Modelling and Control of a Haptic Interface : A Mechatronics Approach

by

Daniel R. Madill

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical Engineering

Waterloo, Ontario, Canada, 1998

©Daniel R. Madill 1998

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisitions et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-38253-2

Canada

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

With the advent of virtual reality and the increasing power of modern computers, there is renewed interest in new modalities for the human-machine interface. Force feedback is gaining momentum as new applications of teleoperation and haptic devices are discovered. This work applies the principles of mechatronics to the design and control of a novel five-bar robot for use as a haptic interface. Since the robot includes flexibility, a general model for a two degree-of-freedom flexible link mounted on an arbitrary robot is developed. The model highlights the contributions of the flexibility to the overall dynamics and presents the dynamics in a matrix form. System identification is performed to isolate the model parameters for the robot and particular attention is paid to Coulomb friction. The model is then used to develop a non-linear state estimator. The estimator is applied to the problem of implementing a virtual wall and the results are shown to be superior to conventional techniques. The importance of the mechatronics approach is highlighted throughout. Frameworks for optimal implicit force control and impedance control are also developed and experiments performed.

Acknowledgements

I wish to acknowledge the helpful insights and direction provided by my supervisor, Dr. David Wang. His patience and friendship are greatly appreciated. His willingness to be available for his students is truly exemplary.

Experimental work never fails to be fraught with unanticipated difficulties. I gratefully acknowledge the assistance of Kevin Krauel, whose support was crucial to the development of the apparatus used in this research.

Finally, I wish to acknowledge the patient support of my wife, who persevered with me through these years of effort. This work would not have been possible without her support and encouragement.

To my wife, Cheryl

Contents

1 Introduction 1

2 Background 4

2.1 Force Control 4

2.1.1 Explicit Force Control 8

2.1.2 Hybrid Position-Force Control 9

2.1.3 Impedance Control 11

2.1.4 Implicit Force Control (a.k.a. Stiffness Control) 14

2.1.5 Inverse Dynamics 14

2.2 Man-Machine Interaction 16

2.2.1 Performance Requirements 17

2.2.2 Interaction Modelling 19

2.2.3 Performance Tradeoffs 21

2.2.4 Performance Tradeoffs in Sampled Data Systems 23

2.2.5 Controller Designs 29

2.3 Haptic Interfaces 31
2.3.1 Parallel Platform Hand Controllers 32
2.3.2 Suspension Systems 33
2.3.3 Robotic Displays 34
2.3.4 Other Haptic Interfaces 36
2.4 Flexible Structures 38
2.4.1 Position-Control of Flexible Structures 39
2.4.2 Force-Control of Flexible Structures 44

3 Modelling **48**

3.1 Motivation 48
3.2 Preliminaries 49
3.2.1 Kronecker Tensor Products [1] 49
3.2.2 Matrix Calculus 52
3.3 Flexible Structures 56
3.4 Forward Kinematics 67
3.4.1 Translation 70
3.4.2 Orientation 70
3.5 Velocity Kinematics 73
3.6 Force-Torque Relationship 76
3.7 Kinetic Energy 76
3.7.1 Centre of Mass 78

3.7.2 Cross-Coupling 79
3.7.3 Moment of Inertia of the Flexible Beam 81
3.7.4 Purely Flexible Component 82
3.7.5 Inertia Matrix 82
3.8 Potential Energy 83
3.8.1 Gravitational Potential Energy 83
3.8.2 Strain Energy 84
3.9 Dynamic Equations 85
3.9.1 General Equations 85
3.9.2 Flexible Dynamics 88
3.9.3 State-Space Representation 93
3.9.4 Linearization 94
3.10 Rigid Links 96
3.11 Force Sensing 98
3.12 Single Flexible Link Revisited 99
3.13 Five-Bar Linkage Robot 103

4 Apparatus **111**

4.1 One D.O.F. Experiment 112
4.2 Direct-Drive Robot 114
4.2.1 Balancing 116
4.2.2 Decoupling 120

4.2.3 Robot Dynamics 122

4.2.4 Adjusting the Counterbalances 124

5 System Identification **126**

5.1 Power Amplifiers 126

5.2 Strain Gauge Calibration 127

5.2.1 What is Measured 127

5.2.2 Mapping to Joint Space 128

5.2.3 Calibration Experiments 130

5.2.4 Force Cancellation 132

5.3 Friction Identification 136

5.3.1 Motivation 136

5.3.2 Identification Technique 136

5.3.3 Results 140

5.4 Vibration 147

5.4.1 Natural Frequencies 147

5.4.2 Analysis 147

6 Controller Design **154**

6.1 Implicit Force Control 154

6.2 Implicit Impedance Matching 160

6.3 State Estimation 170

6.4 H_∞ Control 190
6.4.1 General Framework [2, 3, 4] 190
6.4.2 H_∞ Implicit Force Control 193
6.4.3 H_∞ Impedance Control 213

7 Conclusions and Recommendations **225**
7.1 Conclusions 225
7.2 Recommendations 228
7.2.1 Experimental Work 228
7.2.2 Theoretical Work 229

A Strain Energy of a Flexible Beam **231**

B Clamped-Free Beam Eigenfunctions **235**
B.1 Euler-Bernoulli Beam 235

C Strain Measurements **243**

D Common Matrix Operations **246**
D.1 Rotation Matrices 246
D.2 Skew-Symmetric Matrices 247
D.2.1 Properties 247

