_1 &= 0 \\ md\dot{u}_1 - (M+m)\dot{u}_2 - m\dot{u}_3 + R &= 0 \\ md\dot{u}_1 - m\dot{u}_2 - m\dot{u}_3 + F + R &= 0. \end{aligned} \tag{3.1}$$

When T is in contact with an object fixed in N, the following condition must hold

$$^{N}\boldsymbol{v}^{T} \cdot \boldsymbol{n}_1 = 0. \tag{3.2}$$

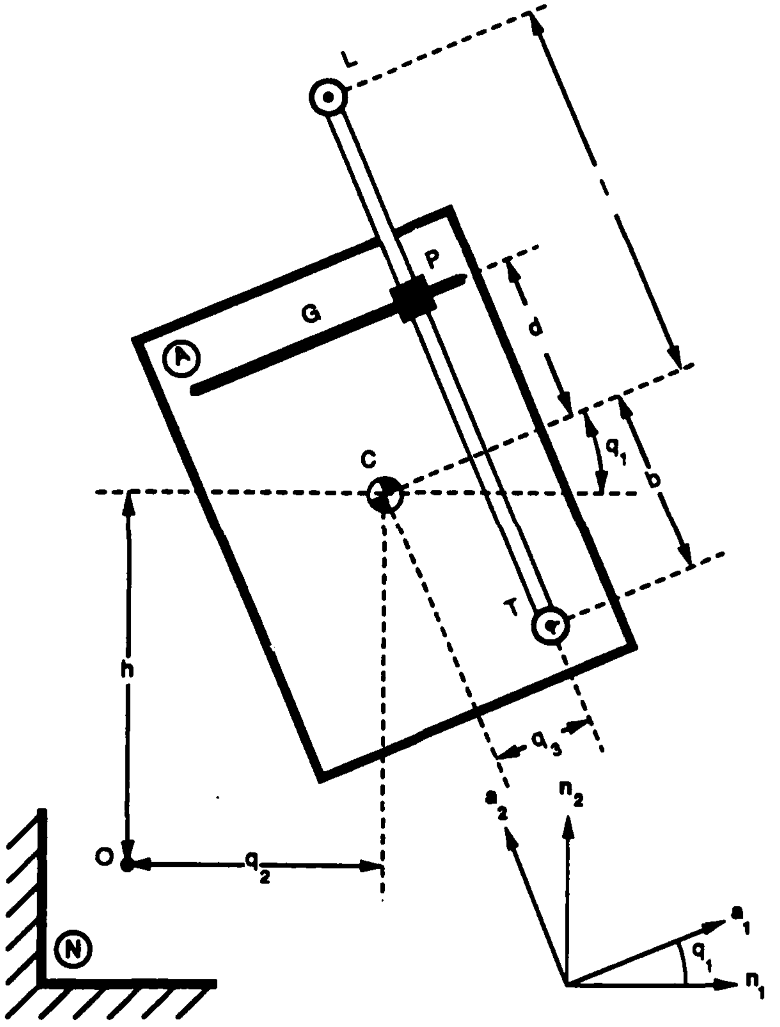

Figure 3.4: Simplified Torsion Model of the Experimental System

Simplifying and linearizing Eq. (3.2) yields

$$bu_1 + u_2 + u_3 = 0. \tag{3.3}$$

3.3.2.1 Position Control

When T is *not* in contact with an object, as is the situation during position control, the object reaction force R is zero. Making this substitution and taking the Laplace transform of Eqs. (3.1) yields

$$\begin{bmatrix} -(I+md^2)s^2 - k & mds^2 & mds^2 \\ mds^2 & -(M+m)s^2 & -ms^2 \\ mds^2 & -ms^2 & -ms^2 \end{bmatrix} \begin{bmatrix} q_1(s) \\ q_2(s) \\ q_3(s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} F(s). \tag{3.4}$$

Solving Eq. (3.4) yields

$$\begin{aligned} \frac{q_1(s)}{F(s)} &= \frac{d}{Is^2+k} \\ \frac{q_2(s)}{F(s)} &= \frac{-1}{Ms^2} \\ \frac{q_3(s)}{F(s)} &= \frac{\left[(\frac{1}{m}+\frac{1}{M})I + d^2\right] s^2 + k(\frac{1}{m}+\frac{1}{M})}{s^2(Is^2+k)}. \end{aligned} \tag{3.5}$$

During position control, the position of the LED at point L is measured. More specifically, one measures a quantity y given by

$$\begin{aligned} y &= \mathbf{r}^{OL} \cdot \mathbf{n}_1 \\ &= q_2 + \cos(q_1)\, q_3 - l \sin(q_1) \\ &\approx -lq_1 + q_2 + q_3. \end{aligned} \tag{3.6}$$

Substituting Eqs. (3.5) into Eq. (3.6) yields

$$\frac{y(s)}{F(s)} = \frac{[I + md(d-l)]\, s^2 + k}{ms^2(Is^2+k)}. \tag{3.7}$$

To achieve the "Favorable Configuration" illustrated in Fig. 3.2, where the frequency of the first resonance zero is less than that of the first resonance pole, Eq. (3.7) indicates that the following condition must hold

$$d(d-l) > 0. \tag{3.8}$$

This can be met by one of the two following conditions

$$d > 0 \text{ and } d > l \tag{3.9}$$

or

$$d < 0 \text{ and } d < l. \tag{3.10}$$

Enforcing the first condition is not easily feasible. By examining Fig. 3.4, one can see that the first condition implies that the LED is located below the point mass P. In this configuration, the point mass would most likely obscure the LED from the overhead vision system.

The second condition is, however, quite feasible. It requires that the point mass P be located below the center of mass of body A, and that the LED be above the point mass P. Thus, we have an easily obtainable system configuration that produces a favorable pole-zero pattern for position control.

3.3.2.2 Force Control

When in force control, the object reaction force R is non-zero and the condition described in Eq. (3.3) must hold. It is convenient to define the following variables:

$$\begin{aligned} z_1 &\triangleq -[I + md(d+b)] \\ z_2 &\triangleq m(d+b) + Mb \\ z_3 &\triangleq m(d+b). \end{aligned} \tag{3.11}$$

Under these circumstances, making the substitutions indicated above, and taking a Laplace transform, Eqs. (3.1) become

$$\begin{bmatrix} z_1 s^2 - k & 0 & b \\ z_2 s^2 & Ms^2 & 1 \\ z_3 s^2 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_1(s) \\ q_3(s) \\ R_3(s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} F(s). \tag{3.12}$$

Solving for $R(s)$ yields

$$\frac{R(s)}{F(s)} = \frac{k - z_1 s^2}{(z_1 - bz_3)s^2 - k}. \tag{3.13}$$

By examining Eq. (3.13), achieving a "Favorable Configuration" requires that

$$I + m(d+b)^2 < I + md(d+b) \tag{3.14}$$

or

$$(d+b)^2 < d(d+b). \tag{3.15}$$

Now there are two cases: one for $(d+b) < 0$, and one for $(d+b) > 0$. For each case, Eq. (3.15) translates into

Case 1: $(d+b) > 0$	Case 2: $(d+b) < 0$
$d + b < d$	$d + b > d$
$\Downarrow$	$\Downarrow$
$b < 0$	$b > 0$

Thus, there are two possibilities for favorable dynamics. The first is that

$$(d+b) > 0 \text{ and } b < 0. \tag{3.16}$$

However, given that for position control it is desirable to have $d < 0$, Eq. (3.16) is impossible to satisfy. The second possiblity is that

$$(d + b) < 0 \text{ and } b > 0. \tag{3.17}$$

But for $d < 0$, it would be very difficult to produce a physical geometry that would satisfy Eq. (3.17). Thus, for force control, the "Unfavorable Configuration" is practically unavoidable.

3.3.3 Summary

In summary, by configuring the system such that the LED is situated above the point P, and that P is in turn below C, one has a system with a favorable pole-zero pattern for position control. Unfortunately, this also results in an unfavorable configuration for force control, but this cannot be avoided. In physical terms of the actual mini-manipulator, this configuration of the simplified model translates into the following: the LED should be located above the center of mass of the mini-manipulator end-point, and the center of mass of the mini-manipulator end-point should be located below the center of mass of the flexible arm.

3.4 Collocated Control

Collocated control is a type of control where the sensors and actuators are located at the same physical position. This is significant because it results in robust, stable control. However, it has the drawbacks of not yielding good end-point accuracy and producing relatively low bandwidth control. Nevertheless, there are times when it is appropriate to use collocated control with the flexible arm and mini-manipulator system.

One such time is when the end-point of the flexible arm or the mini-manipulator is either outside the field of view of the optical end-point sensor, or occluded from the optical end-point sensor. In such instances, it is desirable to switch from high-performance, end-point control to the more dependable collocated control.

Another time when collocated control is appropriate is when the desired end-point position is outside the workspace of the mini-manipulator. Under this circumstance, it is desirable to use a collocated mini-manipulator controller while the flexible arm controller moves the mini-manipulator into range. During this motion, the collocated controller holds the mini-manipulator in some nominal configuration. Once the desired end-point position is in range, the mini-manipulator switches to end-point control.

The type of collocated control used is PD — Proportional and Derivative. More specifically, the control torques for each of the three actuators may be described by

$$\begin{aligned} T_H &= K_H \left[k_{pH}(q_{1_{command}} - q_1) + k_{dH}(0 - u_1)\right] \\ T_A &= K_A \left[k_{pA}(q_{n+1_{command}} - q_{n+1}) + k_{dA}(0 - u_{n+1})\right] \\ T_B &= K_B \left[k_{pB}(q_{n+1_{command}} - q_{n+2}) + k_{dB}(0 - u_{n+2})\right]. \end{aligned} \tag{3.18}$$

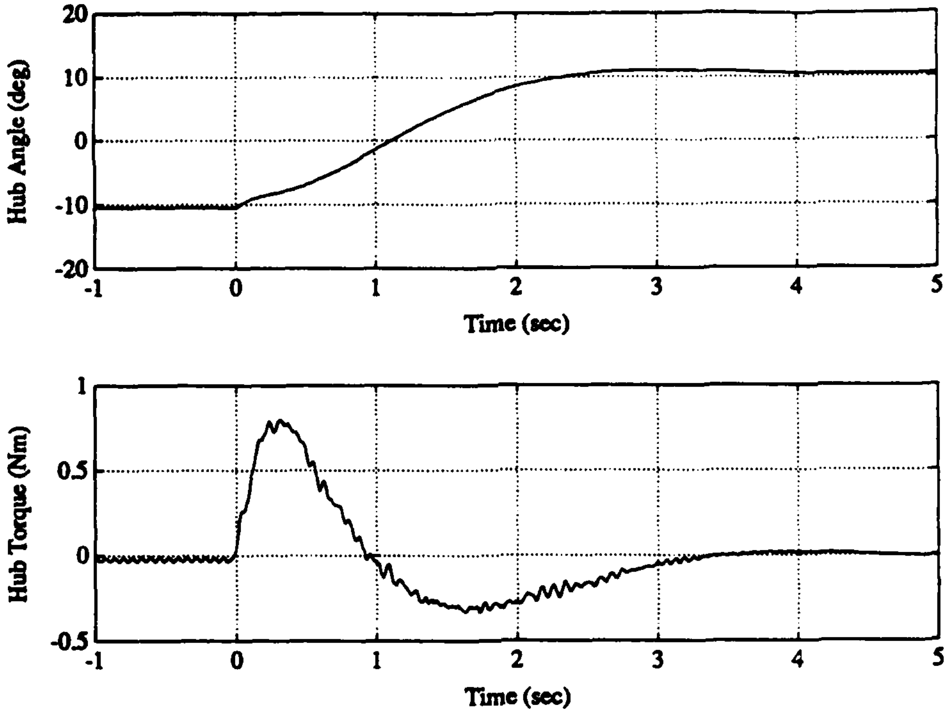

Figure 3.5: Step Response for Collocated Control of Flexible Arm

Here, T_H is the hub motor torque. T_A and T_B are the two mini-manipulator control torques. The gereralized coordinates q_1, q_{n+1}, and q_{n+2} correspond to the hub angle of the flexible arm, and to the two mini-manipulator joint angles respectively. The time derivatives of q_1, q_{n+1}, and q_{n+2} are designated u_1, u_{n+1}, and u_{n+2} respectively. Since this control mode does not require high performance, the selection of feedback gains was not rigorously optimized. Instead, gains were simply selected by using values which experimentally proved to yield acceptable performance.

Figure 3.5 shows the response of the flexible arm to a step command in hub angle position. The hub angle takes just over 2 seconds to make a 20 degree slew with very little overshoot.

Figure 3.6 shows the response of one of the mini-manipulator links to a step command in link angle. The slew takes approximately 0.2 seconds and has negligible overshoot. Notice that there is a significant delay between the time when the actuator begins to output a torque and when the link begins to move. This is due primarily to brush friction in the mini-manipulator motors.

3.5 Friction Compensation

As described in Section 3.2, the hub motor is brushless, and hence adds very little friction to the hub of the flexible arm. On the other hand, the mini-manipulator motors *do* have brushes, and consequently they add a much greater amount of friction to the two base

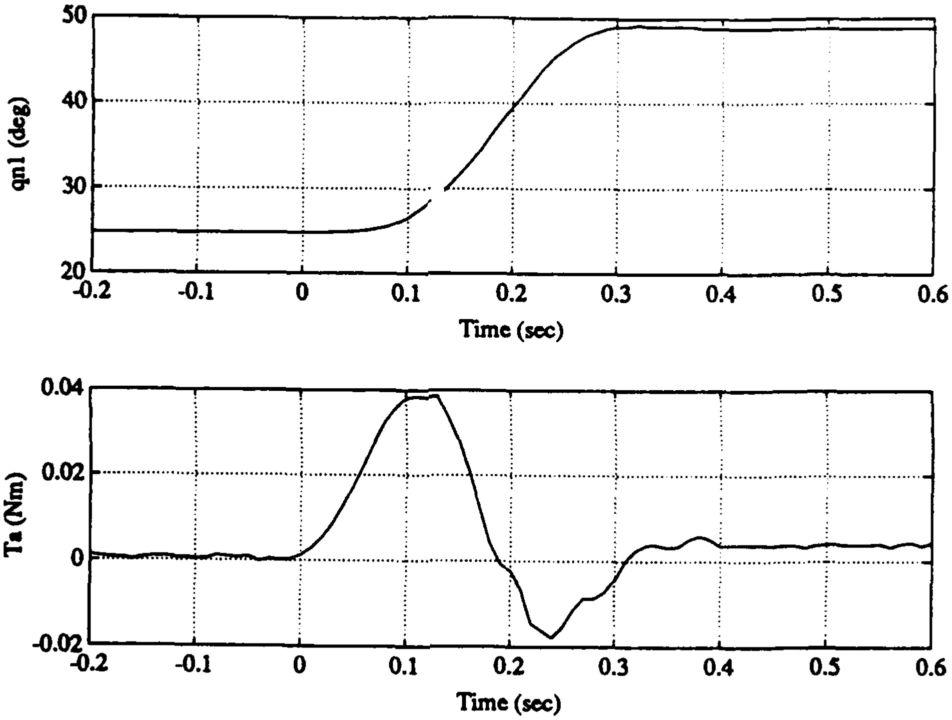

Figure 3.6: Step Response for Collocated Control of Mini-Manipulator

joints of the mini-manipulator. The friction level present in the two mini-manipulator motors is roughly 0.007 Nm – about 5% of the peak torque of 0.15 Nm, and as much as 20% of the torque required to control end-point force to common levels of about 35 grams force. In light of these significant levels of friction, some means of reducing the effect of friction on the system was sought.

It was experimentally determined that the friction present in the motors was essentially Coulomb friction. There was no discernible difference between static and kinetic friction levels, and no significant viscous friction. Thus, the common practice of using a dither signal to break-up static friction is not effective in this case.

Another type of friction compensation that was attempted was to feed-forward an estimate of the Coulomb friction level to the motors. In one case, this estimate was simply the absolute value of the Coulomb friction level multiplied by the sign of the corresponding motor shaft angular velocity. More specifically,

$$T_A = \mathcal{F}\mathrm{sgn}(u_{n+1}) \tag{3.19}$$

$$T_B = \mathcal{F}\mathrm{sgn}(u_{n+2}). \tag{3.20}$$

where T_A and T_B are the two mini-manipulator motor torques, $\mathcal{F}$ is the absolute value of the Coulomb friction level, and u_{n+1} and u_{n+2} are the two motor shaft angular velocities. This scheme of friction compensation has a very noticeable drawback: when the angular velocities are small, the friction compensation torques tend to "chatter" (i.e. oscillate very rapidly) unless $\mathcal{F}$ is made very small. This chatter can be minimized by the addition a dead-band and/or hysteresis.

However, what experimentally proved to yield the best friction compensation was the use of linear, angular velocity feed-forward with limits. Under this scheme, the friction compensation torques may be described by

$$T_A = \begin{cases} \left(\frac{\mathcal{F}}{u_{lim}}\right) u_{n+1} & |u_{n+1}| \leq u_{lim} \\ \mathcal{F} & |u_{n+1}| > u_{lim} \end{cases} \tag{3.21}$$

$$T_B = \begin{cases} \left(\frac{\mathcal{F}}{u_{lim}}\right) u_{n+2} & |u_{n+2}| \leq u_{lim} \\ \mathcal{F} & |u_{n+2}| > u_{lim} \end{cases} . \tag{3.22}$$

This form of friction compensation was used only while the mini-manipulator was under end-point position control. It would also have been desirable to use some kind of friction compensation during force control, but none of the above methods yielded satisfactory results — the primary problem was that they all tended to destabilize the torsion mode.

3.6 Identification of the Flexible Arm

As was demonstrated in Section 3.4, achieving satisfactory collocated control of the flexible arm does not require an accurate model of the system. The price one pays for this is poor performance. Achieving high-performance end-point control of the flexible arm, on the other hand, requires a fairly accurate description of the flexible arm's rigid body and bending modes.

Since the whole point in having a mini-manipulator at the end of the flexible arm is to produce much higher performance than possible with the flexible arm alone, it is reasonable and very generic to assume that the controllers for the two systems will be bandwidth separated. This spectral separation allows one to design the two controllers independently. Thus, in terms of the flexible arm, it is only necessary to identify transfer functions from the hub motor input to the two flexible arm outputs: end-point position and hub angle.

3.6.1 Experimental Procedure

One way to identify the flexible arm is through the use of sine sweeps. That is, inputting a sinusoidally varying torque to the hub motor, and measuring the relative magnitude and phase of the end-point position and hub angle for a wide range of frequencies.

A frequency response analyzer was used to provide the sinusoidal input, and to calculate the relative magnitude and phase of the outputs. The input frequency was varied from as low as practically possible, 0.1 Hz, to a point where the bending mode resonances became insignificantly small, 14.0 Hz.

The experimental transfer function data taken with the frequency response analyzer was then fit to two model transfer functions: one for the transfer function from hub torque to hub angle, and the other for the transfer function from hub torque to flexible arm end-point position. This fit was done using a numerical conjugate gradient search algorithm to minimize a cost function.

3.6.2 A Cost Function for the Transfer Function

Let $G(s)$ represent the actual transfer function we wish to identify, and let $\hat{G}(s)$ represent our estimate of $G(s)$. We then wish to minimize some measure of the difference between $G(s)$ and $\hat{G}(s)$. This has to be done at each of the discrete frequency values s_j $(j = 1, \ldots, n)$ where $G(s)$ was measured. A typical measure to minimize is the square error given by

$$J_2 = \sum_{i=1}^{n} \left\{ G(s_i) - \hat{G}(s_i) \right\}^2 . \tag{3.23}$$

This cost function, however, has the disadvantage of being heavily influenced by large errors. A cost function that is less sensitive to large errors is one that is based on the absolute value of the error.

$$J_1 = \sum_{i=1}^{n} \left| G(s_i) - \hat{G}(s_i) \right| . \tag{3.24}$$

This is the measure that was minimized to obtain an "optimal" estimate of $G(s)$.

3.6.3 Experimental Model of the Flexible Arm

As has been shown in the past, a truncated modal form may be used to approximate the transfer function of the system. Since no resonances beyond the third bending mode were experimentally observed, only the first three modes were modeled. The transfer function from hub torque to hub angle may be approximated as

$$\frac{q_1(s)}{T_H(s)} = \frac{r_{q0}}{s(s + a_H)} + \sum_{i=1}^{3} \left\{ \frac{r_{q_i}}{s^2 + 2\zeta_i \omega_i s + \omega_i^2} \right\} . \tag{3.25}$$

The first term in Eq. (3.25) refers to the rigid body mode of the flexible arm. The remaining three terms describe the first three bending modes of the flexible arm. Similarly, the transfer function from hub torque to the end-point position of the flexible arm may be approximated as

$$\frac{W_{Ex}(s)}{T_H(s)} = \left(\frac{a_{fwex}}{s + a_{fwex}} \right) \left(\frac{r_{wex_0}}{s(s + a_H)} + \sum_{i=1}^{3} \left\{ \frac{r_{wex_i}}{s^2 + 2\zeta_i \omega_i s + \omega_i^2} \right\} \right) . \tag{3.26}$$

Equation (3.26) has a leading factor not present in Eq. (3.25). This leading factor reflects the presence of a 10 Hz analog low-pass filter.

The parameter a_H represents the amount of viscous damping present in the hub motor, and was experimentally determined by examining the rate of decay of the rigid body mode. The parameter a_{wex} represents the known cutoff frequency of the low-pass filter for the flexible arm end-point position sensor. Another parameter, that will be discussed more later, is a_{fu1}. It represents the known cutoff frequency of the pseudo-differentiator that was used to derive angular rate information from measurements of the hub angle. The other parameters in Eqs. (3.25) and (3.26), namely r_{q_i}, r_{wex_i}, ζ_i,

Hub Motor Damping	Filter Cutoff Frequencies	
a_H	a_{fwex}	a_{ful}
0.15 rad/sec	10 Hz	10.27 Hz

Mode	Residue		Frequency (Hz)	Damping Ratio
i	r_{q_i}	r_{wex_i}	ω_i	ζ_i
0	0.2166	-2.6863	0	—
1	0.4219	2.1429	2.60	0.0298
2	8.2046	-2.1077	4.85	0.0153
3	0	2.4795	10.40	0.0248

Table 3.1: Values of the Flexible Arm Model Parameters

and ω_i, were determined by minimizing the cost function J_1. The numerical values for all of these parameters are tabulated in Table 3.1.

Using Eqs. (3.25) and (3.26), and the values listed in Table 3.1, a numerical theoretical model of the flexible arm was constructed. Figures 3.7 and 3.8 show the Bode plots of the flexible arm transfer functions. One trace shows the actual experimental values obtained using the frequency analyzer, and the other trace shows the values obtained using the mathematical model.

Looking at the transfer function from hub torque to hub angle, one notices that there is an alternating pole-zero pattern. Starting from the low end of the frequency scale and moving to the right, one encounters first a zero, then a pole, a zero, and then another pole — basically, two poles and zeros. The third bending mode makes no significant appearance in this transfer function.

In contrast, the transfer function from hub torque to flexible arm end-point position has three discernible poles, and *no* discernible zeros. This apparent lack of zeros arises because the zeros of the transfer function are *not* located near the imaginary axis in the s-plane. In fact, some of the zeros actually lie in the right half-plane. It is this non-minimum phase characteristic of the non-collocated transfer function that makes good end-point control so difficult to achieve.

3.7 End-Point Control Design for the Flexible Arm

3.7.1 State Space Model

For purposes of control law design, it is convenient to recast the equations describing the theoretical model of the flexible arm, namely Eqs. (3.25) and (3.26), into a state space representation. In such a form, the equations of motion of the system may be described simply as

$$\begin{aligned} \dot{x} &= Fx + Gu \\ y &= Hx. \end{aligned} \tag{3.27}$$

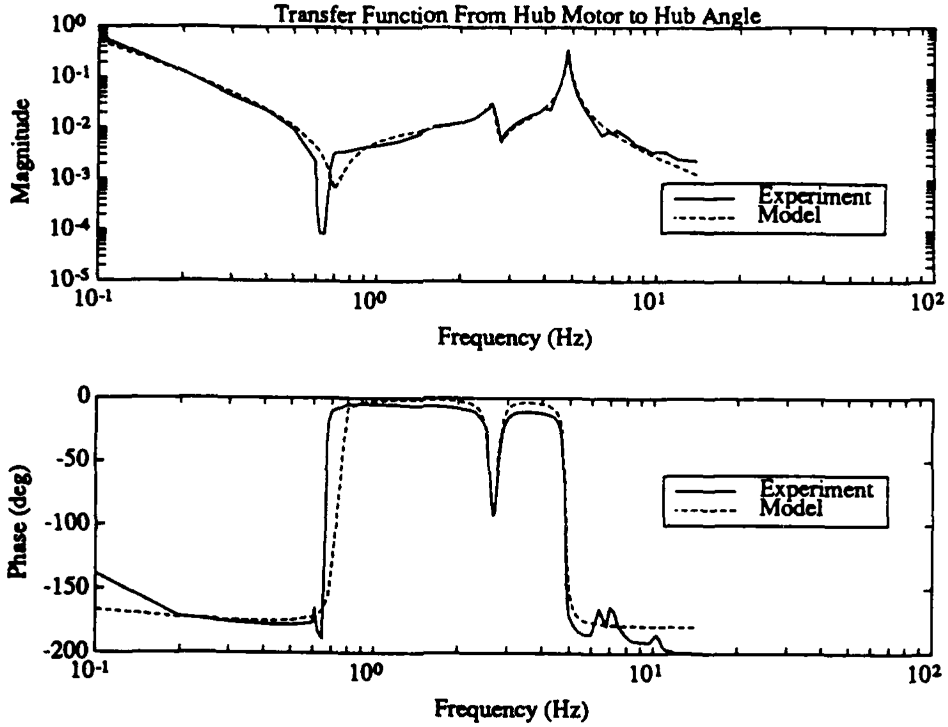
Transfer Function From Hub Motor to Hub Angle

Figure 3.7: Bode Plot for Flexible Arm Hub Angle

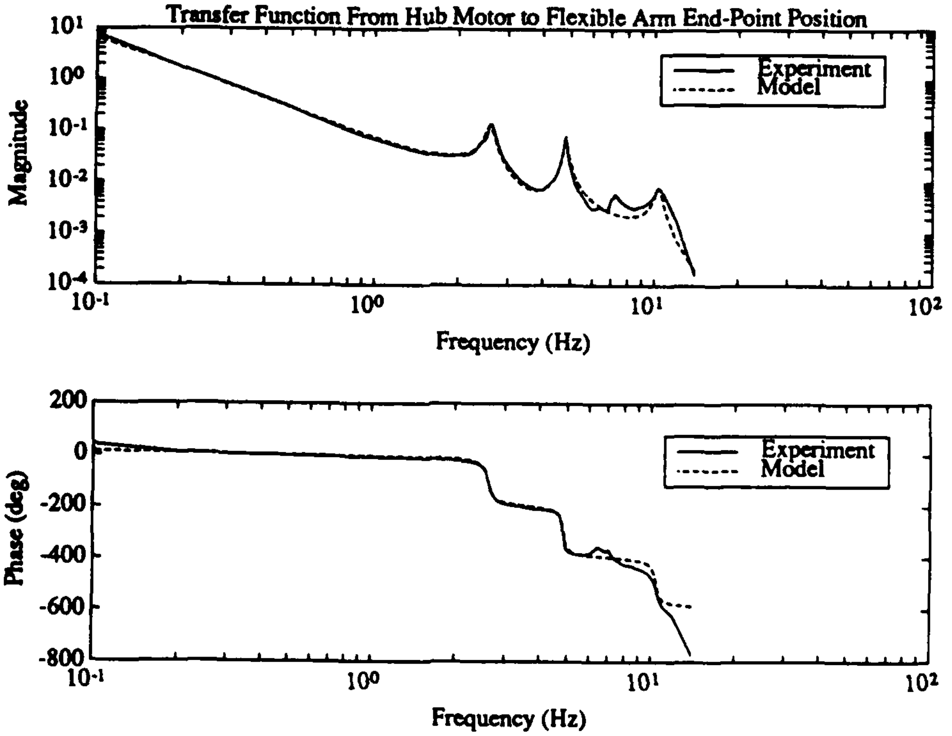
Transfer Function From Hub Motor to Flexible Arm End-Point Position

Figure 3.8: Bode Plot for Flexible Arm End-Point Position

Using a modal representation, one can arrive at the following block diagonal form for the system dynamics matrices

$$\boldsymbol{F}_D = \begin{bmatrix} 0 & 1 & & & & & & \\ 0 & -a_H & & & & & & \\ & & 0 & 1 & & & & \\ & & -\omega_1^2 & -2\zeta_1\omega_1 & & & & \\ & & & & 0 & 1 & & \\ & & & & -\omega_2^2 & -2\zeta_2\omega_2 & & \\ & & & & & & 0 & 1 \\ & & & & & & -\omega_3^2 & -2\zeta_3\omega_3 \end{bmatrix} \tag{3.28}$$

$$\boldsymbol{F} = \left[\begin{array}{c|cc} \boldsymbol{F}_D & & \\ \hline a_{fwex}\boldsymbol{h}_{wex} & -a_{fwex} & 0 \\ a_{fu1}\boldsymbol{h}_{u1} & 0 & -a_{fu1} \end{array}\right] \tag{3.29}$$

$$\boldsymbol{G} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}^T \tag{3.30}$$

$$\boldsymbol{H} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.31}$$

where

$$\boldsymbol{x} = \begin{bmatrix} q_0 & \dot{q}_0 & q_1 & \dot{q}_1 & q_2 & \dot{q}_2 & q_3 & \dot{q}_3 & W_{E_{X_m}} & u_{1m} \end{bmatrix}^T \tag{3.32}$$

$$\boldsymbol{u} = T_H \tag{3.33}$$

$$\boldsymbol{y} = \begin{bmatrix} W_{E_{X_m}} \\ u_{1m} \end{bmatrix}. \tag{3.34}$$

The quantities $W_{E_{X_m}}$ and u_{1m} are simply the filtered measurements of W_{E_X} and u_1 respectively. These two filter transfer functions are given by

$$\frac{W_{E_{X_m}}}{W_{E_X}} = \left(\frac{a_{fwex}}{s + a_{fwex}}\right) \tag{3.35}$$

$$\frac{u_{1m}}{u_1} = \left(\frac{a_{u1}}{s + a_{fwex}}\right). \tag{3.36}$$

The output matrices $\boldsymbol{h}_{wex}$ and $\boldsymbol{h}_{u1}$ are defined by

$$W_{E_X} = \boldsymbol{h}_{wex}\boldsymbol{x} \tag{3.37}$$

$$u_1 = \boldsymbol{h}_{u1}\boldsymbol{x} \tag{3.38}$$

where

$$\boldsymbol{h}_{wex} = \begin{bmatrix} r_{wex_0} & 0 & r_{wex_1} & 0 & r_{wex_2} & 0 & r_{wex_3} & 0 & 0 & 0 \end{bmatrix} \tag{3.39}$$

$$\boldsymbol{h}_{u1} = \begin{bmatrix} 0 & r_{ql_0} & 0 & r_{ql_1} & 0 & r_{ql_2} & 0 & r_{ql_3} & 0 & 0 \end{bmatrix}. \tag{3.40}$$

Since the flexible arm is to be controlled by a digital computer, it is necessary to have a discrete time model of the system. The continuous time system dynamics described by Eqs. (3.27) may be transformed via a zero order hold into a discrete time representation given by

$$\begin{aligned} \boldsymbol{x}_{n+1} &= \boldsymbol{\Phi x}_n + \boldsymbol{\Gamma} u_n \\ y_n &= \boldsymbol{H x}_n . \end{aligned} \tag{3.41}$$

3.7.2 Optimal Regulator Design

An "Optimal Regulator" is a type of controller that minimizes some cost function. The standard formulation is to minimize the cost J given by

$$J = \frac{1}{2} \sum_{n=0}^{N} [\boldsymbol{x}_n^T \boldsymbol{R}_{xx} \boldsymbol{x}_n + \boldsymbol{u}_n^T \boldsymbol{R}_{uu} \boldsymbol{u}_n] \tag{3.42}$$

subject to the constraint

$$\boldsymbol{x}_{n+1} = \boldsymbol{\Phi x}_n + \boldsymbol{\Gamma u}_n . \tag{3.43}$$

$\boldsymbol{R}_{xx}$ and $\boldsymbol{R}_{uu}$ are symmetric weighting matrices on the states and the controls, selected to yield a desired performance. Since the quantity that is directly trying to be controlled is W_{E_X}, it is reasonable to choose $\boldsymbol{R}_{xx}$ such that there is a cost associated with non-zero values of W_{E_X}. This may be done by choosing

$$\boldsymbol{R}_{xxwex} = \boldsymbol{h}_{wex}^T \boldsymbol{h}_{wex} . \tag{3.44}$$

The damping in the system was enhanced by adding costs to the elements of $\boldsymbol{R}_{xxwex}$ corresponding to each of the modes of the system. In this case, the following modifications to the basic $\boldsymbol{R}_{xxwex}$ given in Eq. (3.44) were made:

$$\begin{aligned} R_{xxwex}(2,2) &\Longleftarrow R_{xxwex}(2,2) + 0.2 \\ R_{xxwex}(4,4) &\Longleftarrow R_{xxwex}(4,4) + 1.0 \\ R_{xxwex}(6,6) &\Longleftarrow R_{xxwex}(6,6) + 10.0 \\ R_{xxwex}(8,8) &\Longleftarrow R_{xxwex}(8,8). \end{aligned} \tag{3.45}$$

Using a control cost

$$R_{uuwex} = 0.0025 \tag{3.46}$$

the cost function given by Eq. (3.42) was minimized to obtain the optimal feedback control law

$$u_n = \boldsymbol{K}_{wex}(\boldsymbol{x}_{command} - \boldsymbol{x}_n) \tag{3.47}$$

where $\boldsymbol{x}_{command}$ is the desired state vector and $\boldsymbol{K}_{wex}$ is the feedback gain matrix. The actual numerical values of the feedback gains are

$$\boldsymbol{K}_{wex} = [\ 35.71 \quad 12.34 \quad -11.63 \quad 11.65 \quad 622.01 \quad 35.41 \quad 1.79 \quad 0.02 \quad 0 \quad 0\ ]. \tag{3.48}$$

3.7.3 Steady-State Optimal Estimator Design

Since not all of the states are available for measurement, a steady-state optimal estimator was used to estimate the state vector. For a discrete time system given by

$$\begin{aligned} \boldsymbol{x}_{n+1} &= \boldsymbol{\Phi x}_n + \boldsymbol{\Gamma u}_n + \boldsymbol{\Gamma}_1 \boldsymbol{w}_n & (3.49) \\ \boldsymbol{y}_n &= \boldsymbol{H x}_n + \boldsymbol{v}_n & (3.50) \end{aligned}$$

where the process noise $\boldsymbol{w}_n$ and measurement noise $\boldsymbol{v}_n$ may be characterized by

$$\mathcal{E}\{\boldsymbol{w}_n\} = \mathcal{E}\{\boldsymbol{v}_n\} = 0 \qquad (3.51)$$

$$\mathcal{E}\{\boldsymbol{w}_i \boldsymbol{w}_j^T\} = \mathcal{E}\{\boldsymbol{v}_i \boldsymbol{v}_j^T\} = 0 \quad \text{if} \quad i \neq j \qquad (3.52)$$

and

$$\mathcal{E}\{\boldsymbol{\Gamma}_1 \boldsymbol{w}_n \boldsymbol{w}_n^T \boldsymbol{\Gamma}_1^T\} \triangleq \boldsymbol{Q}_{xx} \qquad (3.53)$$

$$\mathcal{E}\{\boldsymbol{v}_n \boldsymbol{v}_n^T\} \triangleq \boldsymbol{Q}_{yy} \qquad (3.54)$$

then it is desired to choose $\boldsymbol{L}$ such that

$$\begin{aligned} \bar{\boldsymbol{x}}_{n+1} &= \boldsymbol{\Phi} \hat{\boldsymbol{x}}_n + \boldsymbol{\Gamma u}_n & (3.55) \\ \hat{\boldsymbol{x}}_{n+1} &= \bar{\boldsymbol{x}}_{n+1} + \boldsymbol{L}(\boldsymbol{y}_{n+1} - \boldsymbol{H} \bar{\boldsymbol{x}}_{n+1}) & (3.56) \end{aligned}$$

yields an "optimal" estimate $\hat{\boldsymbol{x}}$ of the true state $\boldsymbol{x}$. The actual values of the process noise and measurement noise covariances were not determined. Instead, $\boldsymbol{Q}_{xx}$ and $\boldsymbol{Q}_{yy}$ were treated as design parameters to arrive at an estimator that had satisfactory performance. In choosing $\boldsymbol{Q}_{xx}$ and $\boldsymbol{Q}_{yy}$, there is a trade-off between the "speed" of the estimator error roots, and the "noisiness" of the state estimate. Basically, the noise covariance matrices were chosen to yield the fastest possible estimator error roots, while providing no more noise in the output estimate than was actually measured. Using the values

$$\begin{aligned} \boldsymbol{Q}_{xxwex} &= \boldsymbol{\Gamma \Gamma}^T & (3.57) \\ \boldsymbol{Q}_{yywex} &= \begin{bmatrix} 10^{-4} & 0 \\ 0 & 10^{-2} \end{bmatrix}, & (3.58) \end{aligned}$$

the estimator gain matrix was found to be

$$\boldsymbol{L}_{wex} = \begin{bmatrix} -0.11 & -0.74 & -0.01 & 0.38 & -0.00 & 0.06 & 0.00 & -0.08 & 0.25 & 0.28 \\ -0.00 & 0.04 & 0.00 & 0.06 & 0.00 & 0.06 & 0.00 & -0.02 & 0.00 & 0.47 \end{bmatrix}^T . \qquad (3.59)$$

Now, based on measurements of $W_{E_{X_m}}$ and u_{1m}, the state vector $\boldsymbol{x}$ may be estimated using the current estimator equations given in Eqs. (3.55) and (3.56).

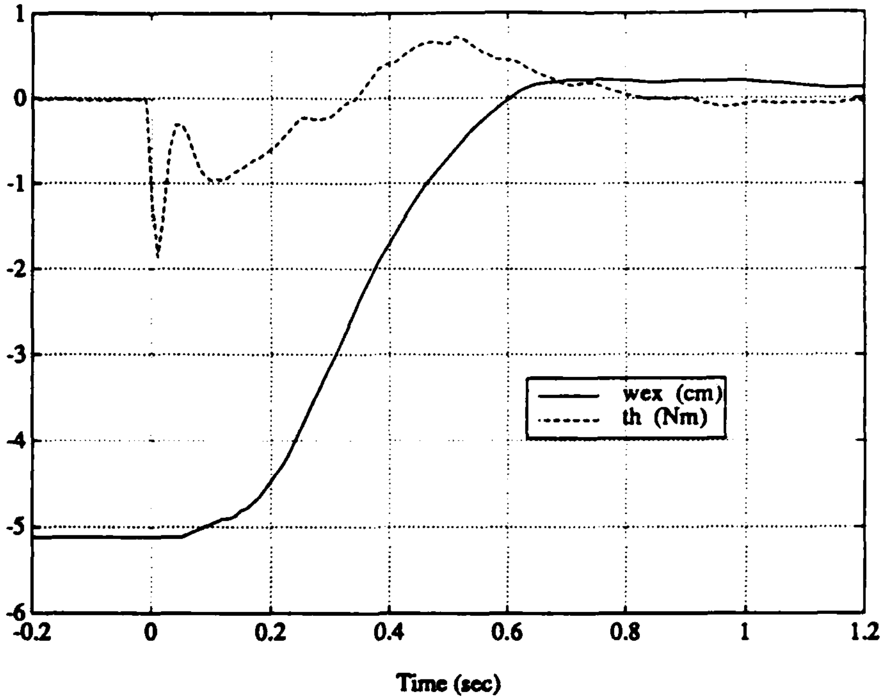

Figure 3.9: End-Point Control Step Response for the Flexible Arm

3.8 End-Point Control of the Flexible Arm

Using the optimal regulator and estimator designed in Section 3.7, end-point position control of the flexible arm is now possible. Running the controller at a sample rate of 100 Hz, the flexible arm was given a step command in end-point position. Figure 3.9 shows the time history of the end-point of the flexible arm and the control torque that was used. One will notice that there is a roughly 50 msec (5 sample period) delay from the time the step command is given to the time the flexible arm end-point begins to move. This represents the time required for a bending wave to propagate down the length of the flexible arm. The commanded slew of just over 5 cm was accomplished in 0.65 seconds with negligible overshoot. This compares quite favorably with the period of the first cantilever mode of the flexible arm which is 1.4 seconds. This step resulted in a peak control torque of approximately the same size as the maximum torque available.