Bibliography **249**

List of Tables

3.1 Homogeneous Transformation Parameters 106

5.1 Current Amplifier Measurements 127

5.2 Optical Encoder Conversion Factors 130

5.3 Strain Gauge Calibration Constants 132

5.4 Friction Identification Results 140

5.5 Theoretical Inertias 142

5.6 Dependence of Horizontal Modes on Robot Configuration 149

List of Figures

2.1 Comparison of Physical and Virtual Environments 5

2.2 General Feedback Structure for Force Control 7

2.3 Explicit Force Control [5] 8

2.4 Hybrid Force Control 10

2.5 Impedance Control 12

2.6 Inverse Dynamics 16

2.7 One D.O.F. Hand Controller [6] 18

2.8 Stiffness Range 19

2.9 Device $\leftrightarrow$ Human Interaction [6] 20

2.10 Model of a Haptic Interface [7] 24

2.11 Parallel Platform [8] 32

2.12 SPIDAR System [9] 34

2.13 Passive, Minimum Phase Output 41

3.1 Single Flexible Link 56

3.2 Beam Element in Bending 58

3.3 Flexible Beam Mounted on a Base Robot 68

3.4 Flexible Beam Orientation at Point, **p**, Along the Beam 71

3.5 Five-Bar Linkage Robot 103

3.6 Reference Frames for the Five-Bar Robot 104

3.7 Link Lengths for the Five-Bar Robot 106

4.1 One D.O.F. Apparatus 112

4.2 Five-Bar Robot Apparatus 115

4.3 Horizontal Moment Balance 117

4.4 Vertical Moment Balance 117

4.5 Decoupling 120

5.1 Strain Gauge Positions 128

5.2 Mapping Strain to Torque 129

5.3 Strain Gauge Calibration Experiments 131

5.4 Force Cancellation 133

5.5 Open-Loop Force Cancellation Experiments 135

5.6 Time Domain Friction Modelling 137

5.7 Friction Identification Experiments 141

5.8 Underdamped Closed-Loop Step Responses 144

5.9 Positive Open-Loop Sinusoid Response 146

5.10 Negative Open-Loop Sinusoid Response 146

5.11 Horizontal Vibrations 148

6.1 General Implicit Force Control 155

6.2 Virtual Wall by Implicit Force Control: Angular Position 157

6.3 Virtual Wall by Implicit Force Control: Output Torques 157

6.4 Magnified View of Angular Position 158

6.5 Magnified View of Output Torques 158

6.6 Preload Function with Deadband 163

6.7 Friction Compensation for Free Motion 164

6.8 Preload 165

6.9 Simulation of a Spring 166

6.10 Virtual Wall Using 5 ms Sampling Period 168

6.11 Virtual Wall Using 10 ms Sampling Period 168

6.12 Block Diagram for State Estimator Simulation 176

6.13 Non-linear Estimation of Axial Joint Angle 179

6.14 Non-linear Estimation of Axial Joint Angle - Magnified View 179

6.15 Non-linear Estimation of Axial Joint Velocity 180

6.16 Filtered Velocity Measurement 181

6.17 Filtered Velocity Measurement - Magnified View 181

6.18 Friction Compensation 183

6.19 Non-linear State Estimator-Based Force Control for the Axial Joint 184

6.20 Virtual Wall by State Estimation: Angular Position 187

6.21 Virtual Wall by State Estimation: Torques 187
6.22 Virtual Wall by State Estimation: Velocities 189
6.23 General H_∞ Framework [2] 190
6.24 Implicit Force Control for Haptic Devices 194
6.25 Implicit Model Matching Framework 198
6.26 Control Effort for "Ideal" Controller C^* 203
6.27 Frequency Response for the Axial Joint 203
6.28 Bode Plot of Controller 208
6.29 Bode Plot of Loop Gain 208
6.30 Performance and Control Effort w.r.t. User Input 211
6.31 Performance and Control Effort w.r.t. Output Disturbances 211
6.32 Torque Applied By Operator 212
6.33 Axial Joint Angle as User Probes Wall 212
6.34 Magnified View of Joint Angles 214
6.35 Preliminary H_∞ Impedance Control Framework 214
6.36 H_∞ Impedance Control Framework 217
6.37 Axial Joint Angles 222
6.38 Axial Joint Torques 222
6.39 Magnified View of Joint Angle 223

A.1 Strain in Beam 231
A.2 Element Geometry 232

A.3 Beam Cross-Section 233

B.1 Bernoulli Beam Element 236

C.1 Cross-Section 243

Chapter 1

Introduction

With the advent of virtual reality and the increasing power of modern computers, there is renewed interest in new modalities for the human-machine interface. Force feedback is gaining momentum as new applications of teleoperation and haptic devices are discovered. However, haptic devices have their own unique set of design considerations that have resulted in a proliferation of novel haptic devices in the literature. Many of these devices have highly non-linear dynamics that make controller design inherently difficult. Even for simple devices, the stability of systems involving non-linear constraints has been a subject of active research, since such constraints are necessary to emulate the environments encountered in everyday life.

This work applies the principles of mechatronics to the design and control of a novel five-bar robot for use as a haptic interface. Mechatronics entails the design of mechanical systems with the goal of optimizing the balance between mechanical performance and controller complexity. The work is a continuation of the efforts of Ching and Wang [10], who developed the rigid version of the robot for implicit force control experiments and virtual reality simulations.

Chapter 2 reviews relevant literature on force control, haptic devices and control of flexible structures.

Chapter 3 presents a general model for the dynamics of a flexible beam mounted on an arbitrary robot. The beam is allowed to vibrate in two directions and any number of vibrational modes may be represented.

Chapter 4 describes the two devices used for the experimental results: a one degree-of-freedom apparatus and a three degree-of-freedom five-bar robot with a flexible link. Unlike other haptic devices, the robot structure itself functions as a force sensing device. The robot is also specifically designed to simplify the control problem, in accordance with a mechatronics approach, and its advantages are expounded in this chapter.

Chapter 5 deals with the problem of identifying the parameters of the robot model. The non-linear effects of friction are identified and modelled. The use of the robot structure to measure forces is explained and a force cancellation experiment is performed to demonstrate the reliability of the technique. These results are later incorporated into the controller design.

The original intent of this work was to apply the general model of chapter 3 to the design of controllers for the five-bar apparatus. However, the system identification revealed unanticipated vibrational modes of the robot structure that dominate the modes of the flexible link. As a result, the general flexible model developed in Chapter 3 is not applied to controller design. While changes to the robot structure, such as increasing the link flexibility or the structural rigidity may help to eliminate the unexpected modes, these options are not pursued in this work. Nor are more sophisticated modelling techniques, such as finite-element modelling, applied. However, suggestions for modifying the controller design to account for these

unanticipated vibrational modes are given in chapters 6 and 7.

Chapter 6 investigates various control strategies for implementing a virtual wall. A non-linear state estimator approach is presented and compared to the traditional implicit force control technique. H_∞ optimal control strategies are applied to the implicit force control and impedance control problems with some success.

Chapter 7 summarizes the contributions of this work and suggests areas of future research.

Chapter 2

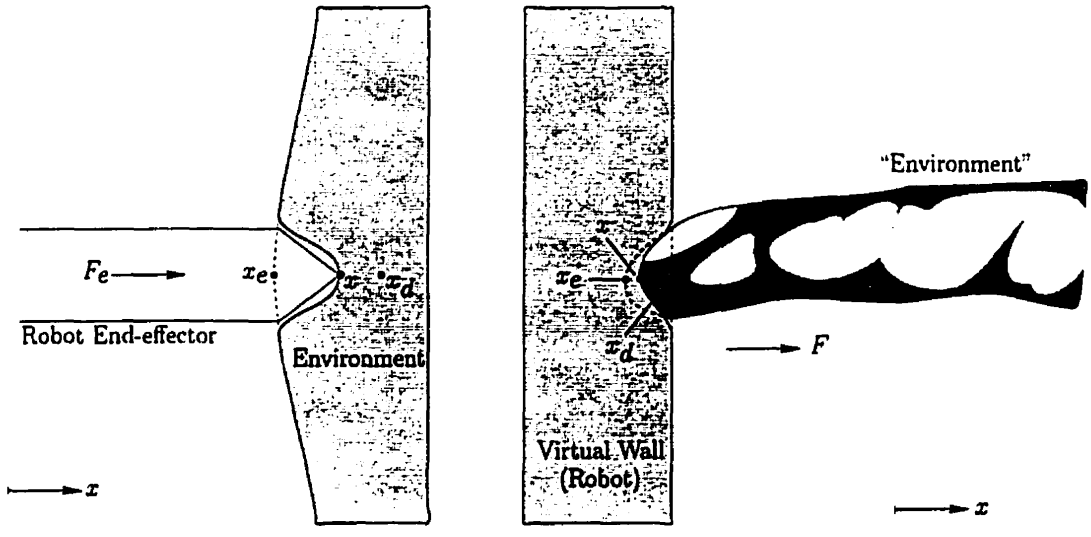
Background

2.1 Force Control

Force control attempts to govern the interaction between the robot and its environment. Consider Figure 2.1(a), which depicts a robot end-effector in contact with an environment. The point x represents the actual position of the end-effector tip, while x_e depicts the position of the environment when it is undeformed. The point x_d represents the desired position of the end-effector and F_e denotes the net force exerted on the environment by the end-effector. Suppose the environment is linear. Let $K_e(s)$ be the Laplace transform of the environment stiffness. Then the relationship between the force exerted upon the environment and the deformation of the environment is:

$$F_e(s) = K_e(s)(x(s) - x_e(s)) \tag{2.1}$$

when the end-effector is in contact with the environment. Equation (2.1) provides a convenient way of representing the relationship between the deformation of the



(a) Robot in contact with wall

(b) Finger in contact with virtual wall

Figure 2.1: Comparison of Physical and Virtual Environments

environment and the force exerted upon the environment. Since it is not possible to measure x_e, it is never used for control. Instead, the force exerted upon the environment is typically measured directly using a force/torque sensor.

Note that the end-effector position, x, is not only a function of the actuator configuration and environment stiffness, but also the stiffness of the end-effector itself, as well as joint flexibilities, link deformation, gear backlash, friction, control, sensor location, etc. These factors may prevent the end-effector from reaching the desired position, x_d.

For haptic displays, the "environment" is usually the operator's hand or fingertip, as illustrated in Figure 2.1(b). The virtual wall is provided by the end-effector of the haptic device. Hence, Figure 2.1(a) and Figure 2.1(b) are *not* mirror images

of each other, but instead depict the same scenario.

The "desired" position, x_d, could be a point on the surface of a virtual wall, as in the example of Figure 2.1(b). Note that both the fingertip and the virtual wall have "deformed" under the contact force, F. The "environment" position, x_e, roughly corresponds to the position of the fingertip were it undeformed. The undeformed shapes of the finger and virtual wall are shown as dotted lines. Let the wall be modelled as a compression spring with stiffness K_w. In this case, the goal of the controller is to make the force produced by the haptic device, F, satisfy:

$$F(s) = -K_w(s)(x(s) - x_d(s)) \tag{2.2}$$

when x lies within the confines of the wall. Similarly, the operator's fingertip deforms. Assume the user's fingertip is linear and let K_f be the fingertip stiffness. Then the deformation of the fingertip, like that of the environment in (2.2), may be modelled as:

$$F(s) = K_f(s)(x(s) - x_e(s))$$

Note that the virtual wall shown in Figure 2.1(b) does not exist physically. Instead, by controlling the contact force according to (2.2), the operator feels a wall. The actual physical configuration, however, is the same as shown in Figure 2.1(a) with the environment replaced by the operator's fingertip. Thus, the "contact force", F, and the force exerted by the end-effector on the environment, F_e, are one and the same. A simple method of determining this contact force is to measure it using a force sensor mounted on the robot end-effector.

While it may seem natural to implement force control in an analogous fashion to position control, where the error between a desired force and the measured force drives the controller, such explicit force control is only one of a variety of "force control" techniques found in the literature. However, all the force control algorithms

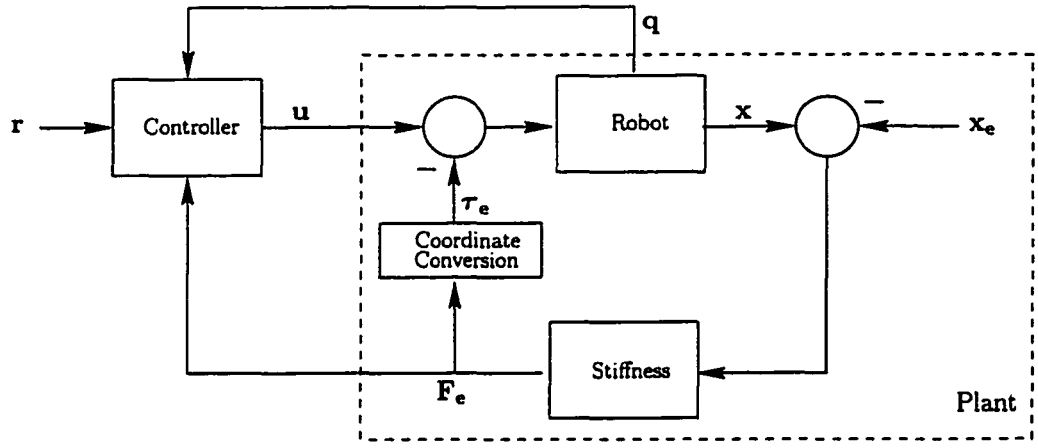


Figure 2.2: General Feedback Structure for Force Control

examined in this proposal conform to the framework depicted in Figure 2.2. The reference input, $\mathbf{r}$, is either a position, velocity, force or a combination. The controller uses measurements of joint angles, $\mathbf{q}$, and contact force, $\mathbf{F_e}$, to compute the control torque, $\mathbf{u}$. Note that the contact force, $\mathbf{F_e}$, is expressed in the end-effector coordinate frame, while the joint angles, $\mathbf{q}$, are expressed in joint space. Thus, coordinate conversions may be necessary within the controller. The control torque is expressed in joint space coordinates. The torque, τ_e, represents the contact force reflected back to the joints. The dotted frame encloses the system to be controlled — including the effects of contact with the environment.

Before discussing different types of force control, consider the relationship between the joint torques and end-effector forces. Let $\mathbf{x}$ represent the end-effector position in task space coordinates. The virtual work, δW, done by the actuator torques, $\mathbf{u}$, and contact force, $\mathbf{F_e}$, is [11]:

$$\delta W = \mathbf{u}^T \delta \mathbf{q} + \mathbf{F_e}^T \delta \mathbf{x}$$

For small displacements, $\delta\mathbf{q}$ and $\delta\mathbf{x}$ are related by the manipulator Jacobian, J:

$$\delta\mathbf{x} = J\delta\mathbf{q}$$

Hence:

$$\delta W = (\mathbf{u} + J^{\mathrm{T}}\mathbf{F_e})^{\mathrm{T}}\delta\mathbf{q} = (\mathbf{u} + \tau_e)^{\mathrm{T}}\delta\mathbf{q}$$

It is clear from this expression that the contact force is reflected back to the joints as a torque:

$$\tau_e = J^{\mathrm{T}}\mathbf{F_e} \tag{2.3}$$

2.1.1 Explicit Force Control

The most easily understood form of force control is explicit force control. In this case, the reference input is a desired force, $\mathbf{F_d}$, and the error between the measured force and desired force is used by the controller to produce the joint torques [12, 5]. Since the contact force, $\mathbf{F_e}$, and desired force, $\mathbf{F_d}$, are expressed in the end-effector frame, a coordinate conversion is necessary to express the force error as a joint torque. The necessary coordinate transformation is provided by (2.3).