3.9 Identification of the Mini-Manipulator

Now that the flexible arm is under end-point position control, there remains the task of closing both a position and force control loop around the end-point of the mini-manipulator. To do this, we must have some description of the dynamics of the mini-manipulator. Figure 3.10 depicts a close-up view of the mini-manipulator. One will note that in terms of the actuator torques applied at points X_A and X_B, and the Cartesian position of the end-point X_E, this is a highly non-linear mechanism. It does, however,

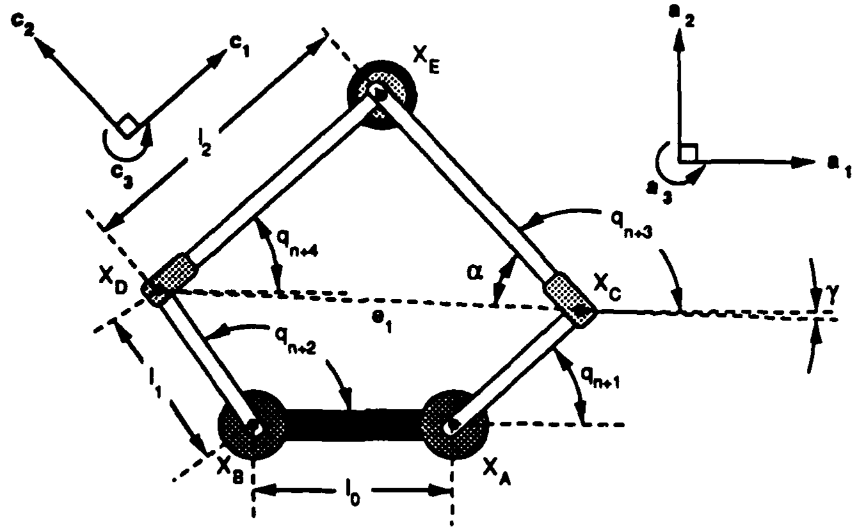

Figure 3.10: A Close-Up View of the Mini-Manipulator

have the advantage of being very elegant mechanically: it is all direct drive, and the motors are not required to accelerate heavy masses such as other motors.

One strategy that is used in controlling non-linear systems involves having a knowledge of the governing equations of motion, and using this to make the non-linear system behave like a desired simple linear system. This is known as computed torque. Since the full equations of motion of the mini-manipulator are far too complex to be dealt with in real time with available computer power, some other means of addressing the control problem must be found.

For a mini-manipulator that has linkages which are light compared to the end-point mass, the equations of motion may be approximated by a *much* simpler set of equations. All of the dynamic non-linear terms in the governing equations of motion are associated with the dynamics of the linkages. Since in this case the linkages are relatively lightweight, the non-linear dynamic terms in the equations of motion may be neglected. Thus, one is left only with kinematic non-linearities and a greatly simplified set of equations which approximate the full equations of motion of the mini-manipulator.

3.9.1 The Mini-Manipulator Jacobian

One can relate the variation in end-point position as seen in the mini-manipulator reference frame A to the variation in joint angles via

$$\delta\begin{bmatrix} R_x \\ R_y \end{bmatrix} = J\,\delta\begin{bmatrix} q_{n+1} \\ q_{n+2} \end{bmatrix}, \tag{3.60}$$

where R_x and R_y are the Cartesian coordinates of the end-point X_E in reference frame A. The 2×2 matrix J is known as the Jacobian. In general, Eq. (3.60) may be written as

$$\delta x = J\,\delta q. \tag{3.61}$$

If one considers the case of static equilibrium, the principle of virtual work may be applied to the mini-manipulator. If $\boldsymbol{T}$ is a vector whose elements are the joint torques, and $\boldsymbol{F}$ is a vector whose elements are the forces exerted *by* the end-point on the environment, then the principle of virtual work may be stated as

$$\boldsymbol{T}^T\delta\boldsymbol{q} - \boldsymbol{F}^T\delta\boldsymbol{x} = 0. \tag{3.62}$$

Substituting Eq. (3.62) into Eq. (3.61), one arrives at the expression

$$\boldsymbol{T} = \boldsymbol{J}^T\boldsymbol{F}, \tag{3.63}$$

or more specifically

$$\begin{bmatrix} T_A \\ T_B \end{bmatrix} = \boldsymbol{J}^T \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}. \tag{3.64}$$

Thus, given a desired applied force, Eq. (3.64) allows one to calculate the joint torques necessary to provide that force. In essence, we can make the system appear to behave as if a force $f_1\boldsymbol{a}_1 + f_2\boldsymbol{a}_2$ is acting on the end-point using only control torques T_A and T_B. One must remember that this relationship holds exactly only when the base link of the mini-manipulator is inertially fixed. However, actual practice has shown that for the relatively slow motions of the flexible arm, Eq. (3.64) is still a very good approximation.

3.9.1.1 Evaluating the Jacobian

Referring to Fig. 3.10, the sine and cosine of the angle q_{n+4}, represented by c_{n+4} and s_{n+4}, may be found from known quantities via

$$\begin{aligned}
e_1 &= \sqrt{[l_0 + l_1(c_{n+1} - c_{n+2})]^2 + [l_1(s_{n+1} - s_{n+2})]^2} && (3.65)\\
c_\alpha &= \frac{e_1}{2l_2} && (3.66)\\
s_\alpha &= \sqrt{1 - c_\alpha^2} && (3.67)\\
s_\gamma &= \frac{l_1}{e_1}(s_{n+2} - s_{n+1}) && (3.68)\\
c_\gamma &= \sqrt{1 - s_\gamma^2} && (3.69)\\
c_{n+4} &= s_\alpha s_\gamma + c_\alpha c_\gamma && (3.70)\\
s_{n+4} &= s_\alpha c_\gamma - c_\alpha s_\gamma. && (3.71)
\end{aligned}$$

Then, the following sequence of equations determine the Jacobian:

$$\begin{aligned}
\frac{\partial e_1}{\partial q_{n+1}} &= \frac{1}{e_1}\left[l_1^2(s_{n+1}c_{n+2} - c_{n+1}s_{n+2}) - l_0 l_1 s_{n+1}\right] && (3.72)\\
\frac{\partial e_1}{\partial q_{n+2}} &= \frac{1}{e_1}\left[l_1^2(c_{n+1}s_{n+2} - s_{n+1}c_{n+2}) + l_0 l_1 s_{n+2}\right] && (3.73)
\end{aligned}$$

$$\frac{\partial\gamma}{\partial q_{n+1}} = \frac{-l_1}{c_\gamma e_1^2}\left[e_1 c_{n+1} + (s_{n+2} - s_{n+1})\frac{\partial e_1}{\partial q_{n+1}}\right] \tag{3.74}$$

$$\frac{\partial\gamma}{\partial q_{n+2}} = \frac{l_1}{c_\gamma e_1^2}\left[e_1 c_{n+2} - (s_{n+2} - s_{n+1})\frac{\partial e_1}{\partial q_{n+2}}\right] \tag{3.75}$$

$$\frac{\partial\alpha}{\partial q_{n+1}} = \frac{-1}{2l_2 s_\alpha}\frac{\partial e_1}{\partial q_{n+1}} \tag{3.76}$$

$$\frac{\partial\alpha}{\partial q_{n+2}} = \frac{-1}{2l_2 s_\alpha}\frac{\partial e_1}{\partial q_{n+2}} \tag{3.77}$$

$$\frac{\partial q_{n+4}}{\partial q_{n+1}} = \frac{\partial\alpha}{\partial q_{n+1}} - \frac{\partial\gamma}{\partial q_{n+1}} \tag{3.78}$$

$$\frac{\partial q_{n+4}}{\partial q_{n+2}} = \frac{\partial\alpha}{\partial q_{n+2}} - \frac{\partial\gamma}{\partial q_{n+2}} \tag{3.79}$$

$$J_{11} = \frac{\partial R_x}{\partial q_{n+1}} = -l_2 s_{n+4}\frac{\partial q_{n+4}}{\partial q_{n+1}} \tag{3.80}$$

$$J_{12} = \frac{\partial R_x}{\partial q_{n+2}} = -l_1 s_{n+2} - l_2 s_{n+4}\frac{\partial q_{n+4}}{\partial q_{n+2}} \tag{3.81}$$

$$J_{21} = \frac{\partial R_y}{\partial q_{n+1}} = l_2 c_{n+4}\frac{\partial q_{n+4}}{\partial q_{n+1}} \tag{3.82}$$

$$J_{22} = \frac{\partial R_y}{\partial q_{n+2}} = l_1 c_{n+2} + l_2 c_{n+4}\frac{\partial q_{n+4}}{\partial q_{n+2}}. \tag{3.83}$$

The Jacobian is then

$$J = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix}. \tag{3.84}$$

3.9.2 Experimental Identification

With the Jacobian and the assumption of relatively lightweight linkages, we now have a means of experimentally identifying the dynamics of the mini-manipulator. The general idea is to input a desired sinusoidally varying end-point force, and to measure the resulting position *of*, and force *at*, the mini-manipulator end-point.

3.9.2.1 Position Transfer Function

Using the frequency response analyzer, a sinusoidal signal $s(t)$ was generated. This was then used to generate two desired end-point force vectors given by

$$F_{d_x} = \begin{bmatrix} s(t) \\ 0 \end{bmatrix} \tag{3.85}$$

$$F_{d_y} = \begin{bmatrix} 0 \\ s(t) \end{bmatrix}. \tag{3.86}$$

One may then substitute these desired force vectors into Eq. (3.63) to yield two sets of mini-manipulator joint torques

$$T_x = J^T F_{d_x} \tag{3.87}$$

$$T_y = J^T F_{d_y}. \tag{3.88}$$

Then, while applying joint torque T_x, one uses the frequency analyzer to measure the lateral position of the mini-manipulator end-point R_x and to determine the relative magnitude and phase. This determines the transfer function from lateral force to lateral position, or $R_x(s)/f_{1d}(s)$. In a similar fashion, by applying joint torque T_y and measuring R_y, one arrives at the transfer function from longitudinal force to longitudinal position, or $R_y(s)/f_{2d}(s)$.

Again, using a truncated modal form, the experimental frequency response data was fit to a model described by

$$\frac{R_x(s)}{f_{1d}(s)} = \left(\frac{a_{fp}}{s+a_{fp}}\right)\left(\frac{r_{rx_0}}{s^2}+\sum_{i=1}^{m_{rx}}\left\{\frac{r_{rx_i}}{s^2+2\zeta_{rx_i}\omega_{rx_i}s+\omega_{rx_i}^2}\right\}\right) \tag{3.89}$$

$$\frac{R_y(s)}{f_{2d}(s)} = \left(\frac{a_{fp}}{s+a_{fp}}\right)\left(\frac{r_{ry_0}}{s^2}+\sum_{i=1}^{m_{ry}}\left\{\frac{r_{ry_i}}{s^2+2\zeta_{ry_i}\omega_{ry_i}s+\omega_{ry_i}^2}\right\}\right). \tag{3.90}$$

It should be noted that although the mini-manipulator represents a highly non-linear device, we are mapping the mini-manipulator dynamics through the Jacobian and attempting to fit the result to a *linear* model. The leading factors in Eqs. (3.89) and (3.90) reflect the presence of a 40 Hz analog low-pass filter. As before, the cost function J_1 given in Eq. (3.24) was minimized to yield optimal estimates of the parameters r_{rx_i}, r_{ry_i}, ω_{rx_i}, ω_{ry_i}, ζ_{rx_i}, and ζ_{ry_i}. In the lateral direction, the experimental data was fit to Eq. (3.89) for $m_{rx} = 1$. The reason for using $m_{rx} = 1$ is that experimental evidence showed that only the rigid body and torsion mode were significant. In the longitudinal direction, the experimental data was fit to Eq. (3.90) for $m_{ry} = 0$. Here, the reason for only considering the rigid body mode is that bending and torsion in the flexible arm affect motions primarily only in the lateral direction. The numerical values of the parameters obtained from the optimization are listed in Table 3.2.

Using Eqs. (3.89) and (3.90), and the values listed in Table 3.2, a numerical, theoretical model of the mini-manipulator position dynamics was constructed. Figures 3.11 and 3.12 show the Bode plots of the mini-manipulator position transfer functions. As before, one trace shows the actual experimental values obtained using the frequency response analyzer, and the other trace shows the values obtained using the mathematical model.

Examining the transfer function from lateral end-point force to position, or $R_x(s)/f_1(s)$, one can see that the two mode linear model does a reasonable job of fitting the experimental data. One will note that, as predicted in Section 3.3, the zero frequency is less than the pole frequency. This results in "favorable" dynamics that make the system easy to control. The single pole seen in Fig. 3.11 is due to the torsion mode. The bending modes in the flexible arm have only a negligible impact on the frequency response — their presence being seen in the slight differences between experiment and model at about 3 Hz and 8 Hz.

The transfer function from longitudinal end-point force to position, or $R_y(s)/f_2(s)$, looks just like a $1/s^2$ plant. Again, the single mode linear model does a reasonable

Filter Cutoff Frequency a_{fp}
40 Hz

Lateral Direction ($m_{rx} = 1$)			
Mode i	Residue r_{rx_i}	Frequency (Hz) ω_{rx_i}	Damping Ratio ζ_{rx_i}
0	85.0834	—	—
1	19.3234	11.40	0.0385

Longitudinal Direction ($m_{ry} = 0$)			
Mode i	Residue r_{ry_i}	Frequency (Hz) ω_{ry_i}	Damping Ratio ζ_{ry_i}
0	120.7804	—	—

Table 3.2: Values of the Mini-Manipulator Model Parameters: Position

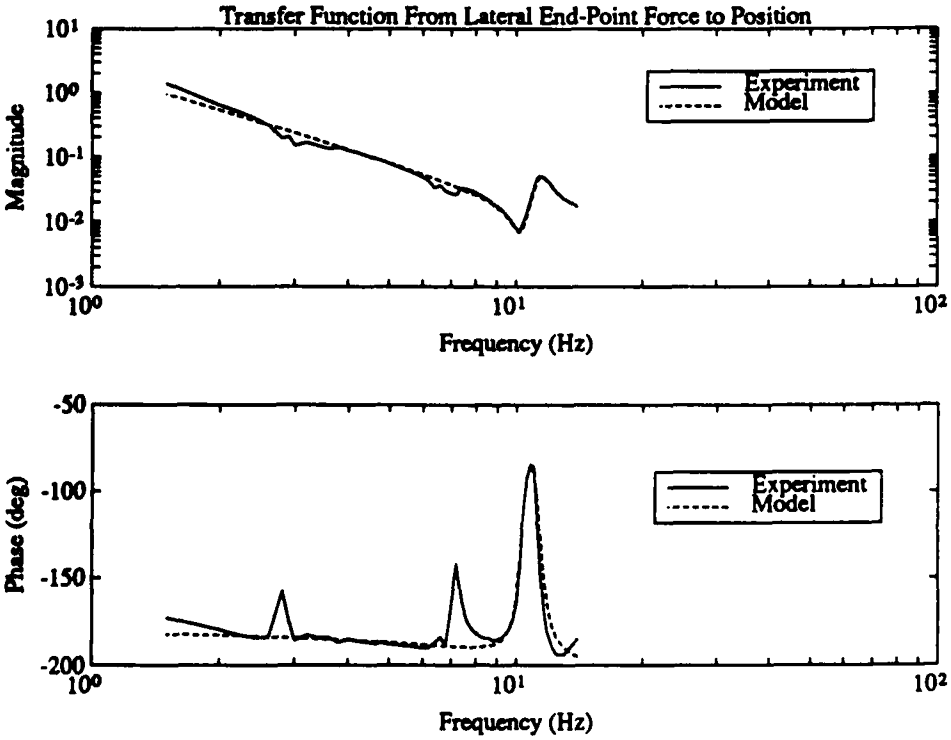

Figure 3.11: Bode Plot for Mini-Manipulator Position R_x

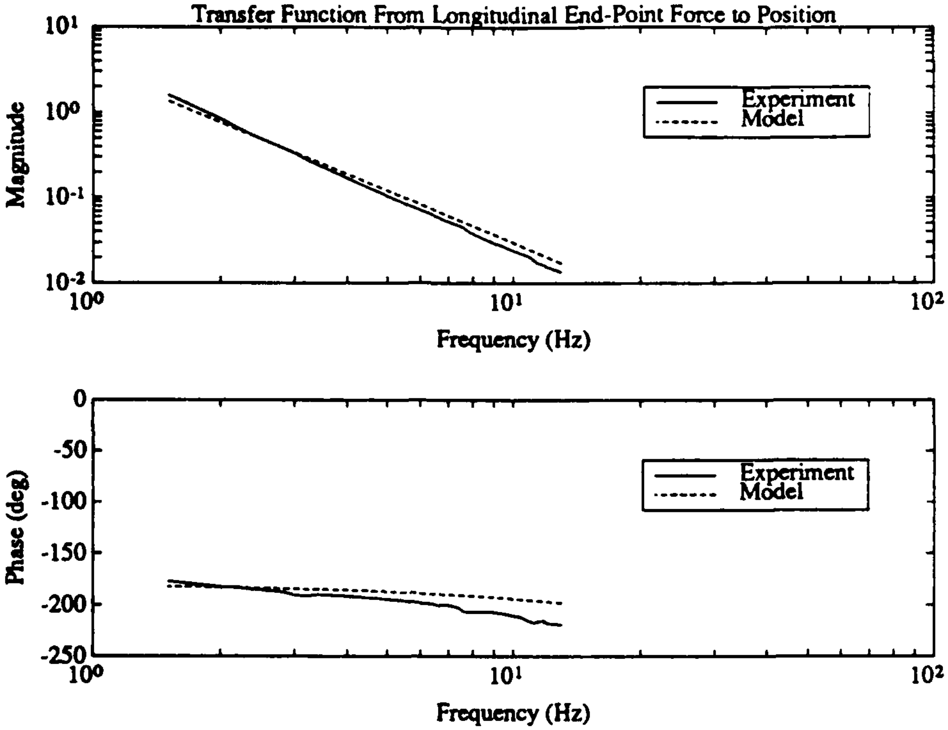

Figure 3.12: Bode Plot for Mini-Manipulator Position R_y

job of fitting the experimental data. The slightly larger phase lag in the experimental model may be attributed to time delays in the digital computer that was used during the identification process.

3.9.2.2 Force Transfer Function

To identify the force transfer functions, the mini-manipulator end-point was brought into contact with an inertially fixed object. As in identifying the position transfer functions of the mini-manipulator, the frequency response analyzer was used to generate the two desired end-point force vectors given by Eqs. (3.85) and (3.86). In this case, the sinusoidally varying desired force vectors contained a DC bias level to ensure that the mini-manipulator end-point remained in contact with the object. However, now instead of measuring the end-point position, the end-point force was measured. The lateral force f_1 and the longitudinal force f_2 were measured, thereby determining the transfer functions $f_1(s)/f_{1d}(s)$ and $f_2(s)/f_{2d}(s)$.

The truncated modal form used to fit the experimental data was

$$\frac{f_1(s)}{f_{1d}(s)} = B(s)\left(r_{f1_0} + \sum_{i=1}^{m_{f1}} \left\{ \frac{r_{f1_i}}{s^2 + 2\zeta_{f1_i}\omega_{f1_i} s + \omega_{f1_i}^2} \right\}\right) \qquad (3.91)$$

$$\frac{f_2(s)}{f_{2d}(s)} = B(s)\left(r_{f2_0} + \sum_{i=1}^{m_{f2}} \left\{ \frac{r_{f2_i}}{s^2 + 2\zeta_{f2_i}\omega_{f2_i} s + \omega_{f2_i}^2} \right\}\right). \qquad (3.92)$$

Bessel Type Filter			
b_{fn}	5.9530×10^7	DC Gain	1
b_{fd2}	819.62	Pole 1	-151.62
b_{fd3}	4.9391×10^5	Pole 2	$-334.00 + 530.16j$
b_{fd4}	5.9530×10^7	Pole 3	$-334.00 - 530.16j$

Lateral Direction ($m_{f1} = 1$)			
Mode i	Residue r_{f1_i}	Frequency (Hz) ω_{f1_i}	Damping Ratio ζ_{f1_i}
0	3.6444	—	—
1	7619.3	10.72	0.0568

Lateral Direction ($m_{f1} = 3$)			
Mode i	Residue r_{f1_i}	Frequency (Hz) ω_{f1_i}	Damping Ratio ζ_{f1_i}
0	2.9885	—	—
1	-237.01	3.46	0.0501
2	1015.2	8.85	0.0694
3	7154.6	10.72	0.0595

Longitudinal Direction ($m_{f2} = 0$)			
Mode i	Residue r_{f2_i}	Frequency (Hz) ω_{f2_i}	Damping Ratio ζ_{f2_i}
0	5.4385	—	—

Table 3.3: Values of the Mini-Manipulator Model Parameters: Force

where $B(s)$ is a third order Bessel type filter given by

$$B(s) = \left(\frac{b_{fn}}{s^3 + b_{fd2}s^2 + b_{fd3}s + b_{fd4}} \right). \tag{3.93}$$

Minimizing the cost function J_1 given in Eq. (3.24), the force transfer function parameters were determined for both $m_{f1} = 1$ and $m_{f1} = 3$, and for $m_{f2} = 0$. The numerical values of the parameters are listed in Table 3.3.