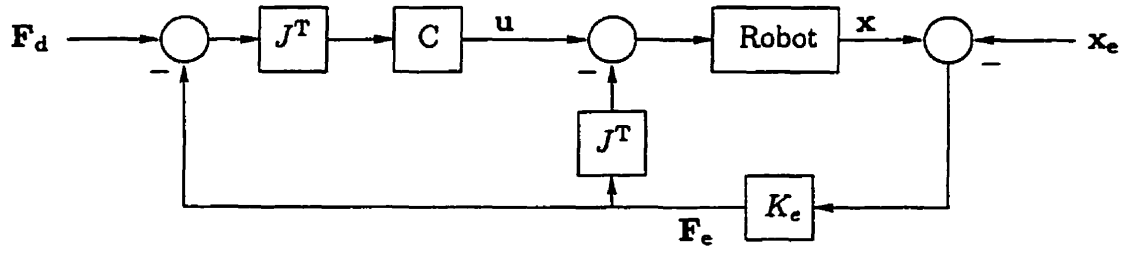


Figure 2.3: Explicit Force Control [5]

The signal-flow diagram for explicit force control is illustrated in Figure 2.3. Note that when perfect force tracking is achieved, $\mathbf{F_e} = \mathbf{F_d}$. Also, since position information is not used, the end-effector position is uncontrolled. For example, if the desired force is zero, then an impulse given to the robot end-effector will cause the robot to move to bring the force to zero. However, driving the contact force to zero does not prevent the robot from continuing to move at constant velocity, if no obstacles constrain it. Physical obstacles, such as table surfaces, are known as *natural position constraints.* Thus, explicit force control is primarily useful where natural position constraints exist. The principal difficulty encountered when implementing pure force control is noise in the force measurements [12]. Including derivative terms in the controller, C, only compounds the problem.

Borowiec and Tzes [13] apply an adaptive form of explicit force control to the control of a flexible link manipulator. A recursive least-squares estimator produces a model of the manipulator dynamics from the input torque to the force exerted on the environment. This model is then used to tune a regulator. The controller output is computed using the identified model such that a quadratic cost function of the control input and output error is minimized over a fixed number of sample periods.

2.1.2 Hybrid Position-Force Control

There are few applications which require control of force without position regulation. It is possible to combine both position and force control by controlling position and force along axes orthogonal to each other.

For example, during insertion of a peg into a hole, position control can be used to maintain the position of the peg over the hole, while force control can be applied

simultaneously in the direction collinear with the hole axis in order to maintain a prescribed insertion force.

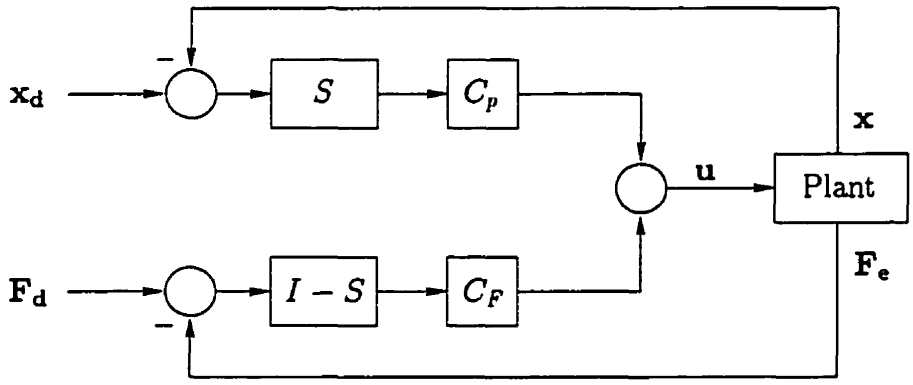


Figure 2.4: Hybrid Force Control

Figure 2.4 illustrates the feedback configuration for hybrid control. The desired position, $\mathbf{x_d}$, and desired force, $\mathbf{F_d}$, are in task frame coordinates. The selection matrix, S, is a diagonal matrix of ones and zeros. Entries containing a one correspond to the axes (in task space) which are to be position controlled. For clarity, the robot system enclosed in the dotted frame in Figure 2.2 has been collapsed into a single box labelled "Plant" in Figure 2.4. The mapping C_p represents the position controller and C_F represents the explicit force controller. Note that while the selection matrix decouples the position and force controllers, it does not eliminate coupling between force and position within the "plant".

For example, in a flexible beam of square cross-section, coupling will likely be observed between the horizontal and vertical modes of vibration. If the horizontal direction is controlled using position control while the vertical direction is regulated using force control, then the actions of the position controller will influence the force

control via coupling between the vibrational modes of the two axes (and vice-versa). Such coupling may be especially important in flexible robotic systems that employ hybrid control. No results have been observed by the author in the literature concerning simultaneous design of the position and force controllers to account for such coupling.

As an example of recent work in hybrid position/force control, see Joly et al. [14], who use mechanical analogies to suggest a simple and robust controller applicable, according to the authors, to hybrid control, teleoperation and multirobot cooperation. Experimental work validates their results.

2.1.3 Impedance Control

Consider the relationship between contact force, F, and position, X, for a single-input single-output linear system:

$$\alpha(s) = \frac{X(s)}{F(s)}$$

The transfer function $\alpha(s)$ is known as the compliance and is the inverse of stiffness. Applications involving contact with an environment generally need the robot to exhibit some compliance. Otherwise, even very small displacements can lead to extremely large forces, possibly damaging the robot or the environment. Let $V(s)$ denote velocity. Compliance may be expressed in terms of the mechanical impedance, $Z(s)$, defined to be [12]:

$$Z(s) = \frac{F(s)}{V(s)} = \frac{F(s)}{sX(s)}$$

The notion of impedance becomes clearer when considering the extremes. If the end-effector is immovable (i.e. zero velocity) regardless of the external force applied,

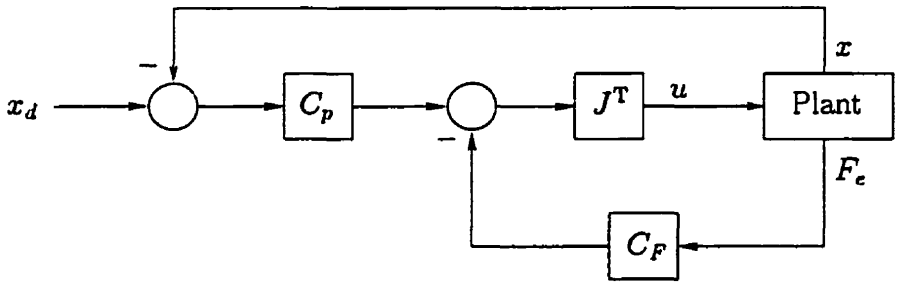


Figure 2.5: Impedance Control

the robot impedance is infinite. In contrast, if the end-effector moves at a constant non-zero velocity with no force being applied then its impedance is zero [12].