Using Eqs. (3.91) and (3.92), and the values listed in Table 3.3, a numerical theoretical model of the mini-manipulator force dynamics was constructed. Figures 3.13 and 3.14 show the Bode plots of the mini-manipulator force transfer functions.

For the transfer function from lateral end-point force to measured force, or $f_1(s)/f_{1d}(s)$, one sees bode plots of the experimental system, the single pole model, and the three pole model. The three pole model does a reasonably good job of matching the experimental bode plot. The single pole model does not do as good a job of modeling some of the lesser resonances, however it does do well at modeling the major torsion resonance at 10.72 Hz. Experimentation has shown that using the three pole model does not produce

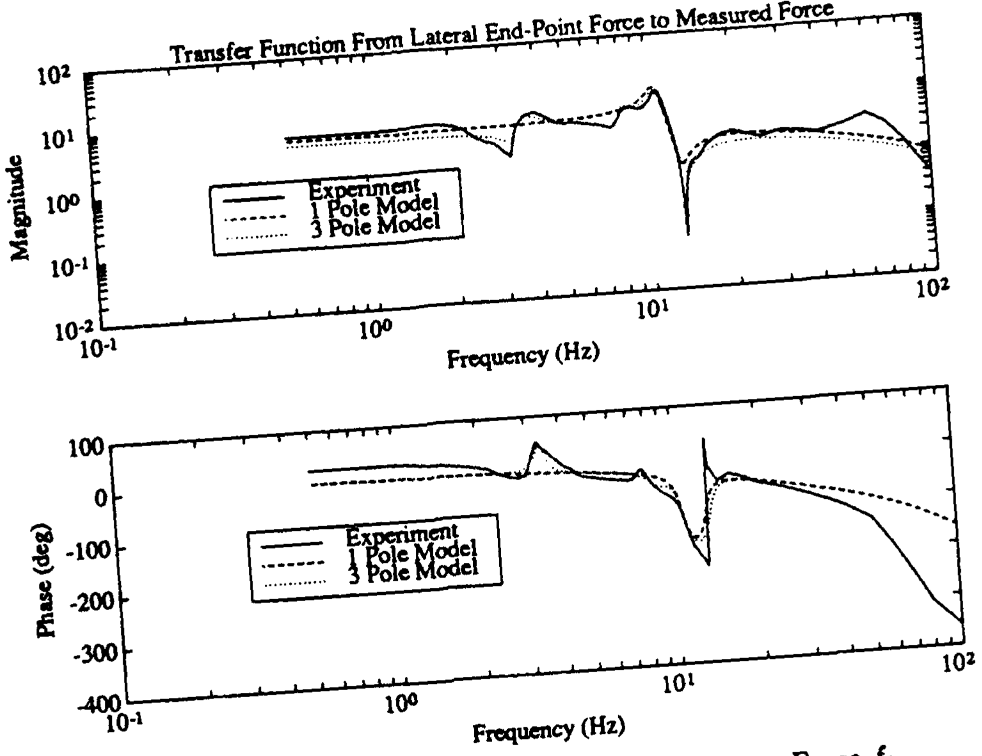

Figure 3.13: Bode Plot for Mini-Manipulator Force f_1

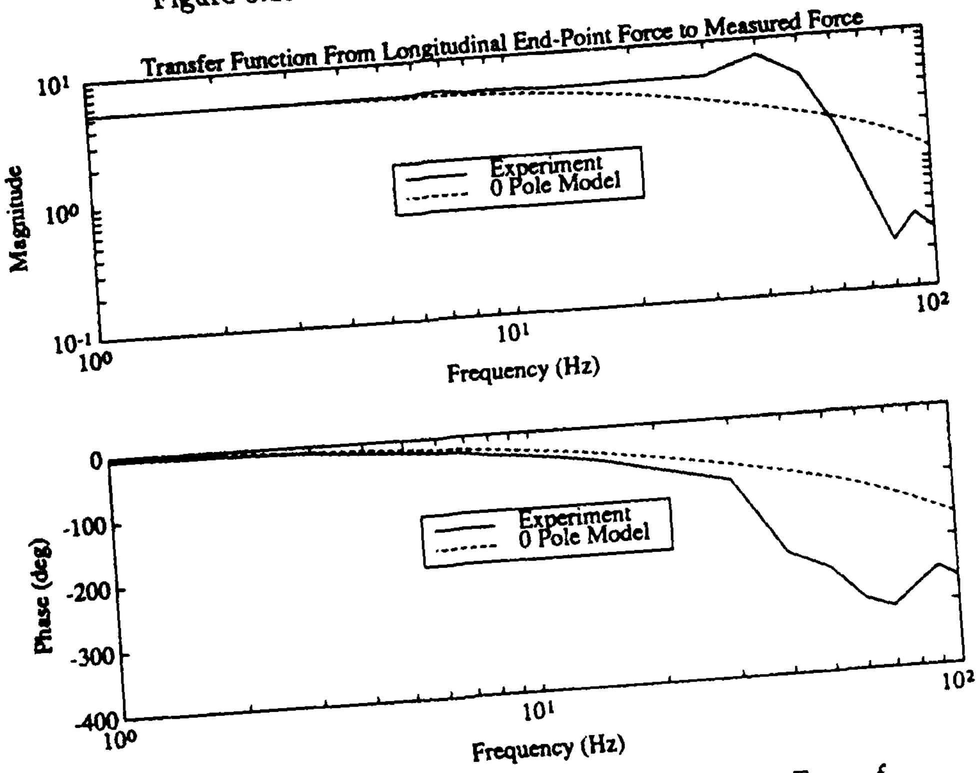

Figure 3.14: Bode Plot for Mini-Manipulator Force f_2

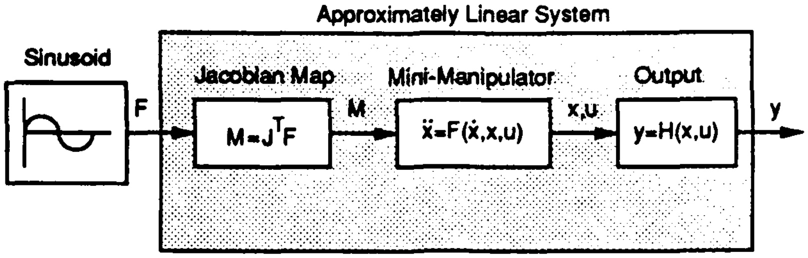

Figure 3.15: Identification by Approximate Linearization

significantly better closed-loop performance than the single pole model. For this reason, the simpler single pole model was used.

In the frequency range of interest, from DC to about 10 Hz, one can see from Fig. 3.14 that the transfer function from longitudinal end-point force to measured force, or $f_2(s)/f_{2d}(s)$, is almost a pure gain — no dynamics. Thus, for control purposes, a zero pole model is adequate.

3.9.3 Approximate Linearization Via the Jacobian

The mini-manipulator represents a highly non-linear dynamic system. In general, identifying such a non-linear system would involve measuring all of the mass, stiffness, and damping properties of the flexible arm and mini-manipulator, and using these measured values in the very complicated equations of motion that describe the system dynamics. However, as demonstrated, for a mini-manipulator that has lightweight linkages and most of its mass concentrated at the end-point, this highly non-linear system may be "mapped", via the Jacobian, into an approximately *linear* system.

Using the Jacobian, one can calculate the joint torques necessary to provide a desired end-point force for the case of static equilibrium. This is done using Eq. (3.63). If, however, the linkages are lightweight, and the end-point mass is relatively large, the joint torques calculated from Eq. (3.63) approximate the torques necessary to make the mini-manipulator end-point behave as if it is a point mass acted upon by the desired force — a linear system. Figure 3.15 shows a graphical illustration of the identification procedure.

Intuitively, the reasoning behind why the Jacobian can be used to map such a non-linear system into an approximately linear one is as follows. The full equations of motion contain many non-linear terms. Some of these are related to dynamic effects, and others are related to purely kinematic effects. As the linkages of a manipulator become more lightweight, and the end-point mass becomes larger, the dynamic non-linearities become insignificant. Thus, one is left primarily with equations of motion that contain kinematic and dynamic linear terms, and purely kinematic non-linear terms. However, these non-linear kinematic terms are exactly what the Jacobian describes, and using the Jacobian, one can essentially subtract them out of the equations of motion. This method of linearizing the mini-manipulator dynamics is generic, and can be applied to all systems that have relatively lightweight linkages, relatively large end-point masses, and a square Jacobian.

3.10 Mini-Manipulator Control Design

As with the control design of the flexible arm controller, the mini-manipulator controller was designed using modern optimal control techniques. Using the model parameters identified in Section 3.9, modal state space models of the position and force dynamics were constructed. Cost matrices and covariance matrices were chosen, and the corresponding optimal regulator and estimator were then calculated.

3.10.1 Position Control Design

3.10.1.1 State Space Model

To facilitate control design, the transfer functions described by Eqs. (3.89) and (3.90) were recast into a state space form given by

$$\begin{aligned} \dot{\boldsymbol{x}}_{rx} &= \boldsymbol{F}_{rx}\boldsymbol{x}_{rx} + \boldsymbol{G}_{rx}u_{rx} \\ y_{rx} &= \boldsymbol{H}_{rx}\boldsymbol{x}_{rx} \\ \dot{\boldsymbol{x}}_{ry} &= \boldsymbol{F}_{ry}\boldsymbol{x}_{ry} + \boldsymbol{G}_{ry}u_{ry} \\ y_{ry} &= \boldsymbol{H}_{ry}\boldsymbol{x}_{ry}. \end{aligned} \tag{3.94}$$

Using a modal representation, one can arrive at the following block diagonal form for the system dynamics matrices

$$\boldsymbol{F}_{rx} = \left[\begin{array}{cccc|c} 0 & 1 & & & \\ 0 & 0 & & & \\ & & 0 & 1 & \\ & & -\omega_{rx_1}^2 & -2\zeta_{rx_1}\omega_{rx_1} & \\ \hline \multicolumn{4}{c|}{a_{fp}\boldsymbol{h}_{rx}} & -a_{fp} \end{array}\right] \tag{3.95}$$

$$\boldsymbol{G}_{rx} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \end{bmatrix}^T \tag{3.96}$$

$$\boldsymbol{H}_{rx} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.97}$$

where

$$\boldsymbol{x}_{rx} = \begin{bmatrix} q_{rx_0} & \dot{q}_{rx_0} & q_{rx_1} & \dot{q}_{rx_1} & R_{xm} \end{bmatrix}^T \tag{3.98}$$

$$u_{rx} = f_{1d} \tag{3.99}$$

$$y_{rx} = R_{xm} \tag{3.100}$$

and

$$\boldsymbol{F}_{ry} = \left[\begin{array}{cc|c} 0 & 1 & \\ 0 & 0 & \\ \hline \multicolumn{2}{c|}{a_{fp}\boldsymbol{h}_{ry}} & -a_{fp} \end{array}\right] \tag{3.101}$$

$$\begin{aligned} \boldsymbol{G}_{ry} &= \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T && (3.102) \\ \boldsymbol{H}_{ry} &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} && (3.103) \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{x}_{ry} &= \begin{bmatrix} q_{ry0} & \dot{q}_{ry0} & R_{ym} \end{bmatrix}^T && (3.104) \\ u_{ry} &= f_{2d} && (3.105) \\ y_{ry} &= R_{ym}. && (3.106) \end{aligned}$$

The quantities R_{xm} and R_{ym} are simply the filtered measurements of R_x and R_y respectively. The filter transfer functions are given by

$$\begin{aligned} \frac{R_{xm}}{R_x} &= \left(\frac{a_{fp}}{s + a_{fp}}\right) && (3.107) \\ \frac{R_{ym}}{R_y} &= \left(\frac{a_{fp}}{s + a_{fp}}\right). && (3.108) \end{aligned}$$

The output matrices $\boldsymbol{h}_{rx}$ and $\boldsymbol{h}_{ry}$ are defined by

$$\begin{aligned} R_x &= \boldsymbol{h}_{rx}\boldsymbol{x}_{rx} && (3.109) \\ R_y &= \boldsymbol{h}_{ry}\boldsymbol{x}_{ry} && (3.110) \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{h}_{rx} &= \begin{bmatrix} r_{rx_0} & 0 & r_{rx_1} & 0 & 0 \end{bmatrix} && (3.111) \\ \boldsymbol{h}_{ry} &= \begin{bmatrix} r_{ry0} & 0 & 0 \end{bmatrix}. && (3.112) \end{aligned}$$

Again, to facilitate digital control, a discrete time model of the system was formed. The continuous time system described by Eqs. (3.94) was transformed via a zero order hold into a discrete time representation given by

$$\begin{aligned} \boldsymbol{x}_{rx_{n+1}} &= \boldsymbol{\Phi}_{rx}\boldsymbol{x}_{rx_n} + \boldsymbol{\Gamma}_{rx}u_{rx_n} \\ y_{rx_n} &= \boldsymbol{H}_{rx}\boldsymbol{x}_{rx_n} \\ \boldsymbol{x}_{ry_{n+1}} &= \boldsymbol{\Phi}_{ry}\boldsymbol{x}_{ry_n} + \boldsymbol{\Gamma}_{ry}u_{ry_n} \\ y_{ry_n} &= \boldsymbol{H}_{ry}\boldsymbol{x}_{ry_n}. \end{aligned} \tag{3.113}$$

3.10.1.2 Optimal Regulator Design

As with the design of the regulator for the flexible arm, one needs to assign some cost to state variables and controls. This cost can then be minimized subject to constraint equations — the discretized equations of motion — to yield an optimal feedback gain. The cost matrices $\boldsymbol{R}_{xxrx}$, R_{uurx}, $\boldsymbol{R}_{xxry}$, and R_{uury} were selected to yield acceptable closed-loop roots. The selections arrived at were

$$\begin{aligned} \boldsymbol{R}_{xxrx} &= \boldsymbol{h}_{rx}^T \boldsymbol{h}_{rx} && (3.114)\\ R_{uurx} &= 0.002 && (3.115)\\ \boldsymbol{R}_{xxry} &= \boldsymbol{h}_{ry}^T \boldsymbol{h}_{ry} && (3.116)\\ R_{uury} &= 0.004. && (3.117) \end{aligned}$$

To enhance damping in the system, some of the costs associated with state rates, $\dot{q}_i$'s, were modified. The modifications were

$$\begin{aligned} R_{xxrx}(2,2) &\Longleftarrow R_{xxrx}(2,2) + 5.0\\ R_{xxrx}(4,4) &\Longleftarrow R_{xxrx}(4,4) + 10.0\\ R_{xxry}(2,2) &\Longleftarrow R_{xxry}(2,2) + 5.0. \end{aligned} \qquad (3.118)$$

Using these cost matrices, the optimal feedback control law is

$$\begin{aligned} u_{rx_n} &= \boldsymbol{K}_{rx}(\boldsymbol{x}_{rxcommand} - \boldsymbol{x}_{rx_n}) && (3.119)\\ u_{ry_n} &= \boldsymbol{K}_{ry}(\boldsymbol{x}_{rycommand} - \boldsymbol{x}_{ry_n}) && (3.120) \end{aligned}$$

where $\boldsymbol{x}_{rxcommand}$ and $\boldsymbol{x}_{rycommand}$ are the desired state vectors, and the feedback gains are

$$\begin{aligned} \boldsymbol{K}_{rx} &= \begin{bmatrix} 995.95 & 54.44 & 965.50 & 29.54 & 0 \end{bmatrix} && (3.121)\\ \boldsymbol{K}_{ry} &= \begin{bmatrix} 1340.63 & 57.42 & 0 \end{bmatrix}. && (3.122) \end{aligned}$$

The actual control torques are then calculated using the Jacobian. Given u_{rx} and u_{ry}, which correspond to f_{1d} and f_{2d} respectively, the control torques are

$$\begin{bmatrix} T_A \\ T_B \end{bmatrix} = J^T \begin{bmatrix} f_{1d} \\ f_{2d} \end{bmatrix}. \qquad (3.123)$$

3.10.1.3 Steady-State Optimal Estimator Design

As with the optimal estimator for the flexible arm, the process noise and measurement noise covariances were treated as design parameters. Satisfactory estimator performance was achieved using

$$\begin{aligned} \boldsymbol{Q}_{xxrx} &= \boldsymbol{\Gamma}_{rx} \boldsymbol{\Gamma}_{rx}^T && (3.124)\\ Q_{yyrx} &= 10^{-3} && (3.125)\\ \boldsymbol{Q}_{xxry} &= \boldsymbol{\Gamma}_{ry} \boldsymbol{\Gamma}_{ry}^T && (3.126)\\ Q_{yyry} &= 10^{-3}. && (3.127) \end{aligned}$$

Using the above covariance matrices, the steady-state, optimal estimator, feedback gains were found to be

$$L_{rx} = \begin{bmatrix} 0.0088 & 0.2189 & -0.0811 & 0.7031 \end{bmatrix}^T \tag{3.128}$$

$$L_{ry} = \begin{bmatrix} 0.0076 & 0.2057 & 0.8225 \end{bmatrix}^T. \tag{3.129}$$

Due to calculation time delays resulting from signal processing, a predictor estimator was used. Thus, the optimal estimate of the state vectors x_{rx} and x_{ry} may be found via the equations

$$\hat{x}_{rx_{n+1}} = \Phi_{rx} x_{rx_n} + \Gamma_{rx} u_{rx_n} + L_{rx}(y_{rx_n} - H_{rx} x_{rx_n}) \tag{3.130}$$

$$\hat{x}_{ry_{n+1}} = \Phi_{ry} x_{ry_n} + \Gamma_{ry} u_{ry_n} + L_{ry}(y_{ry_n} - H_{ry} x_{ry_n}). \tag{3.131}$$

3.10.2 Force Control Design

3.10.2.1 State Space Model

Again, it is convenient to recast Eqs. (3.91) and (3.92) into a modal state space form.

$$\begin{aligned} \dot{x}_{f1} &= F_{f1} x_{f1} + G_{f1} u_{f1} \\ y_{f1} &= H_{f1} x_{f1} \\ \dot{x}_{f2} &= F_{f2} x_{f2} + G_{f2} u_{f2} \\ y_{f2} &= H_{f2} x_{f2}. \end{aligned} \tag{3.132}$$

In the modal representation, the dynamics matrices are

$$F_{f1} = \left[\begin{array}{cc|ccc} 0 & 1 & & & \\ -\omega_{f1_1}^2 & -2\zeta_{f1_1}\omega_{f1_1} & & & \\ \hline & & 0 & 1 & 0 \\ & & 0 & 0 & 1 \\ 1 & 0 & -b_{fd4} & -b_{fd3} & -b_{fd2} \end{array} \right] \tag{3.133}$$

$$G_{f1} = \begin{bmatrix} 0 & r_{f1_1} & 0 & 0 & r_{f1_0} \end{bmatrix}^T \tag{3.134}$$

$$H_{f1} = \begin{bmatrix} 0 & 0 & b_{fn} & 0 & 0 \end{bmatrix} \tag{3.135}$$

where

$$x_{f1} = \begin{bmatrix} q_{f1} & \dot{q}_{f1} & \xi_{f1m} & \dot{\xi}_{f1m} & \ddot{\xi}_{f1m} \end{bmatrix}^T \tag{3.136}$$

$$u_{f1} = f_{1d} \tag{3.137}$$

$$y_{f1} = f_{1m} \tag{3.138}$$

and

$$F_{f2} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -b_{fd4} & -b_{fd3} & -b_{fd2} \end{bmatrix} \tag{3.139}$$

$$\begin{aligned} \boldsymbol{G}_{f2} &= \begin{bmatrix} 0 & 0 & r_{f2_0} \end{bmatrix}^T && (3.140) \\ \boldsymbol{H}_{f2} &= \begin{bmatrix} b_{fn} & 0 & 0 \end{bmatrix} && (3.141) \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{x}_{f2} &= \begin{bmatrix} \xi_{f2m} & \dot{\xi}_{f2m} & \ddot{\xi}_{f2m} \end{bmatrix}^T && (3.142) \\ u_{f2} &= f_{2d} && (3.143) \\ y_{f2} &= f_{2m}. && (3.144) \end{aligned}$$

The quantities f_{1m} and f_{2m} are simply the filtered measurements of f_1 and f_2 respectively. The filter transfer functions are given by

$$\frac{f_{1m}(s)}{f_1(s)} = \frac{f_{2m(s)}}{f_{2(s)}} = \frac{b_{fn}}{s^3 + b_{fd2}s^2 + b_{fd3}s + b_{fd4}}. \qquad (3.145)$$

The continuous time system described by Eqs. (3.132) was transformed via a zero order hold into a discrete time representation given by

$$\begin{aligned} \boldsymbol{x}_{f1_{n+1}} &= \boldsymbol{\Phi}_{f1}\boldsymbol{x}_{f1_n} + \boldsymbol{\Gamma}_{rx}u_{f1_n} \\ y_{f1_n} &= \boldsymbol{H}_{f1}\boldsymbol{x}_{f1_n} \\ \boldsymbol{x}_{f2_{n+1}} &= \boldsymbol{\Phi}_{f2}\boldsymbol{x}_{f2_n} + \boldsymbol{\Gamma}_{f2}u_{f2_n} && (3.146) \\ y_{f2_n} &= \boldsymbol{H}_{f2}\boldsymbol{x}_{f2_n}. \end{aligned}$$

3.10.2.2 Optimal Regulator Design

Since the force transfer functions behave much like a pure gain at low frequency, it is desirable to include integral control to eliminate steady state tracking error to step commands, and to provide good disturbance rejection. To facilitate the addition of integral control, the state vector may be augmented with an integral of the measured forces.

In discrete time, the integral of the force may be expressed by

$$\begin{aligned} e_{f1_{n+1}} &= e_{f1_n} + \tau f_{1m_n} = e_{f1_n} + \tau\boldsymbol{H}_{f1}\boldsymbol{x}_{f1_n} && (3.147) \\ e_{f2_{n+1}} &= e_{f2_n} + \tau f_{2m_n} = e_{f2_n} + \tau\boldsymbol{H}_{f2}\boldsymbol{x}_{f2_n} && (3.148) \end{aligned}$$

where τ is the sample period — 0.01 seconds for the sample frequency of 100 Hz. Thus the augmented state vector may be written as

$$\begin{aligned} \boldsymbol{x}'_{f1_n} &= \left[\frac{\boldsymbol{x}_{f1_n}}{e_{f1_n}}\right] && (3.149) \\ \boldsymbol{x}'_{f2_n} &= \left[\frac{\boldsymbol{x}_{f2_n}}{e_{f2_n}}\right]. && (3.150) \end{aligned}$$

We may now define a new system that contains the augmented state vectors $\boldsymbol{x}'_{f1}$ and $\boldsymbol{x}'_{f2}$. The new discrete time dynamics matrices are:

$$\boldsymbol{\Phi}'_{f1} = \left[\begin{array}{c|c} \boldsymbol{\Phi}_{f1} & 0 \\ \hline \tau \boldsymbol{H}_{f1} & 1 \end{array}\right] \tag{3.151}$$

$$\boldsymbol{\Gamma}'_{f1} = \left[\begin{array}{c} \boldsymbol{\Gamma}_{f1} \\ \hline 0 \end{array}\right] \tag{3.152}$$

$$\boldsymbol{H}'_{f1} = \left[\begin{array}{c|c} \boldsymbol{H}_{f1} & 0 \end{array}\right] \tag{3.153}$$

$$\boldsymbol{\Phi}'_{f2} = \left[\begin{array}{c|c} \boldsymbol{\Phi}_{f2} & 0 \\ \hline \tau \boldsymbol{H}_{f2} & 1 \end{array}\right] \tag{3.154}$$

$$\boldsymbol{\Gamma}'_{f2} = \left[\begin{array}{c} \boldsymbol{\Gamma}_{f2} \\ \hline 0 \end{array}\right] \tag{3.155}$$

$$\boldsymbol{H}'_{f1} = \left[\begin{array}{c|c} \boldsymbol{H}_{f1} & 0 \end{array}\right]. \tag{3.156}$$

We will now assign a cost to the augmented state vectors and controls to arrive at an optimal controller that minimizes these costs. For the lateral force controller which regulates f_1, it is desirable to assign a cost to both the torsion mode and to the integral of the force. For the longitudinal force controller which regulates f_2, it is desirable to assign a cost to the integral of the force. Thus, the following cost matrices were arrived at:

$$\boldsymbol{R}_{xxf1} = \begin{bmatrix} 0.003 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.157}$$

$$\boldsymbol{R}_{uuf1} = 0.015 \tag{3.158}$$

$$\boldsymbol{R}_{xxf2} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.159}$$

$$\boldsymbol{R}_{uuf1} = 0.020. \tag{3.160}$$

The optimal feedback control law is then

$$u_{f1_n} = \boldsymbol{K}_{f1}(\boldsymbol{x}_{f1command} - \boldsymbol{x}_{f1_n}) \tag{3.161}$$

$$u_{f2_n} = \boldsymbol{K}_{f2}(\boldsymbol{x}_{f2command} - \boldsymbol{x}_{f2_n}) \tag{3.162}$$

where $\boldsymbol{x}_{f1command}$ and $\boldsymbol{x}_{f2command}$ are the desired state vectors, and the feedback gains are

$$\boldsymbol{K}_{f1} = \begin{bmatrix} 0.1466 \\ 0.0047 \\ 4.5666 \times 10^{6} \\ 1.7084 \times 10^{3} \\ 2.8536 \\ 5.7886 \end{bmatrix}^{T} \tag{3.163}$$

$$\boldsymbol{K}_{f2} = \begin{bmatrix} 4.6502 \times 10^{6} \\ 1.7037 \times 10^{3} \\ 2.8530 \\ 5.9288 \end{bmatrix}^{T} . \tag{3.164}$$

As in position control, the control torques are calculated using the Jacobian. Given $\boldsymbol{u}_{f1}$ and $\boldsymbol{u}_{f2}$, which correspond to f_{1d} and f_{2d} respectively, the control torques are given by Eqs. (3.123).

3.10.2.3 Steady-State Optimal Estimator Design

Using process noise and measurement noise covariance matrices given by

$$\boldsymbol{Q}_{xxf1} = \boldsymbol{\Gamma}_{f1}\boldsymbol{\Gamma}_{f1}^{T} \tag{3.165}$$

$$Q_{yyf1} = 10^{-3} \tag{3.166}$$

$$\boldsymbol{Q}_{xxf2} = \boldsymbol{\Gamma}_{f2}\boldsymbol{\Gamma}_{f2}^{T} \tag{3.167}$$

$$Q_{yyf2} = 10^{-3}, \tag{3.168}$$

satisfactory estimator performance was achieved. The corresponding steady-state, optimal estimator, feedback gains were

$$\boldsymbol{L}_{f1} = \begin{bmatrix} 0.3215 \\ 12.4454 \\ 7.5940 \times 10^{-9} \\ -3.7997 \times 10^{-7} \\ 1.6615 \times 10^{-4} \\ 0 \end{bmatrix} \tag{3.169}$$

$$\boldsymbol{L}_{f2} = \begin{bmatrix} 4.6368 \times 10^{-9} \\ -6.9493 \times 10^{-7} \\ 1.7783 \times 10^{-4} \\ 0 \end{bmatrix} . \tag{3.170}$$

Again, due to computation time delays resulting from signal processing, a predictor estimator was implemented to estimate the force states. Thus, the optimal estimate of the state vectors $\boldsymbol{x}_{f1}$ and $\boldsymbol{x}_{f2}$ may be found via the equations

$$\hat{\boldsymbol{x}}_{f1_{n+1}} = \boldsymbol{\Phi}_{f1}\boldsymbol{x}_{f1_n} + \boldsymbol{\Gamma}_{f1}u_{f1_n} + \boldsymbol{L}_{f1}(y_{f1_n} - \boldsymbol{H}_{f1}\boldsymbol{x}_{f1_n}) \tag{3.171}$$

$$\hat{\boldsymbol{x}}_{f2_{n+1}} = \boldsymbol{\Phi}_{f2}\boldsymbol{x}_{f2_n} + \boldsymbol{\Gamma}_{f2}u_{f2_n} + \boldsymbol{L}_{f2}(y_{f2_n} - \boldsymbol{H}_{f2}\boldsymbol{x}_{f2_n}). \tag{3.172}$$

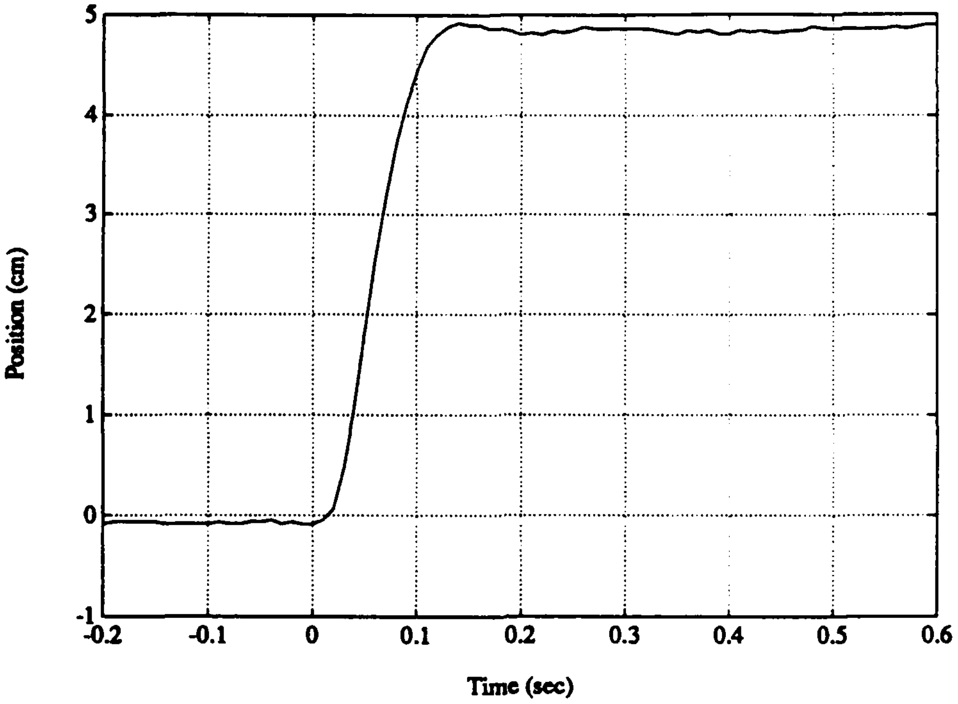

Figure 3.16: Position Control Step Response for the Mini-Manipulator

3.11 End-Point Control of the Mini-Manipulator

Using the optimal regulators and estimators designed in Sections 3.7 and 3.10, end-point position and force control of the flexible arm and mini-manipulator system is now possible. The position controller for the flexible arm and the position and force controller for the mini-manipulator were designed separately. This was made possible by the fact that the flexible arm and the mini-manipulator are bandwidth separated — the mini-manipulator is about six times as fast as the flexible arm.

Running the controller at a sample rate of 100 Hz, the mini-manipulator was given a step command in end-point position. Figure 3.16 shows the time history of the end-point of the mini-manipulator. The roughly 5 cm step takes 0.11 seconds to complete. In comparison, the flexible arm required about 0.65 seconds to complete a 5 cm slew.

Next, the mini-manipulator was given a step command in end-point force. In Fig. 3.17 we see a time history of the force level during this step command. It takes about 0.06 seconds to complete a transition in force level from 21 grams force to 35 grams force. In comparison, the time required to achieve a step change in force level using the earlier flexible arm system, which did not have a mini-manipulator, was about 0.7 seconds.

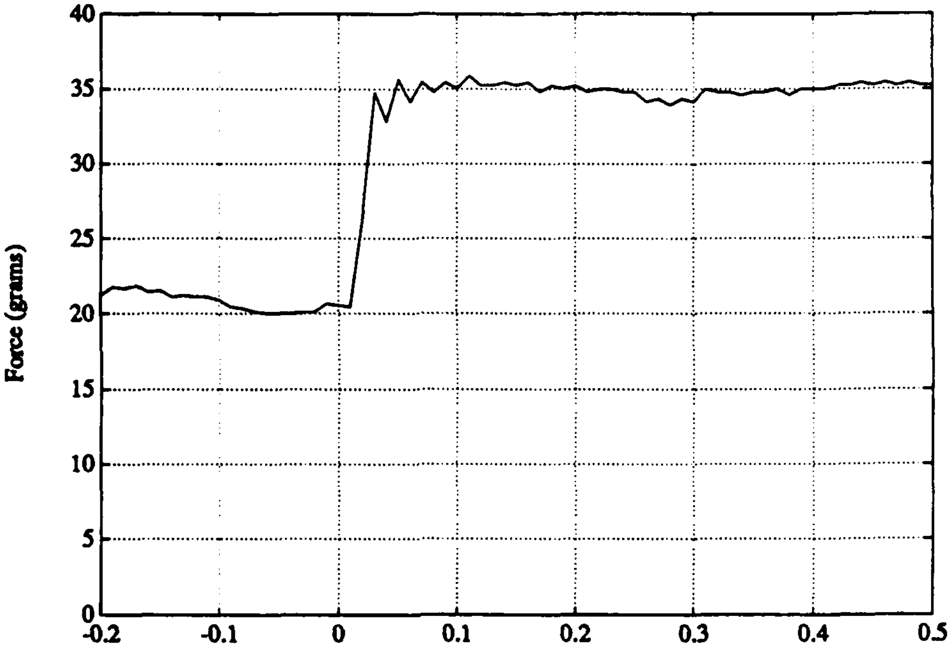

Figure 3.17: Force Control Step Response for the Mini-Manipulator

3.12 Surface Following

As humans, we can appreciate the value of being able to follow the contours of surfaces using touch. It is an ability that is useful in tasks such as cleaning windows, mopping floors, and sanding wood. A robot with this ability would hence be very valuable. Since the flexible arm and mini-manipulator system possess a sense of touch, namely the force sensing beam, it is possible to demonstrate a surface following ability.

When the mini-manipulator end-point comes into contact with an object, it senses a force. Because the force sensing beam is equipped with a contact roller, the forces sensed are ideally only normal forces. Thus, given the direction of the contact force, one knows that the surface normal lies in the opposite direction. The surface tangent is then orthogonal to this. Thus, one knows the local direction of the surface, and it is possible to move in this direction while applying forces in the surface normal direction.

The surface following algorithm that was implemented is as follows. At time t_0, determine the surface normal and tangent directions. Next, command a position that lies a distance k in the tangent direction. Also, command a force in the direction opposite to the surface normal. Now, at a time t_1 that is one sample period after t_0, determine the new surface normal and tangent directions. Project the old commanded position onto a line that is parallel to the new surface tangent and passes through the current contact point. To this projected position, step a distance k in the tangent direction and define this as the new commanded position. Again, command a force in the direction opposite to the current surface normal. Repeat this every sample period.

One may also state this algorithm in a more mathematical form. If the current applied force vector is

$$f = f_x n_1 + f_y n_2, \tag{3.173}$$

then the current surface normal vector is

$$\begin{aligned} t &= t_x n_1 + t_y n_2 \\ &= \frac{f_y}{|f|} n_1 - \frac{f_x}{|f|} n_2. \end{aligned} \tag{3.174}$$

Let (p_x, p_y) be the Cartesian coordinates of the current end-point position, and let (p_{xc1}, p_{yc1}) be the Cartesian coordinates of the last commanded position. Then the projection of the last commanded position on a line passing through (p_x, p_y) and parallel to t is

$$l = (p_{xc1} - p_x)t_x + (p_{yc1} - p_y)t_y. \tag{3.175}$$

Finally Cartesian coordinates of the next commanded position, (p_{xc2}, p_{yc2}), are given by

$$p_{xc2} = p_x + (l + k)t_x \tag{3.176}$$

$$p_{yc2} = p_y + (l + k)t_y, \tag{3.177}$$

where k is a constant.

This algorithm involves position control in one direction, and force control in an orthogonal direction. One will recall that the position controller generates a control effort u_{rx} corresponding to f_{1d}, and u_{ry} corresponding to f_{2d}. Let $f_{1d_{pos}}$ and $f_{2d_{pos}}$ be the control efforts determined by the position controller. The force controller generates a control effort u_{f1} corresponding to f_{1d}, and u_{f2} corresponding to f_{2d}. Let $f_{1d_{for}}$ and $f_{2d_{for}}$ be the control efforts determined by the force controller. The combined position and force control effort is then

$$f_{1d_{combined}} = f_{1d_{pos}} + f_{1d_{for}} \tag{3.178}$$

$$f_{2d_{combined}} = f_{2d_{pos}} + f_{2d_{for}}. \tag{3.179}$$

One then uses the relation

$$\begin{bmatrix} T_A \\ T_B \end{bmatrix} = J^T \begin{bmatrix} f_{1d_{combined}} \\ f_{2d_{combined}} \end{bmatrix} \tag{3.180}$$

to determine the joint torques.

Using this algorithm, the flexible arm was commanded to follow the contours of a target at a speed of 4 cm/sec. Figure 3.18 shows an experimental plot describing this maneuver. The solid line describes the path made by the end-point of the mini-manipulator, and essentially outlines the shape of the target surface. The many short, straight lines emanating from the target surface describe what the force controller estimated the local surface normal to be. These estimates are shown every 0.2 seconds. Based on measurements of link angles and LED positions, the configuration of the flexible arm and mini-manipulator is shown every 3.0 seconds.

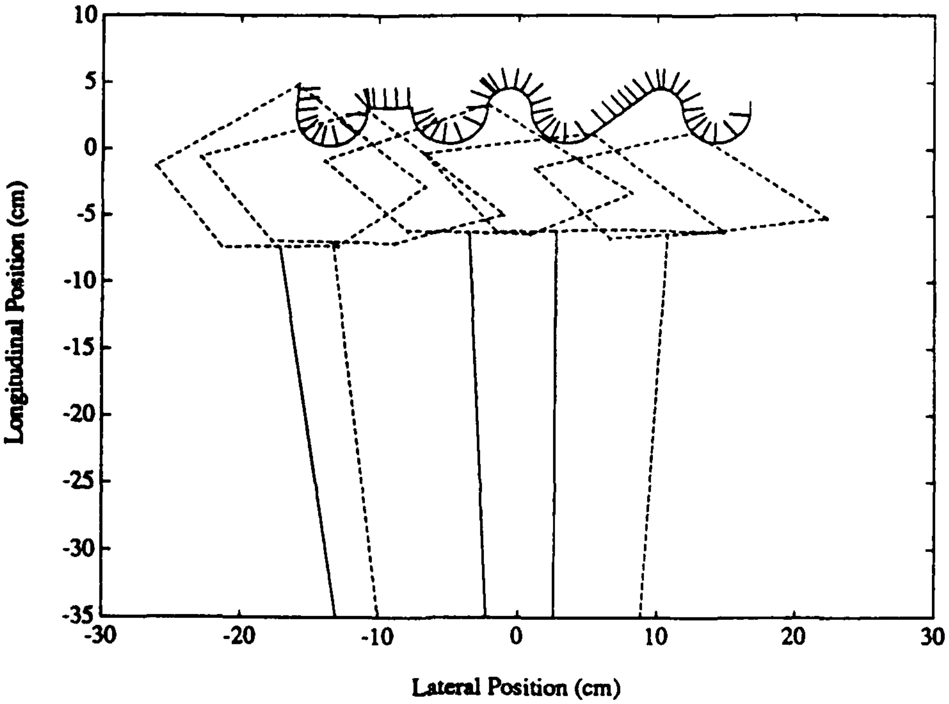

Figure 3.18: Surface Following

3.13 Conclusion

End-point position and force control of a flexible arm with a two degree-of-freedom mini-manipulator has been demonstrated. The end-point position of the flexible arm can be controlled at a bandwidth of 1.5 Hz — this compares quite favorably with the first cantilever mode of the flexible arm which is 0.7 Hz.

The torsional interactions of the mini-manipulator and the flexible arm were modeled. From this, it was determined that the original system could be modified to greatly simplify position control by providing a favorable pole-zero configuration. At the same time, it was shown that there was no convenient means of achieving this favorable pole-zero configuration for force control.

A generic means of mapping a non-linear system into an approximately linear one was demonstrated. This approach is applicable to any system with a square Jacobian, and linkages that are light compared to the end-point mass. In this experiment, a non-linear mini-manipulator was mapped into an approximately linear system. This linearization facilitated the use of standard techniques for identifying and controlling linear systems.

Using optimal control techniques, an end-point position and force controller were developed for the mini-manipulator. These controllers took advantage of the fact that the flexible arm and the mini-manipulator could be bandwidth separated. This made it possible to design controllers for the mini-manipulator and the flexible arm separately.