Impedance control attempts to regulate the mechanical impedance of the robot system. For example, by controlling the robot such that $sZ(s) = -(Ms^2+Bs+K)$, the system will behave like a mass-spring-damper.

Consider the control system in Figure 2.5 at equilibrium i.e., when the robot velocity and acceleration are zero. For these quantities to be zero, the net force acting on the robot must also be zero. Hence, in equilibrium, the control signal, u, must equal the torque due to contact with the environment, $\tau_e = J^T F_e$. (See Figure 2.2 for an expansion of the "Plant" block.)

For small displacements about this equilibrium position, the Jacobian is approximately constant. Let the Jacobian at equilibrium be denoted J_0. Define Z_0 to be the mechanical impedance at the equilibrium position. Thus, for small displacements about the equilibrium position and for linear time-invariant controllers, C_F and C_p, with no poles at the origin:

$$J_0^T F_e(s) = J_0^T(-C_F(s)F_e(s) - C_p(s)(x(s) - x_d(s)))$$

Assuming J_0 is full rank,

$$(I + C_F(s))F_e(s) = -C_p(s)(x(s) - x_d(s))$$

$$\therefore \quad Z_0(s) = \frac{F_e(s)}{s(x(s) - x_d(s))} = -s^{-1}(I + C_F(s))^{-1}C_p(s) \tag{2.4}$$

Observe that setting $C_p = 0$ results in zero impedance. From the block diagram in Figure 2.5, setting $C_p = 0$ yields explicit force control with zero reference i.e., the system attempts to drive the contact force to zero. Thus, if the force between the end-effector and environment was due to friction, for example, then the controller would compensate for the friction and the end-effector would move with constant velocity across the surface of the environment.

Fixing $C_F = -I$ instead results in infinite impedance (see equation (2.4)). In this case, it can be seen from Figure 2.5 and the expansion of the "Plant" block in Figure 2.3, that the force feedback exactly cancels the contact force, leaving only "unconstrained" position control i.e., assuming no limitations on the actuator torques, the system will be driven to the desired position regardless of the force due to the environment.

For treatment of the stability issues involved in the transition to and from contact using impedance control, refer to Šurdilović [15].

Modifications to the above impedance control framework do, of course, exist in the literature. For example, Jung et al. [16] switch between two slightly different impedance controllers depending on whether the manipulator is in contact with the environment, the goal being to ensure that the end-effector makes contact with the environment but remains robust to robot model uncertainty and inaccurate knowledge of environment position.

2.1.4 Implicit Force Control (a.k.a. Stiffness Control)

All the previous force control schemes used a force sensor to measure the contact force. Implicit force control regulates the impedance (or equally the stiffness) of the system without using a force sensor. Let $C_F = 0$ in equation (2.4). Then the impedance about equilibrium is simply $Z_0(s) = -s^{-1}C_p(s)$. However, from Figure 2.5, setting $C_F = 0$ is equivalent to eliminating the force feedback, and hence any need for a force sensor. The disadvantage of this scheme, as well as the sample impedance control scheme of Figure 2.5, is that only the impedance for small displacements about an equilibrium position is controlled. The transient response may be undesirable (although the system of Figure 2.5 has more control over the transient response since it incorporates force feedback).

Recently, Ching and Wang [10] demonstrated a virtual wall based on an implicit force control scheme. Ching and Wang [10] employ inertial compensation as the end-effector leaves the virtual wall in order to cancel the inertia of the robot.

2.1.5 Inverse Dynamics

Consider an n-link robot. Let:

$q(t)$ = generalized coordinates

$u(t)$ = control torques

$J(q)$ = manipulator Jacobian

$M(q)$ = inertia matrix

$h(q, \dot{q})$ = gravitational, Coriolis, centripetal and damping terms

Then the dynamic equations of the robot are [12]:

$$M(q)\ddot{q} + h(q, \dot{q}) = u - J(q)^{\mathrm{T}} F_e$$

The term $J^{\mathrm{T}}F_e$ accounts for the contact forces as the robot comes in contact with the environment. This equation may be transformed to the end-effector frame [12] by defining:

$$\overline{M}(x) = (J^{\mathrm{T}}(q))^{-1}M(q)J(q)^{-1}$$

$$\text{and} \qquad \overline{h}(x,\dot{x}) = (J^{\mathrm{T}}(q))^{-1}(h(q,\dot{q}) - M(q)J(q)^{-1}\dot{J}(q)J^{-1}(q)\dot{x})$$

Let F be the control torques in the end-effector frame i.e., $u = J^{\mathrm{T}}(q)F$. Then the dynamic equation in the end-effector frame becomes:

$$\overline{M}(x)\ddot{x} + \overline{h}(x,\dot{x}) = F - F_e$$

Let f be a new input to the system. It is easily seen from standard inverse dynamics techniques, that to obtain the double integrator system:

$$\ddot{x} = f$$

all that is necessary is to define:

$$F = F_e + \overline{M}(x)f + \overline{h}(x,\dot{x})$$

Note, however, that the expressions for $\overline{M}(x)$ and $\overline{h}(x,\dot{x})$ are highly non-linear and depend on accurate estimates of system parameters. Furthermore, difficulties arise at the singularities of the Jacobian, as well as its derivative. Also, the end-effector position, x, is seldom measured directly, but is generally computed from joint angle measurements. Thus, the calculation even of x depends on accurate estimation of the forward kinematics.

The inverse dynamics control strategy is illustrated in Figure 2.6. The area enclosed in the dotted line represents the robot dynamics in the task or end-effector space. Note that by measuring F_e and adding it to the control signal F, the contact

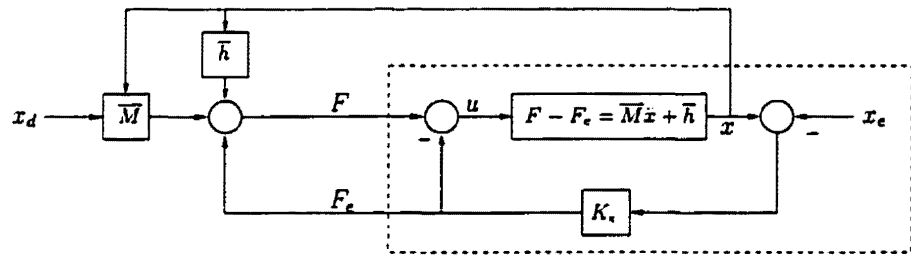


Figure 2.6: Inverse Dynamics

force is negated. This same cancellation appeared in the impedance control of section 2.1.3 for $C_f = -I$. i.e. when the impedance is designed to be infinite. Intuitively, cancelling the contact force allows the relationship between force and position to be specified arbitrarily. For example, let $\alpha(s)$ and $\beta(s)$ be transfer functions and:

$$f = -\alpha x - \beta F_e$$

Then:

$$s^2 x = -\alpha x - \beta F_e$$

and the impedance of the system is:

$$Z = -\frac{s^2 + \alpha}{\beta}$$

Thus, ideally, using inverse dynamics, the impedance of the system is controlled "exactly" i.e., even transients are controlled.