The resulting bandwidth for position and force control at the end-point of the mini-manipulator was about 10 Hz — roughly ten times better than for a flexible arm without a mini-manipulator.

Finally, the ability to follow the contours of a surface, without a priori knowledge of its shape, was demonstrated. This involved having the end-point of the mini-manipulator "feel" its way around an object. It also demonstrated the ability to negotiate sharp, right angle changes in surface orientation. The surface following algorithm is insensitive to slow motions of the target object.

3.14 Status

The work described in this chapter is in a state of near completion. The original goals of demonstrating position and force control, as well as surface following, have been ccomplished. This work is currently being written up as a PhD thesis.

Chapter 4

Force Control of a Two-Link Arm With Flexible Drive Train

Brian Andersen

Introduction

A crucial capability for two-arm cooperation will be exquisite force control at each arm tip. The supporting research effort described in this chapter is aimed at advancing that capability in the context of a single two-link arm having a very flexible drive train.

This section of the final report describes work that has been accomplished to date on the two-link arm with flexible tendon drive and mini-manipulator. The objective of this research is to allow the arm to slew into contact with a target, make a smooth touchdown, and maintain a constant force on the target, using the mini-manipulator for augmentation of precision control. We believe the mini-manipulator is going to play a key role in successful high-performance force control. In this chapter we describe its use on a two-link arm. Chapter 4 describes additional fundamental force-control experiments with the two-degree-of-freedom mini-manipulator mounted on a single, very flexible beam.

During the time covered by this report we have demonstrated initial position and force control with the mini-manipulator on a fixed base. This work will form the basis for the control algorithms when the mini-manipulator is added to the two-link arm.

This is a facility which was completed three years ago, and with which we have already demonstrated precise, rapid control of very large pick-and-place tasks with heavy targets of varying weight. A schematic representation of the two-link-arm apparatus is shown in Fig. 4.1, in which the mini-manipulator is not shown to scale. The main arm actually has approximately six times the reach of the mini-manipulator.

The rationale for using a mini-manipulator at the tip of the two-link arm is that, for most robotic tasks, a great deal of the manipulation is performed in a few localized workspaces. A mini-manipulator can be used to perform these tasks in the localized

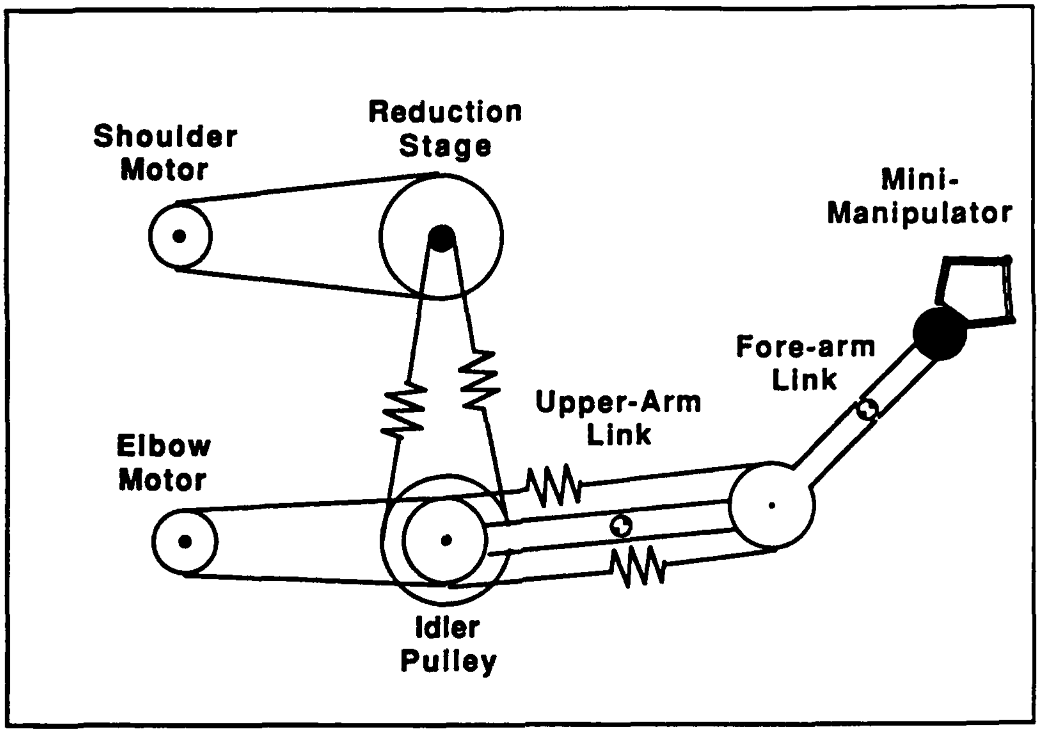

Figure 4.1: The Two-Link Arm with Mini-Manipulator

workspace at a much higher bandwidth than is achievable by the main arm. The main arm is used to move the mini-manipulator between the localized workspaces.

The status of this project is that the experimental hardware has been built and is functional. The computer hardware and interface electronics have also been built and are functional. Initial experiments with the mini-manipulator on a fixed base have demonstrated position and force control. Using the same experimental set-up, following a surface while maintaining a desired force has also been demonstrated. This work can be extended when the mini-manipulator is added to the two-link arm.

This chapter of the report describes the various demonstrations that will be performed to test the control methods developed, the design of the experimental hardware, the various position and force control algorithms that have been developed, results from their experimental implementation, and future plans for this project.

4.1 Objective of this Project

The capability we wish to develop here is for very-high-bandwidth, precise control of the force at the tip of a two-link manipulator having a very flexible drive train. We plan to demonstrate this capability by (1) performing rapid slew and touch with a fixed target while having no overshoot in the force, (2) controlling the arm so that its tip moves along a wavy surface while maintaining a constant force on the surface, and (3)

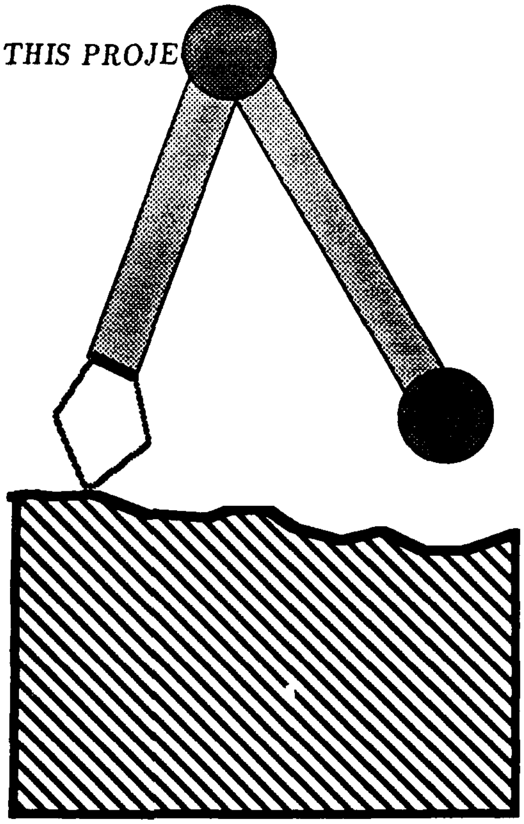

Figure 4.2: Demonstration of Force Control Capabilities

slewing into contact with a moving target and maintaining a constant force on it while it continues to move.

The initial experiments will be performed with some massive object for a target that will not move when the arm tip exerts a force against it. The important elements to demonstrate are that the arm is controlled accurately both in and out of contact with the environment, and that there is quick switching between modes of control. This will be preceded by performing the same experiment with the mini-manipulator on a fixed base.

The next demonstration, shown in Fig. 4.2, requires a massive target which will have surface waves of the order of magnitude of the range of the mini-manipulator. The arm will be controlled to maintain a constant force while following the surface. The main arm will be controlled to track primarily along a straight line along the target while the mini-manipulator follows the waves in the target surface.

This demonstration will also be performed with the mini-manipulator on a fixed base. In some ways, performing this task on a fixed base is more difficult. When the mini-manipulator is attached to the main arm, the tip of the main arm can be moved so that the mini-manipulator is in the center of its workspace. When the mini-manipulator is on a fixed base, however, it can only apply the desired force wherever the target is in its workspace.

The final demonstration, shown in Fig. 4.3, will address controlling force applied to a moving target. We plan to use as a target a model railroad train similar to one that has been developed for another experiment and which should require little if any

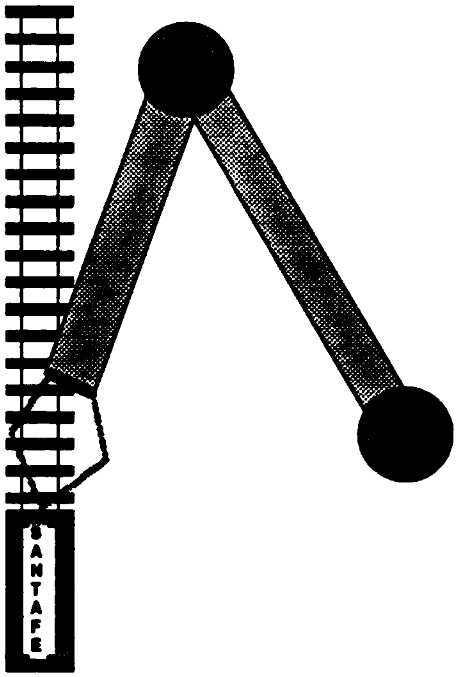

Figure 4.3: Demonstration of Force Control Capabilities

modification. This will demonstrate the ability to follow a moving surface in a large-disturbance environment.

4.2 Experimental Apparatus

4.2.1 Two-Link Arm

The experimental two-link manipulator is a SCARA manipulator operating in the horizontal plane with two revolute joints and vertical plunge actuator at the tip. A drawing of the manipulator is shown in Fig. 4.4 with the base, supports, and mini-manipulator removed to show the essential features. Both of the links of the manipulator are approximately $0.5m$ in length. The reach of the manipulator ranges from a minimum of about $0.5m$ to a maximum of about $1.0m$. The vertical plunge actuator gives a range of motion of about $15.0cm$.

The links of the manipulator are driven by two identical DC electric motors with maximum torque capability of $11.2N - m$. The upper link is connected to the shoulder motor via a reduction stage and a cable drive with four springs mounted in-link to provide for exaggerated flexibility. Similarly, the forearm is connected to the elbow motor via an idler pulley and a cable drive with two springs mounted in-line. The shoulder drive train provides a 5.91:1 gear reduction, and the elbow drive train provides a 2.44:1 gear reduction.

The motors, both joints of the manipulator and the vertical plunge actuator are

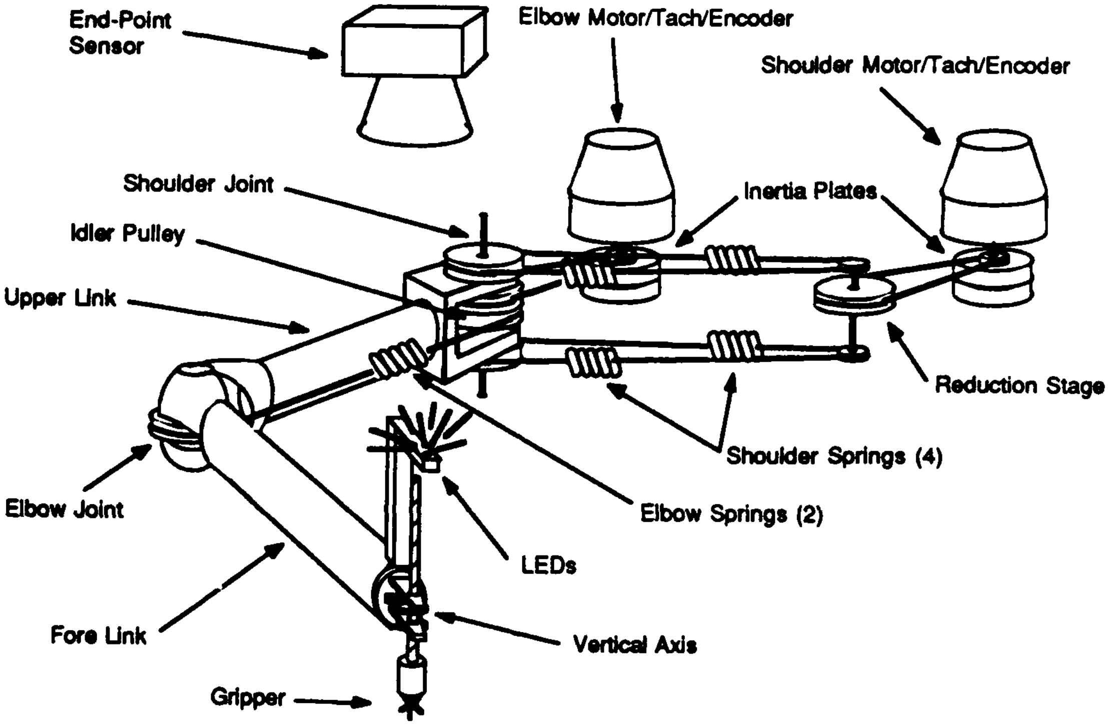

Figure 4.4: Drawing of the Experimental Two-Link Manipulator

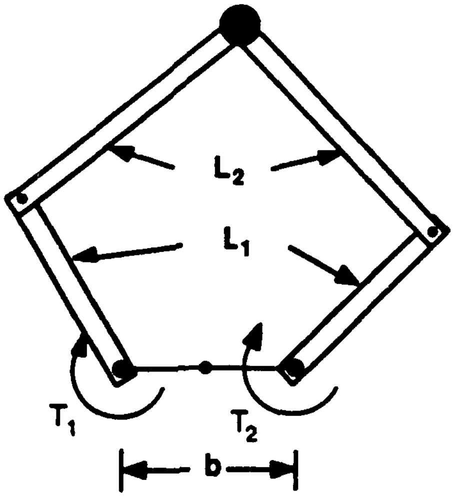

Figure 4.5: The Mini-Manipulator

equipped with rotary encoders to provide angular position information. These encoder signals are processed electronically, using a first difference at low rates or a period count at high rates, to give angular velocity information. In addition, each motor has a tachometer to provide angular velocity information. Finally, there is an optical sensor that senses center of brightness. This observes the triad of infrared LEDs mounted on the top of the vertical plunge actuator to determine the end-point position of the manipulator.

4.2.2 Mini-Manipulator

The mini-manipulator is a small five-link, closed-kinematic-chain manipulator which operates in a horizontal plane. It is shown schematically in Fig. 4.5. The connection between the two-link manipulator and the mini-manipulator is shown in greater detail in Fig. 4.6. The base link, which is rigidly attached to the vertical plunge actuator of the two-link manipulator, is approximately $5.1cm$ in length. The two inner links of the mini-manipulator are approximately $7.6cm$ in length and the two outer links are approximately $10.2cm$ in length. The inner and outer links are hollow tubes for light weight and fast speed. At the end of the outer links where they are connected through a revolute joint, there is a vertical beam with a square cross section. At the bottom end of this beam is a bearing enclosed by a hard rubber ring. It is the rubber ring that makes contact with the environment.

The two inner links of the mini-manipulator are driven relative to the base link by two small DC electric motors. These motors have a peak torque of $0.35N - m$.

Mounted on top of the mini-manipulators are rotary encoders to provide angular position information. These encoder signals are processed using the same electronics as the two-link arm encoders to give angular velocity information.

The vertical beam at the tip of the mini-manipulator is equipped with two pairs of

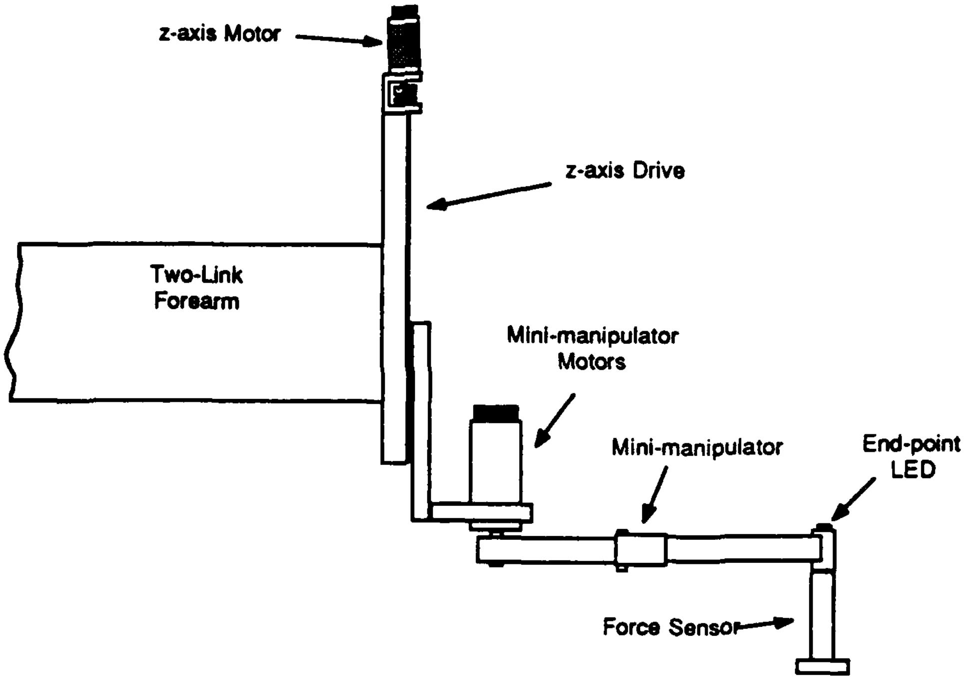

Figure 4.6: Connection of the Two-Link Arm with Mini-Manipulator

strain gages near the top of the beam. One gage is mounted on each face of the beam. Each pair of gages on opposite faces of the beam is connected in a bridge configuration with two resistors on a strain-gage amplifier board to produce a signal proportional to the force applied in one direction with only a small amount of noise. These gages and electronics are used to produce an estimate of the force applied at the tip of the mini-manipulator in two perpendicular directions in the horizontal plane.

4.2.3 Control Computer and Interface Electronics

The real-time control of the mechanical system is handled by a Motorola 68020 single-board processor with a 68881 floating point coprocessor. Both operate at $16.67MHz$. This board is connected via a bus repeater to a Sun 3/160 workstation using the VME Bus. Using this connection, software can be developed and debugged in the flexible development environment of the Sun workstation and then down-loaded to the 68020 processor. Using software developed in this laboratory, the 68020 processor controls the experimental hardware while sending data through the VME Bus back to the Sun workstation for display and processing.

Control signals are output from the computer using the VMIVME-4116 8-channel 16-bit Digital-to-Analog Converter. Analog signals, such as the force sensor and the motor tachometers, are sampled by the XVME-566 Analog-to-Digital Converter that provides 16 differential input channels with 12-bit resolution. Finally, the MVME340

Parallel Interface/Timer Module drives the encoder interface electronics built in the laboratory and provides for output of discrete signals such as a motor heartbeat. All three of these boards are connected to the 68020 processor via a VME Bus backplane.

4.3 Analysis and Controller Design

4.3.1 Equations of Motion

The equations of motion of the two-link arm with mini-manipulator and the mini-manipulator alone, both in and out of contact with its environment, have been determined. These equations will be used for simulation and for comparison with the actual system.

The equations were derived using Kane's dynamics. It includes intermediate pulleys between the motors and the joints of the main arm. Friction is included in the motors, springs, and joints of the main arm. It is not included in the joints of the mini-manipulator since that is considered to be small compared to the friction in the rest of the system. This could be included later without much difficulty. The environment is modelled as a spring and a linear damper attached at one end to a point of contact on the target and to the tip of the arm on the other end. This models any elasticity in the target, force sensor, or both. A large spring constant for the environment can be used if it is desired to model the target surface as rigid.

The system as modelled has eight rigid bodies (two main arm motors, two main arm links, and only four mini-manipulator links since the base link is rigidly attached to the fore arm link) and four constraint equations (the matching of position and velocity of the ends of the two outer links of the mini-manipulator), for a total of twelve degrees of freedom for the system. Due to the complexity of these twelve equations of motion, they are not presented here.

The equations of motion for both the mini-manipulator alone and the mini-manipulator on the end of the two-link arm have been incorporated into a simulation. The simulation is written as subroutines th are linked into MatrixX/System Build, a non-linear system simulation package.

4.3.2 Control Methods

The combination of the main two-link arm and the mini-manipulator makes an intricate dynamic system with very complicated equations of motion. Rather than try to develop a controller design that can handle the complicated dynamics of the whole system, we take advantage of the fundamental dynamic behavior of the system. The mini-manipulator is designed to be smaller and quicker than the main two-link arm, so its control bandwidth will be much larger. We can take advantage of this spectral separation of the two systems to design controllers for the two systems independently. The main arm will have a slower controller designed without considering the dynamics of the mini-manipulator. This is a good approximation since the mini-manipulator dynamics have a small effect on the dynamics of the main arm.

The mini-manipulator will have a control algorithm designed independently of the dynamics of the main arm. However, because the dynamics of the main arm are strongly coupled with the dynamics of the mini-manipulator, the motion of the main arm must be taken into effect in some way. This probably can be accomplished by simply using the kinematic quantities of the tip of the main arm, such as position, velocity, and possibly acceleration. The exact method has yet to be determined.

4.3.2.1 Main Arm

Control of the main arm will draw on the work of Michael Hollars from this laboratory. He demonstrated high-bandwidth position control for the tip of the main arm carrying a variety of payload masses, as described in [5]. His control algorithm involved designing a Linear Quadratic Regulator (LQR) for the system linearized about an elbow angle of 35°. Attempts to use a Kalman Filter for the estimator failed, however, due to the large non-linearities in the system. Instead of a linear estimator, a method called the Constant Gain Extended Kalman Filter (CGEKF) was used. This method uses the same linear measurement update as the usual Kalman Filter. Instead of using a linear state update, though, it integrates the full equations of motion of the system to arrive at the new state estimate. This method has proved to work very well for the two-link arm with no mini-manipulator while not in contact with its environment.

Specifically, the CGEKF was implemented in three equations: a measurement update:

$$\hat{x}(k) = \bar{x}(k) + L_d(y(k) - C_d\bar{x}(k)) \tag{4.1}$$

a time update:

$$\bar{x}(k+1) = \int_{t(k)}^{t(k+1)} f(\hat{x}(k), u(k))dt \tag{4.2}$$

and finally the control equation:

$$u(k+1) = K_d(x_c(k+1) - \bar{x}(k+1)) \tag{4.3}$$

where $\hat{x}(k)$ is the estimated value of x at the current time, $\bar{x}(k+1)$ is the predicted value at the next time, L_d is the estimator gain matrix, $y(k)$ is the current measurement, C_d is the matrix that converts states to measured values, $u(k)$ is the control torque, K_d is the controller gain matrix, and $\dot{x} = f(x(k), u(k))$ are the nonlinear equations of motion of the system.

4.3.2.2 Mini-Manipulator

To date, there have been three control algorithms tested experimentally using the mini-manipulator on a fixed base. The first two of these are methods for controlling the position of the tip, and the third controls the force applied by the tip.

The first control algorithm tested was a simple Proportional-Derivative (PD) control loop, using the motor angles. This scheme indirectly controls tip position by using the inverse kinematics of the mini-manipulator to determine the angles that give the desired tip position. It does not, however, address the motion of the tip through the workspace.

The second control algorithm tested was an impedance controller. This control algorithm was originally described by Hogan in [3]. This control scheme takes into account the kinematics of the mini-manipulator so that the tip is controlled directly in Cartesian space. However, it does not consider the dynamics of the mini-manipulator, so it does not compensate for dynamic effects such as centrifugal acceleration. It can be considered in a somewhat simplistic view as a PD controller of the tip position and velocity.

An impedance controller establishes a fixed relation between the position of the tip and the force on the tip. In this case, the relationship is

$$F = K_P(x - x_{des}) + K_V(\dot{x} - \dot{x}_{des}) \tag{4.4}$$

where F and x are the force on and the position of the tip, respectively, x_{des} and $\dot{x}_{des}$ are the desired position and velocity of the tip, and K_P and K_V are gains that can be adjusted to change the impedance of the arm. Using the relation

$$\tau = J^T F \tag{4.5}$$

where J is the Jacobian of the mini-manipulator, this relation becomes

$$T = J^T K_P(x - x_{des}) + J^T K_V(\dot{x} - \dot{x}_{des}) \tag{4.6}$$

where T is the torque of the mini-manipulator motors.

The non-dynamic impedance controller just described has the desirable property of regulating the interaction between the error in tip position and the force applied to the environment, at least in a static sense. It can be said to be a method of regulating the force applied to the environment. Unfortunately, the input to this controller is a desired position, not a desired force. It is possible to determine a desired position from a desired force, but only if the exact position of the edge of the target and the combined compliance of the target and end-effector are known. In practice, neither of these will be known exactly. Moreover, the properties of the target must be known, so all objects would have to be categorized in advance, thus limiting the generality of the algorithm.

To overcome these limitations, feedback can be used to set the desired position instead of setting it a priori. For example, it can be modified using the difference between the desired and actual force. This is the basis of the third control algorithm that has been experimentally tested. The same impedance controller operates at all times, whether in or out of contact with the environment. When contact has been made, however, a control loop is used to regulate the desired position. In this method, the force error, integral force error, and error in velocity normal to the perceived surface of the target are combined to determine an incremental change in the desired position. The desired position thus determined is then used by the impedance controller.

The distance within the surface to place the desired position is determined by the equation

$$Step = K_F(F - F_{des}) + K_{V_{Normal}}\dot{x}_{Normal} + K_I \int (F - F_{des})dt \tag{4.7}$$

where K_F, $K_{V_{Normal}}$, and K_I are constant gains, F and F_{des} are the actual and desired force applied by the arm respectively, $\dot{x}_{Normal}$ is the velocity normal to the surface of the target, and $Step$ is the distance inside the surface to command the desired position. Using this value, the desired positions for the impedance controller can be computed using the equations

$$x_{des} = x_{tip} - Step\cos\theta_F \tag{4.8}$$

$$y_{des} = y_{tip} - Step\sin\theta_F \tag{4.9}$$

where x_{des} and y_{des} are the Cartesian coordinates of the desired position of the tip, x_{tip} and y_{tip} are the actual position of the tip, and θ_F is the angle at which force is applied by the target.

With this scheme, the addition of sliding along the surface is not very difficult. The desired velocity along the surface can be set to the desired sweep speed. The desired position is set to the desired position calculated by the force loop in Eqns. 4.8 and 4.9 plus a step along the surface equal to the sweep speed multiplied by the sample period. In this way, the calculation of the desired position along and into the surface are independent of one another.

The addition of sliding along the surface does not change Eqn. 4.7. Eqns. 4.8 and 4.9 become

$$x_{des} = x_{tip} - Step\cos\theta_F - V_{Sweep}T_S\sin\theta_F \tag{4.10}$$

$$y_{des} = y_{tip} - Step\sin\theta_F + V_{Sweep}T_S\cos\theta_F \tag{4.11}$$

with the additional equations

$$\dot{x}_{des} = -V_{Sweep}\sin\theta_F \tag{4.12}$$

$$\dot{y}_{des} = V_{Sweep}\cos\theta_F \tag{4.13}$$

where V_{Sweep} is the desired speed at which to sweep along the surface, T_S is the sample period, and $\dot{x}_{des}$ and $\dot{y}_{des}$ are the Cartesian coordinates of the desired velocity for the impedance controller.

4.4 Experimental Results

This section will examine and discuss the experimental results obtained to date. All results are for the mini-manipulator on a fixed base. All three control methods discussed in Sect. 4.3.2.2 are demonstrated and discussed.

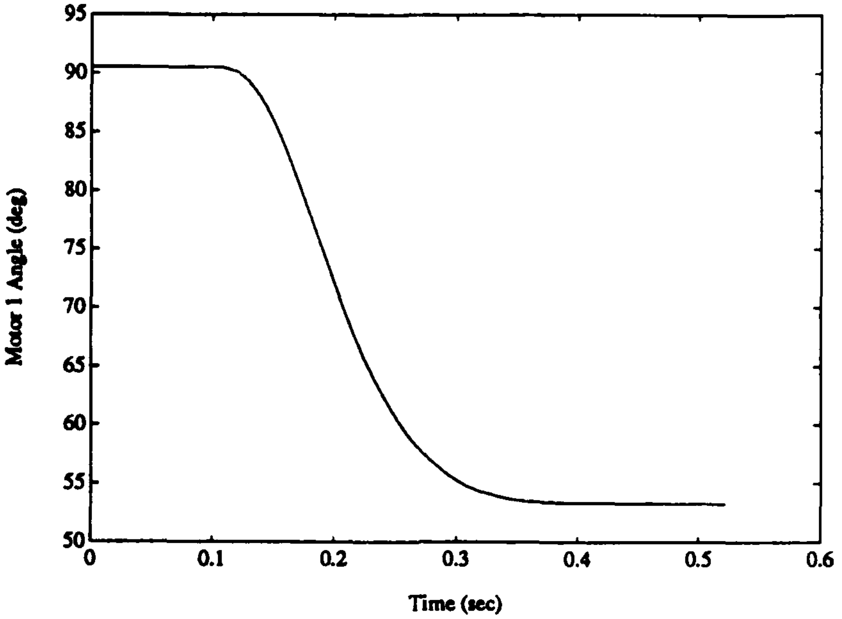

Figure 4.7: **Step Command in Angle for PD Angle Controller**

4.4.1 Position Control

The PD angle controller was given a step command such that the mini-manipulator tip moved parallel to the X axis from a Y coordinate of 10 cm to a Y coordinate of 0 cm The angle of motor 1 versus time is shown in Fig. 4.7. The motor 1 angle follows a well-damped trajectory to the new desired angle. This is expected because the motor angles are being controlled.

The motor angles of the mini-manipulator are not the important quantities, however. It is the position of the tip of the mini-manipulator that must be controlled. The response of the tip of the mini-manipulator in Cartesian space is shown in Fig. 4.8. The trajectory followed by the tip has a large curve in it and is not the straight path that is desired. This shows graphically the result of controlling the motor angles and not the tip position.

The impedance controller, on the other hand, is controlling position of the mini-manipulator tip directly. The response of the mini-manipulator tip to the same step command described for the PD controller is shown in Fig. 4.9. The tip position follows a nicely damped trajectory, which is the desired behavior.

A comparison of the manipulator tip trajectories for the two controllers is shown in Fig. 4.10[1]. This shows clearly the superior behavior of the impedance controller in following a straight-line trajectory at the mini-manipulator tip. Other than the deviation at the end of the trajectory, which might be due to the friction in the system,

---

[1] These trajectories do not start and end at exactly the same place because these points were specified to one in terms of angles and the other in terms of cartesian tip coordinates. In addition, friction causes positioning errors.

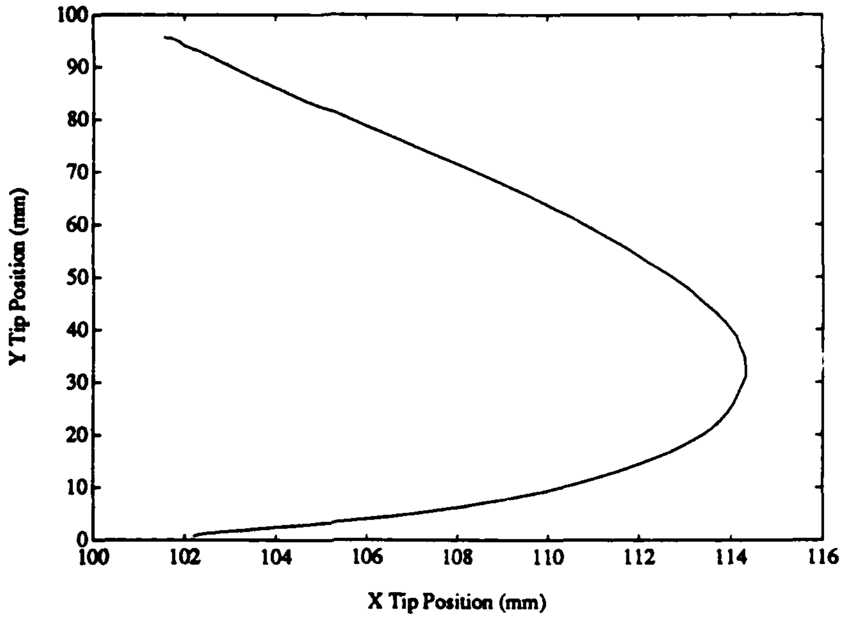

Figure 4.8: Cartesian Trajectory for PD Angle Controller

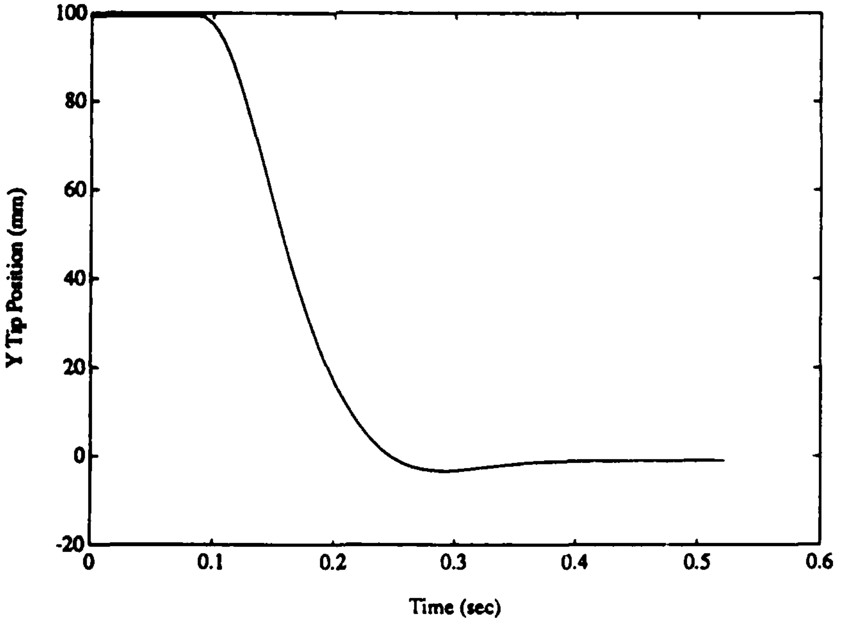

Figure 4.9: Step Command in Y Coordinate for Impedance Controller

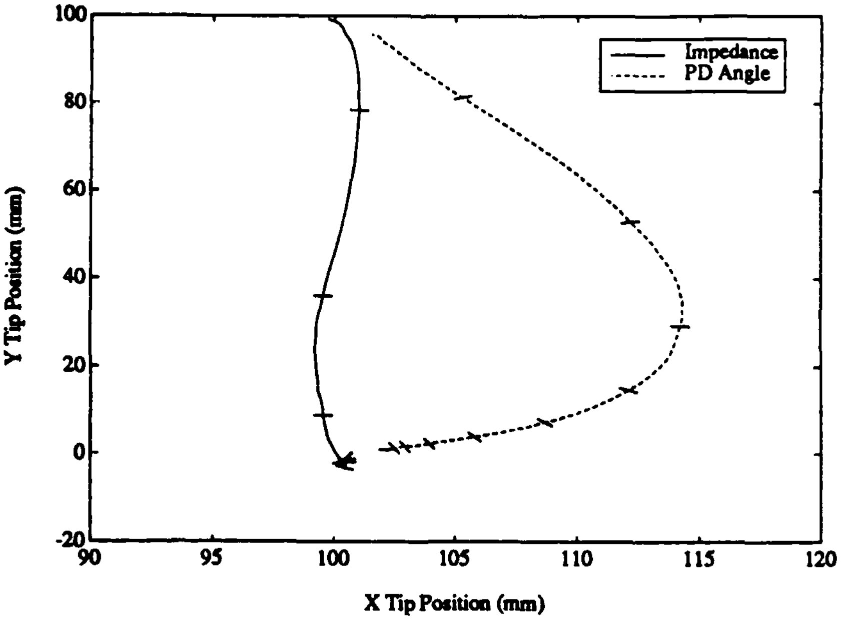

Figure 4.10: **Trajectory Comparison for Two Position Controllers**

the impedance controller takes the mini-manipulator tip very close to a straight line. The time tics on the plot are spaced evenly at approximately 40 msec intervals along the trajectory. Note that not only is path closer to a straight line, the impedance controller takes about half the time to arrive in the vicinity of the desired point.

However, the impedance controller implemented is strictly non-dynamic control. It doesn't take into account any of the acceleration terms in the equations of motion. Its ability to follow a straight-line trajectory should therefore get worse over larger slews when the manipulator is moving faster and these terms are larger. These effects are demonstrated in Fig. 4.11. For a step command in position approximately twice the size of the previous one, the trajectory deviates from a straight line more than the previous one. This is partially due to the larger velocities and accelerations, especially in the middle of the trajectory, and partially due to the saturation of the motors at the beginning of the trajectory. It does not, however, deviate as much from a straight line as the trajectory for the PD angle controller.

If the performance of this controller is not good enough to meet performance specifications, there are ways to increase performance. One possibility is to use trajectories rather than simple step commands in position. In that way, acceleration terms could be kept within set bounds, so they will have less effect on the trajectory. Also, motor saturation can be avoided.

If this still does not result in the desired performance gain, an impedance controller could be implemented using the full equations of motion of the manipulator. This controller algorithm was described by Hogan in [4]. This would theoretically result in a straight-line trajectory, with only the effects of unmodelled dynamics, errors in modelling, and the digital rather than analog implementation of the controller causing

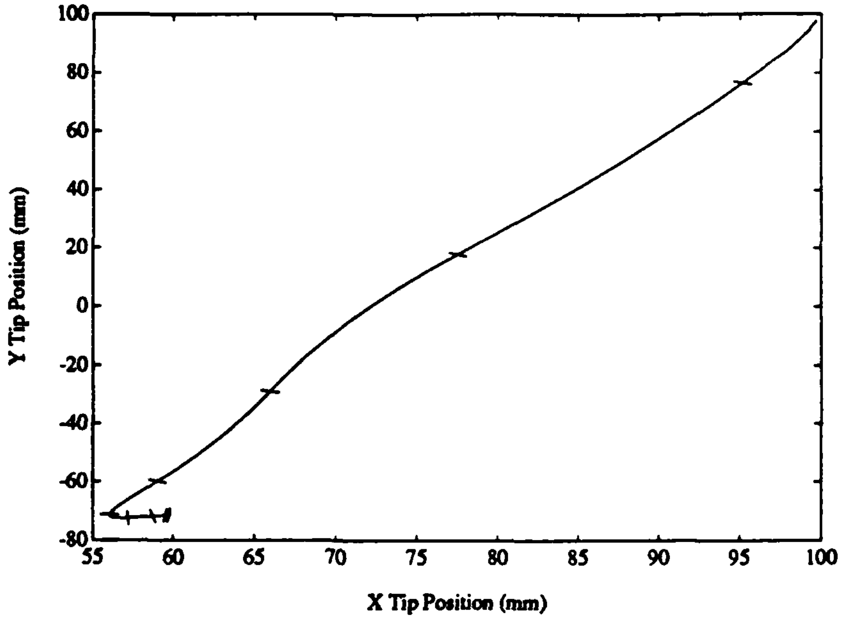

Figure 4.11: **Trajectory for Impedance Controller for a Larger Step**

deviations. Because of the complexity of the equations of motion of this system, however, this would be a computationally intensive process to be performed inside a control loop. The computational complexity might cause a reduction of sample rate to the point of offsetting any gains from using the full equations of motion.

4.4.2 Force Control

The force control experiments were performed with the mini-manipulator tip already in contact with a massive target. The ability to make a smooth touchdown is not tested in any of these experiments.

The first test of the force controller was to give a step command increase to the desired force from $0.2N$ to $1.0N$. The force applied to the target as a function of time is shown in Fig. 4.12. In this plot and all plots in this section, the desired force is shown as a dashed line. The response is well damped with a very small overshoot and a rise time of less than $0.14sec$. This is a satisfactory response to a step command.

The next experiment was the reverse of the previous. While the force was stabilized at $1.0N$, the commanded force was set to $0.2N$. The response should be the inverse of the previous one. This experiment was designed to test whether a step from a high to a low force level would cause the mini-manipulator tip to lose contact with the target. The response, which is shown in Fig. 4.13, is similar to the previous one. More importantly, the manipulator did not lose contact with the target. Again, there is almost no overshoot to the step command in force.