2.2 Man-Machine Interaction

The dynamic capabilities of the human operator are important for two reasons. First, knowledge of the bandwidth limitations and nonlinearities of the human

operator facilitates design decisions. Second, the human is also a feedback system, reacting with the haptic display based on proprioceptive, aural and visual sensory information. Hence, the human operator dynamics can affect the stability of the haptic display - human operator system.

The interaction between haptic display and human operator depends on a variety of factors. Certainly, the specific dynamics will vary with each individual operator. The interaction also varies with the operation modality, for example, the type of hand grip (palm grip, fingertip grip, tight or loose grip, etc.) [6]. Bergamasco [17] investigates the changes in contact area and forces as a human grasps or explores an object with the hand, revealing the complexity of the human grasping operation. Such investigations are important even for simple haptic interfaces that allow the user to release and re-grasp the haptic device during operation. The dynamics of the human operator also change according to the "degrees of freedom" used by the operator [6]. For example, wrist flexion, finger flexion and elbow flexion all exhibit different dynamic characteristics and different force profiles. Thus, it is important for analysis and design to classify the different modalities and degrees of freedom employed when operating the haptic display.

2.2.1 Performance Requirements

It would be extremely difficult to empirically determine the dynamics associated with all possible operator modalities and motions, for every known haptic device. However, by investigating the tradeoffs inherent in a particular haptic device and operator motion, a number of general "rules-of-thumb" may be developed for application to haptic interface design.

Lawrence and Chapel [6] designed a simple hand controller, depicted in Fig-

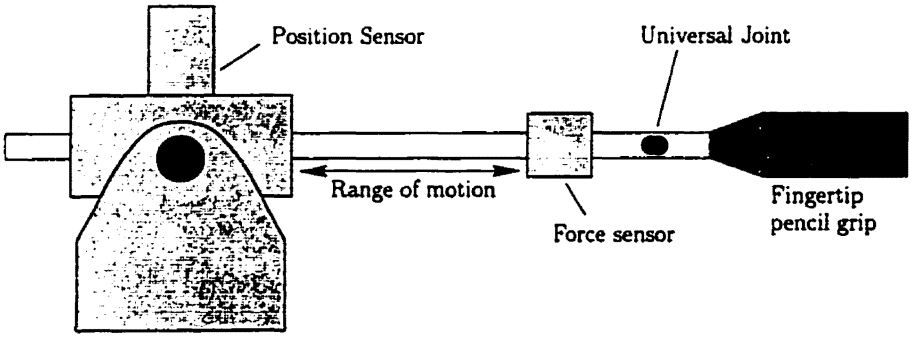


Figure 2.7: One D.O.F. Hand Controller [6]

ure 2.7, with a single degree of freedom. The haptic device was tested using a fingertip "pencil" grip, moved in an approximately sinusoidal trajectory of amplitude A_0. This amplitude can be maintained up to a frequency of ω_0 rad/sec, after which Lawrence and Chapel assumed that the motion amplitude would decrease with frequency at 40dB/decade. Thus, the input "signal" had a magnitude response of approximately [6]:

$$|X(j\omega)| = \frac{A_0\omega_0^2}{\omega^2 + \omega_0^2}$$

Now, if the force required to move the hand controller is less than some threshold, F_0, then the motion of the hand controller will feel "free" to the operator. Lawrence and Chapel assumed that this threshold remained constant for a given hand grip and motion. Thus, the maximum mechanical stiffness, $K_f(s)$, that the hand controller can exhibit before losing the feel of "free motion" is [6]:

$$|K_f(j\omega)| = \frac{F_0(\omega^2 + \omega_0^2)}{A_0\omega_0^2}$$

Similarly, Lawrence and Chapel considered fully constrained motion — the "feel" of a wall. To feel rigid, the hand controller must feel completely motionless

for all forces exerted by the operator up to some "maximum reasonable force" [6]. Let this maximum force be F_m. Assume there exists some lower bound, ∂x, on the displacement the operator can feel. i.e., for any displacement less than ∂x, the user will not detect the motion of the hand controller. Then for frequencies within the bandwidth, ω_0, the stiffness must be at least $F_m/\partial x$. At higher frequencies, Lawrence and Chapel assumed that the hand controller inertia would dominate the response, leading to a minimum stiffness, $K_c(s)$, required to achieve the sensation of a "wall" of [6]:

$$|K_c(j\omega)| = \frac{\partial x \omega^2 + F_m A_0 \omega_0^2}{\partial x A_0 \omega_0^2}$$

Thus, for a hand controller to be able to emulate the two extremes of free motion and fully constrained motion, it must be able to produce stiffnesses lying beyond [6] the shaded region shown in Figure 2.8. For $A_0 = 2$cm, Lawrence and Chapel [6] determined approximate values for ω_0, F_0 and $\frac{F_m}{\partial x}$ of 20 rad/s, 0.1 N and 104 N/m respectively.

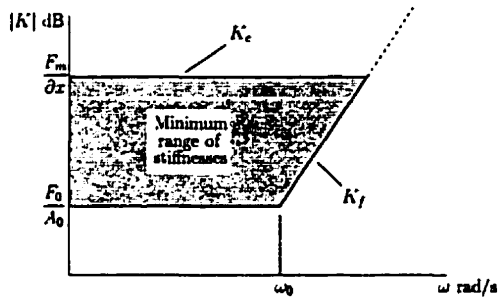


Figure 2.8: Stiffness Range

2.2.2 Interaction Modelling

To model the interaction between the operator and the haptic device, a simple mass-spring-damper model is typically used. Continuing with the example of Lawrence and Chapel's work [6], who modelled the interaction explicitly as a mass-spring-damper system, consider Figure 2.9.