Another experiment performed was to test the response to disturbances in force. The mini-manipulator was held against the surface with a larger force than would have

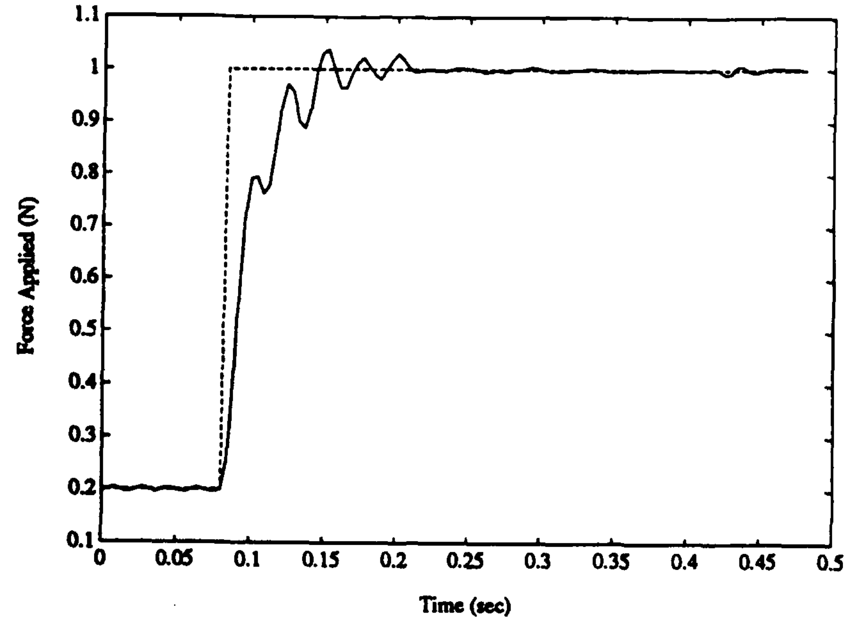

Figure 4.12: Force Applied for a Step Command Increase in Force

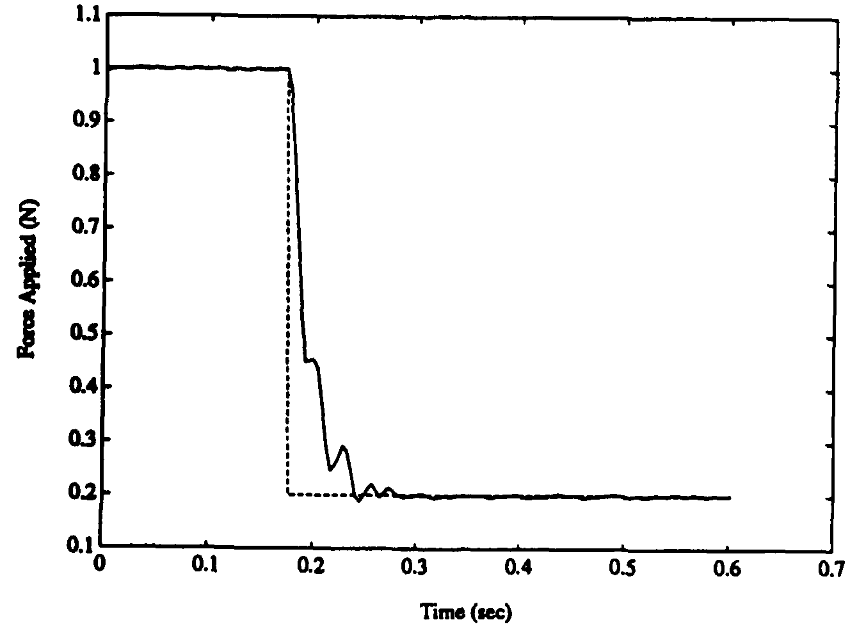

Figure 4.13: Force Applied for a Step Command Decrease in Force

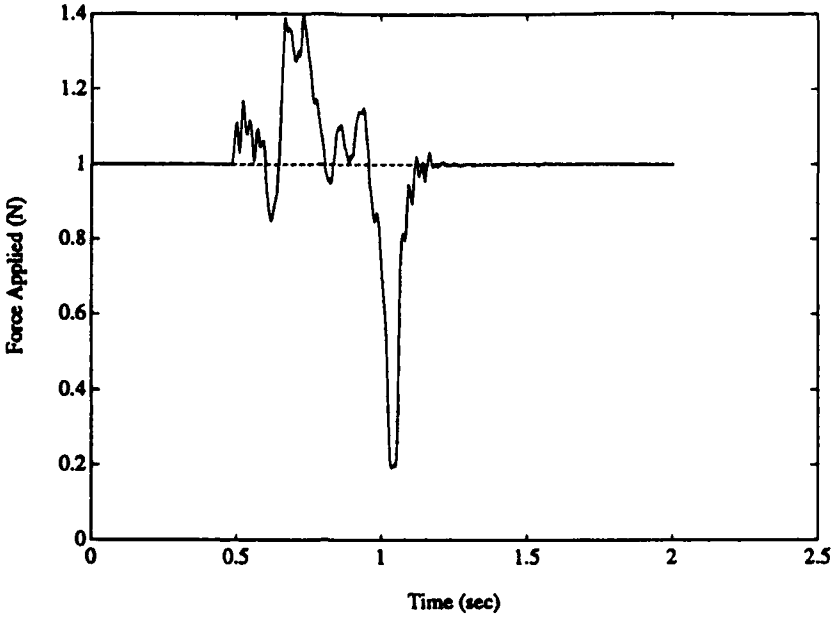

Figure 4.14: **Response to a Disturbance in Force**

been applied by the motors alone. This force was released and the response recorded. This response is shown in Fig. 4.14. It shows a large swing in the force below the desired force when the disturbance force is released. While this behavior is expected, the extent of it is somewhat distressing. Had the disturbance force been very much larger, the mini-manipulator would have lost contact with the target. The gains should be changed to prevent this occurence.

The final test of the ability of this controller to maintain a constant force at a fixed point on the target was to move the target while the mini-manipulator tip was in contact. The target was moved by hand with a speed on the order of $1cm/sec$. The results are shown in Fig. 4.15. This shows that the errors in force are all less than 10%. This bodes well for the ability of the mini-manipulator to follow a moving target and reject any position disturbances in the target.

One more part of the control algorithm should be noted. The sine and cosine of the angle at which the force is being applied is being filtered with a single pole roll-off digital filter. This was necessary only for the case when the two motor angles are nearly equal and opposite. Any significant noise in the force sensor at this point can cause the force angle to switch rapidly between positive and negative. In this part of the workspace, this will also cause the motor torques to switch rapidly between positive and negative. If this switching is too rapid, it exceeds the mechanical time constant of the motors,which will oscillate and not apply the desired torque.

In order to have the manipulator maintain a desired force on a fixed point on the target, two problems remain to be solved. First, as shown above, the response to force disturbances is currently unacceptable. This problem can probably be solved simply by changing the gains on the controller. The second problem is that the manipulator

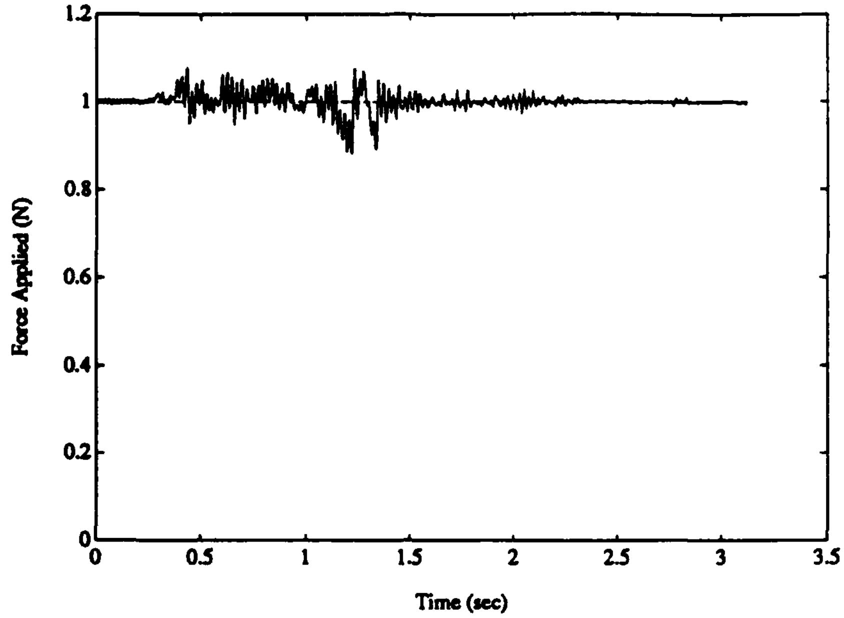

Figure 4.15: **Response with moving target**

tip, while at a few positions in the workspace and in contact with a few different target geometries, will not stay at the contact point but will drift a small distance along the surface. This problem has not been solved as yet. It is currently believed to be caused by the force sensor being slight misaligned from the vertical, causing the actual contact point to be different from the point calculated from the mini-manipulator kinematics. This problem is still being investigated.

4.4.3 Surface Following

The control algorithm to have the mini-manipulator tip follow the surface of the target while maintaining a constant force, described in Sect. 4.3.2.2, was experimentally tested and the results are described in this section. All the tests were performed using a circular target, the outline of which is shown in Fig. 4.16. In all cases, the commanded force was $1.0N$.

The first test was to command the manipulator tip to sweep along the surface at $3.0cm/sec$. The response is shown in Fig. 4.17. This shows that the force is maintained at all points on the surface to within 7%.

Next, the sweep speed was increased to $13cm/sec$. The response is shown in Fig. 4.18. This shows that the maximum force error has increased to about 13%. This increase in force error is expected since the controller is now trying to track variations in the surface at a faster rate. At even higher rates, the manipulator tip will lose contact with the target.

This experiment was attempted at several sweep speeds to study how the maximum force error changed with sweep speed. The results are shown in Fig.4.19. The error

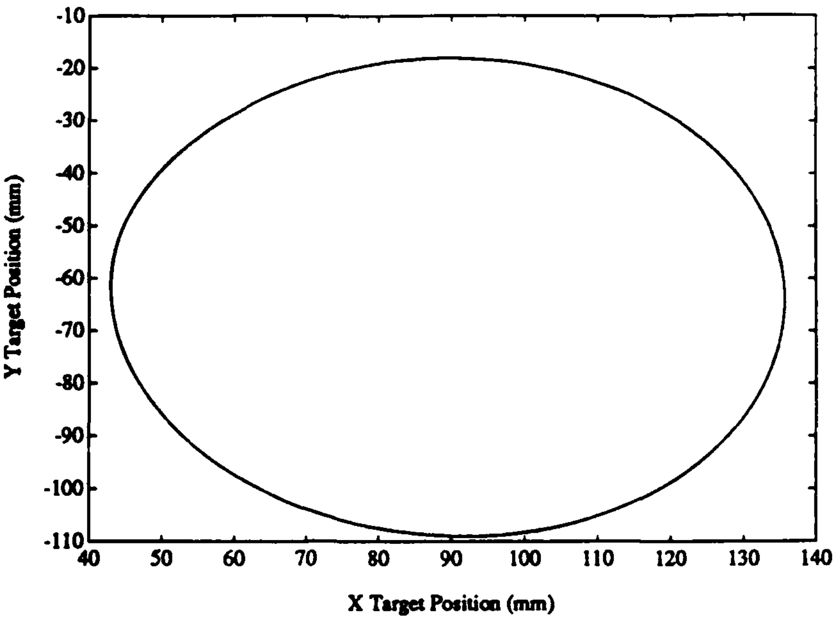

Figure 4.16: Target Surface for Surface Following Experiments

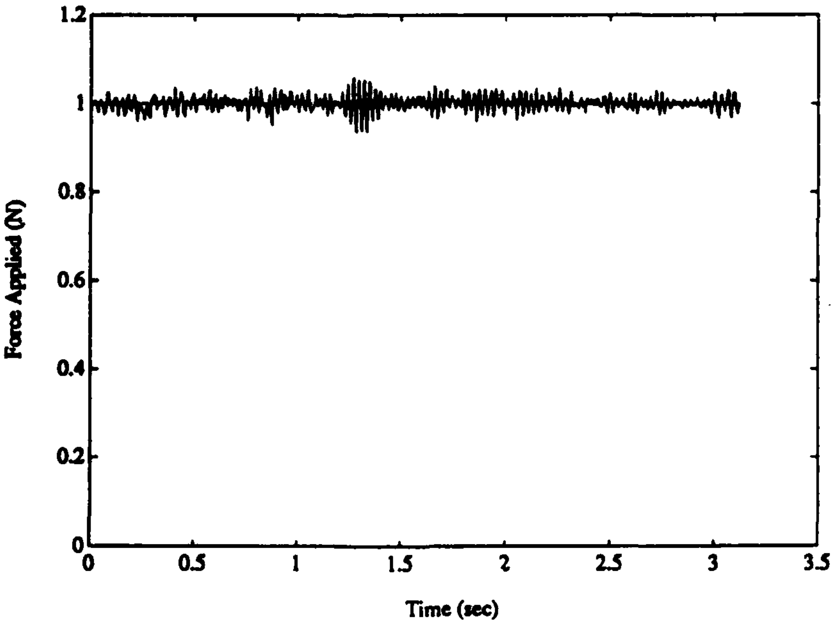

Figure 4.17: Surface Following at 3.0cm/sec

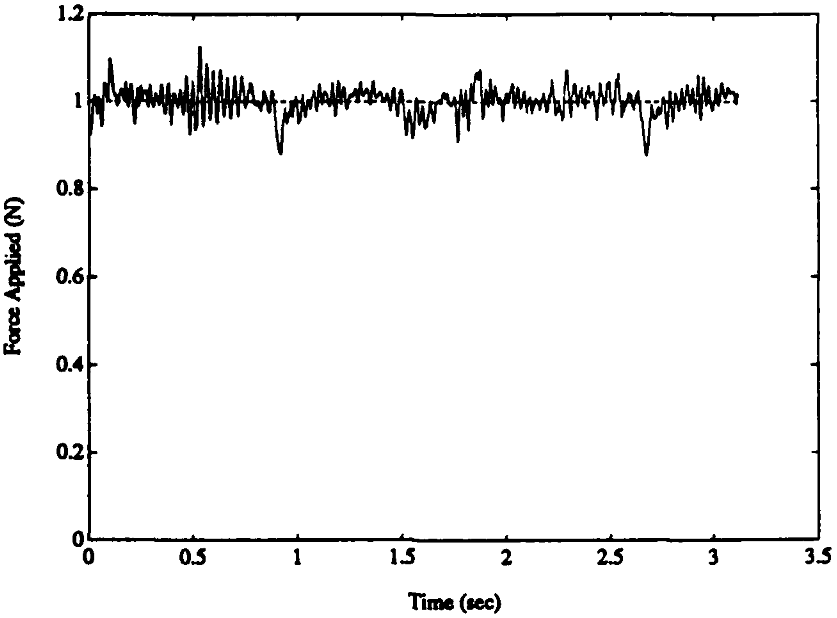

Figure 4.18: **Surface Following at** $13.0cm/sec$

does generally increase with sweep speed as expected.

There are currently two problems with the performance of this controller. The first is that below a certain commanded sweep speed, the mini-manipulator tip will stick at one point on the surface. While this is expected at very low rate due to friction between the force sensor and the target, the sweep speed at which sticking currently appears is unexpectedly large.

The second problem is that the performance appears to be unsymmetric with respect to the direction in which the target surface is traced. In fact, the maximum sweep speed at which sticking occurs is approximately twice as large in one direction as the other. Also, the sweep speed at which the mini-manipulator tip looses contact with the surface is approximately twice as large in one direction as the other. While the actual speeds at which sticking and loss of contact occur differ with different target locations, the relation between these quantities in the two directions seems to be the same.

The current explanation for these occurences is that the force sensor probe is slightly misaligned from the vertical. Thus the position of the manipulator tip calculated using kinematics is not precisely the same as the actual position of the tip of the force sensor. The difference between these will differ throughout the workspace, causing differences in behavior with differences in position and velocity.

4.4.4 Summary of Experimental Results

The response of the mini-manipulator on a fixed base in acceptable, both in position and force control. With the correction of a few problems, these algorithms should be ready for use controlling the mini-manipulator on the main arm.

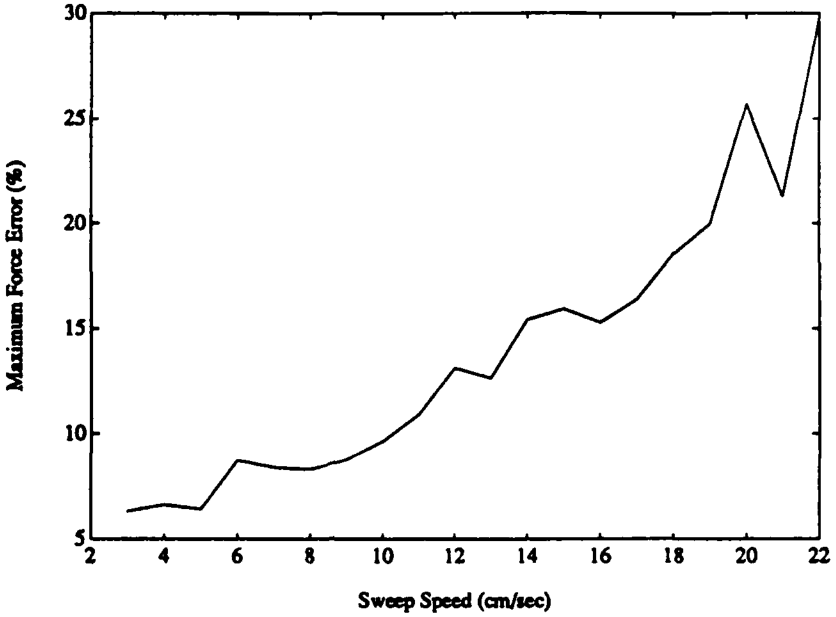

Figure 4.19: **Maximum Force Error as a Function of Sweep Speed**

4.5 Current Status

Work with the mini-manipulator on a fixed base is nearing completion. There are still a few problems yet to be resolved, however, as described in Sects. 4.4.2 and 4.4.3. Using the mini-manipulator on a fixed base has helped work out many of the problems that would otherwise have been seen when the mini-manipulator was attached to the main arm.

After completion of the work on the mini-manipulator on a fixed base, the next step experimentally will be to work on control of the main two-link arm. While the actual algorithms will be the same as those implemented by Hollars, as described in Sect. 4.3.2.1, this was implemented on a different control computer with a slightly different mechanical configuration. Therefore, there will be some changes in the actual coding of the algorithm required.

4.6 Further Research

The force controller described in this report uses force error, force integral error and velocity normal to the surface to set the desired position for the impedance controller in a straightforward way. While this has proved to be effective so far, there are other schemes that could possibly take into account the system dynamics and that would result in better performance of this controller. This is one subject which must be examined.

The control of the main two-link arm to a desired position should not be very difficult to implement, using the work of Hollars described in Sect. 4.3.2.1. However, there is a question of what position should be the desired position of the main arm. It should be possible to set the desired position of the main arm so that the mini-manipulator is in a configuration that is "optimal" in some sense. This is a topic for continuing research.

Question about how the effects of a moving mini-manipulator base will be included in the control algorithm are raised by the incorporation of the mini-manipulator on the tip of the two-link arm. It should be possible to use information about the tip of the main arm to compensate for this motion. The exact method has yet to be determined.

Currently, the speed at which the surface of the target is traced is a fixed quantity, specified by the user. It would be better, however, to set this speed based on information about the surface such as slope and curvature. For example, it would be better to go slowly when going around a sharp corner, but the speed can be much faster when tracing a straight line. This information can be obtained either from previous tracings of the same target or on estimates based on the surface just traversed. The method of determining this speed has not been determined.

The best method for going into and out of contact with the target has yet to be determined. The desired position for the impedance control should be chose to be withing the surface such that bouncing does not occur upon contact. Also, the mini-manipulator tip should return to the surface in the event of loss of contact. The best method for doing this is being looked into.

Bibliography

[1] Robert H. Cannon, Jr., Lawrence Pfeffer, Brian Andersen, and Raymond Kraft. DARPA Annual Report 1987. Annual Report 1, Stanford University Aerospace Robotics Laboratory, Stanford, CA 94305, October 1987.

[2] John J. Craig. *Introduction to Robotics Mechanics and Control.* Addison-Wesley, Reading, MA, 1986.

[3] Nevill Hogan. Impedance control: An approach to manipulation. *Trans of the ASME, Journal of Dyanmic Systems, Measurement, and Control*, 107:1–24, March 1985.

[4] Nevill Hogan. Stable execution of contact tasks using impedance control. In *Proceedings of the International Conference on Robotics and Automation*, pages 1047–1053, Raleigh, NC, April 1987. IEEE, IEEE Computer Society.

[5] Michael G. Hollars. *Experiments in End-Point Control of Manipulators with Elastic Drives.* PhD thesis, Stanford University, Department of Aeronautics and Astronautics, Stanford, CA 94035, May 1988. Also published as SUDAAR 568.

[6] Lawrence Pfeffer, Oussama Khatib, and John Hake. Joint torque sensory feedback in the control of a PUMA manipulator. In *Proceedings of the American Control Conference*, pages 818–824, Seattle, WA, June 1986.