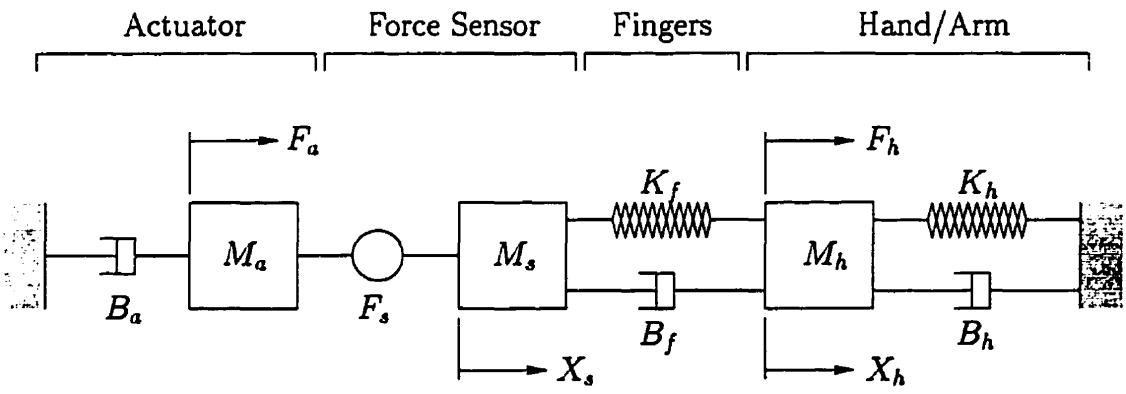


Figure 2.9: Device $\leftrightarrow$ Human Interaction [6]

M_a and B_a represent the actuator mass and damping respectively. The mass of the end-effector which moves with the force sensor is M_s. The measured force and position are F_s and X_s respectively. The spring-damper combination, K_f and B_f, represents the operator's grip on the end-effector. The hand and arm are modelled as a mass, spring and damper with parameters M_h, K_h and B_h. The position of the hand is denoted by X_h and the force exerted by the hand and arm as F_h.

The interesting features of this model are not so much the modelling of the actuator and hand/arm combination, but the inclusion of a model for the operator's grip. The strength of the grip can significantly affect the stability and performance of the system [6]. From a human force-control perspective, changing one's grip is of great utility for radically altering the compliance of the fingers/hand/arm system. In some sense, the fingers act as an active "remote-centre compliant device" while the hand functions as the end-effector and the arm as the rigid robot.

An enhanced version of this model of the human arm is provided by Lee et al. [18]. Lee conceives of the human arm as driven by two components: a force control component, driven by the error between the desired force and force sensed

by the human proprioceptive system, and a position control component, driven by the error between the desired position and the position detected by visual feedback.

Let M_h and B_h denote the mass and damping of the hand/arm system, as before. Let e_p and e_f represent the errors between the desired position and actual position, and between the desired force and actual force, respectively. Let F_e be the force exerted by the environment upon the hand/arm. Define K_p, B_p, K_f and B_f to be the spring and damping parameters for the position and force "components" of the human arm. Then the dynamics of the human-environment interaction may be expressed, according to Lee [18], as:

$$M_h\ddot{x} + B_h\dot{x} = K_p e_p + B_p\dot{e_p} + K_f e_f + B_f\dot{e_f} + F_e$$

The form of the human hand/arm model was chosen to contain similar terms to the desired impedance, since Lee et al. [18] implemented a form of impedance control, dubbed generalized impedance control (GIC), for their teleoperation system. (Note that the choice of name is rather unfortunate – only impedances of a very specific form can be achieved with the GIC control algorithm, although it does include the traditional mass-spring-damper impedance).

2.2.3 Performance Tradeoffs

A consideration of performance tradeoffs must inevitably focus on a particular control structure and specific performance objectives. However, some general insight can be gained by investigating a variety of control techniques. Consider once more the work of Lawrence and Chapel [6], since it is a simple one-dimensional system and Lawrence et al. examined the performance tradeoffs inherent in their system.

Let X_d and X_s be the desired and sensed positions respectively. Similarly, let F_d, F_s and F_a denote the desired force, measured force and actuator force respectively.

The control strategy chosen in their work was an impedance controller of the form:

$$F_a = C_f(F_d - F_s) + C_p(X_d - X_s)$$

where C_f and C_p are a force controller and position controller.

To achieve "free motion", the controller C_p should be set to zero. With F_d equal to zero, the force controller, C_f, will attempt to make the measured force equal to zero. i.e., the control system will actually assist the operator in moving the pencil by compensating for friction forces as well as inertial forces. The net result is that the controller effectively reduces the hand controller mass, damping and friction, as felt by the operator.

To achieve constrained motion, let C_p be non-zero. If C_f is zero, then the operator will experience a stiffness of C_p. Thus, it is clear that to achieve higher stiffnesses, the gain of C_p must be higher.

To consider the effects of hand grip strength, Lawrence and Chapel [6] introduced a *grip-factor*, α, which scales the hand/arm parameters M_h, B_f, B_h, K_f and K_h (see section 2.2.2). Thus, a grip factor of zero represents a loose grip and a factor of one represents a tight grip, assuming K_f and B_f represent the stiffness and damping for the tightest grip. Now, using the performance objectives already discussed in section 2.2.1, Lawrence and Chapel reached the following conclusions. First, the "tight grip regime defines the limits of force loop performance" [6] because the highest force loop gains are experienced when the grip is tightest. If the gain of C_f is too high, instability results. (Lawrence and Chapel used a Nyquist argument to evaluate stability). Intuitively, this observation makes sense. The force loop is used to provide free motion. For a tight grip, the force controller is faced with a larger mass to move (plus larger damping and stiffness). Thus, to maintain unimpeded motion, it must generate higher actuator forces. If the force controller

gain is too high the system will overcompensate and go unstable (especially since the operator will likely be attempting to provide their own compensation!).

Second, the "position loop performance limits are defined by the loose grip regime" [6]. Intuitively, the position controller is used to provide constrained motion. A tight grip helps to restrict the hand controller motion, thus improving stability. However, a loose grip barely constrains the motion of the hand controller at all, so if the position gains are too high, the hand controller will go unstable for loose grips.

Thirdly, the achievable stiffness is limited by the actuator damping and the gain of C_f. Larger actuator damping, B_a, improves the obtainable stiffness but degrades the free motion impedance. Increasing the gain C_f reduces the stiffness. A smaller actuator mass, M_a, ameliorates both the achievable stiffness and free impedance.

Finally, it is preferable for the position and force control loop bandwidths to be "one order of magnitude larger than the characteristic mechanical frequency, $\omega_m = B_a/M_a$. Significantly larger bandwidths do not improve the achievable performance" [6].

2.2.4 Performance Tradeoffs in Sampled Data Systems

While Lawrence and Chapel [6] examine performance tradeoffs with respect to the dynamics of the human-machine interaction and achieving a requisite dynamic range of stiffnesses, Colgate and Brown [7] investigate the problem of emulating a virtual wall using a haptic display coupled with a sampled data control system. The problem of implementing a virtual wall is interesting for two reasons. First, very large stiffnesses are generally required and thus, as seen in the preceding section, stability in the presence of a loose hand grip may be compromised without due

precautions. Second, a virtual wall is a "unilateral constraint" [7]. When the end-effector is away from the virtual wall, the haptic device should provide free motion. Inside the virtual wall, the motion should be constrained.

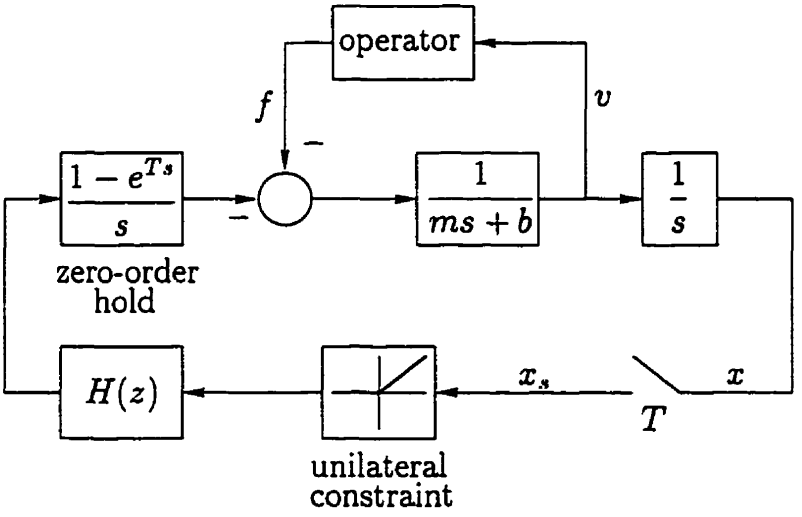


Figure 2.10: Model of a Haptic Interface [7]

Colgate et al. [7] use a one degree of freedom haptic interface similar to the device built by Lawrence and Chapel [6]. The virtual wall is implemented as a very stiff compression spring when the end-effector lies within the virtual wall. The model used by Colgate and Brown [7] to represent their haptic interface and the virtual wall is illustrated in Figure 2.10. The parameters m and b represent the inherent mass and damping respectively of the haptic device. The variables x and v are the position and velocity and u is the control effort. The force exerted by the operator on the haptic display is f. Finally, the sampling period is T and the sampled position is x_s. $H(z)$ is the discrete-time transfer function of the virtual environment. Note that the controller is essentially a discrete-time implicit force

controller.

Observe that the unilateral constraint in their model does not necessarily eliminate the output torque from the controller when the user exits the virtual wall. If $H(z)$ has memory - in other words, if it has any state - then the human operator will experience the *zero-input response* of $H(z)$ when they exit the virtual wall. The torque does not immediately drop to zero, as might be expected. This raises the question of whether these residual torques result in greater instability.

Since, as Colgate and Brown assert, the "physical world relies heavily upon the property of passivity" [7], Colgate et al. consider an impedance, $H(z)$, achievable if it can be implemented passively, according to the model of Figure 2.10. Instrumental in the assessment of passivity, is the following theorem [7]:

Theorem 2.2.1 *Passivity of a Haptic Interface [7]*
Let $w_N = \frac{\pi}{T}$ (the Nyquist frequency). The haptic interface model of Figure 2.10 is passive if and only if:

$$b > \frac{T}{2}\left(\frac{1}{1-\cos\omega T}\right)\Re e\{(1-e^{j\omega T})H(e^{j\omega T})\} \qquad for\ 0 \leq \omega \leq \omega_N$$

The proof is derived using Nyquist arguments. The interested reader may refer to [19] and [20].

Suppose the virtual wall is implemented as a spring and damper of stiffness $K > 0$ and damping coefficient B i.e. the transfer function $H(z)$ is:

$$H(z) = K + B\frac{z-1}{Tz}$$

Substituting this expression into the theorem above leads to the following condition for passivity [7]:

$$b > \frac{KT}{2} - B\cos\omega T \tag{2.5}$$

or, equivalently (consider $\omega = 0$ and $\omega = \omega_N$) [7]:

$$b > \frac{KT}{2} + |B| \tag{2.6}$$

From the above inequality, Colgate and Brown [7] concluded that:

- The haptic device must exhibit some inherent damping for the system to be passive, since the sampling period is necessarily non-zero and either the stiffness, K, or the damping, B, of the virtual wall must be non-zero for the operator to feel any effect.
- The maximum achievable virtual stiffness, K, is inversely proportional to the sampling period, T, when the damping is fixed.
- The maximum achievable virtual damping, B, is independent of the sampling period, T, when the virtual stiffness is zero.

While it is tempting to conclude that lower virtual damping, B, results in a higher achievable stiffness, K, Colgate and Brown [7] observe that the opposite conclusion can be drawn if the bandwidth of the human operator dynamics is less than half the Nyquist frequency. In this case, the $\cos \omega T$ term of equation (2.5) is always positive within the operator bandwidth. Thus, higher virtual damping allows higher virtual stiffnesses for the same physical damping, b.

Alternatively, *increasing the physical damping, b, and decreasing the sampling period, T*, allows higher virtual stiffness to be implemented [7]. While both these factors might be expected to improve the ability of the haptic display to simulate virtual walls (consider putting a real wall at the same location as the virtual wall!), one might also expect increased physical damping to interfere with the implementation of free motion. However, Colgate and Brown [7] suggest that *negative* virtual

damping may be used to compensate for physical damping. For example, let $K = 0$ and $B = -b$ in equation (2.6). Then the system is still marginally passive and exhibits no damping. This suggestion was implemented experimentally by Colgate et al. [7], confirming the conclusion that higher physical damping can improve the simulation of virtual walls. Indeed, Colgate and Brown refer to the maximization of the haptic device's inherent damping as the "least expensive and highest payoff measure available" [7] for implementing virtual walls.

Colgate and Brown [7] also investigated the effect of using the Euler difference method to calculate the velocity from encoder position measurements, as in the damping term of $H(z)$. Colgate et al. observed that the derivative estimate is highly sensitive to the sampling period and encoder resolution. If one encoder count is equivalent to Δ degrees and the sampling period is T, then the minimum velocity discernible by differentiating the encoder counts is Δ/T °/sec. Thus, at 8000 counts/rev and a sampling period of 1 ms, the minimum detectable velocity is 450 °/sec [7]. Ironically, increasing the sampling period improves the velocity resolution. Of course, the preferable solution is to improve the encoder resolution or implement digital filtering. Colgate and Brown [7] report at least an order of magnitude improvement in velocity resolution by using a simple first-order digital filter on the raw velocity estimate. Surprisingly, Colgate and Brown also show that introduction of the digital filter actually reduces the requirements for passivity. See [7] for details.

It is important to observe that in their experimental results, Colgate and Brown [7] noticed high frequency oscillations. According to Colgate and Brown, the high frequency oscillations are caused by limit cycles that arise due to the use of sampled data control with a unilateral constraint. The amplitude of the limit cycles is reduced by increasing the encoder resolution [21]. Reducing the amplitude of the

high frequency oscillations improves the subjective "feel" of the virtual wall [7]. Colgate and Brown [7] also observed that a loose one-finger grip led to instability much more readily than a full-fingered grip. This result confirms the analysis of Lawrence and Chapel [6] discussed in the preceding section.

Love and Book [22] employ Jury's discrete-time stability test to analyse the stability of virtual walls. While the results are interesting in that they demonstrate that both a lower and upper bound on the wall damping is apparently required to ensure stability, it appears that the results are only valid for the case where the user is in constant contact with the wall. Due to the unilateral constraint inherent in any virtual wall implementation, including Love and Book's, an impedance controller for virtual walls is essentially a time-varying gain-scheduling controller where the gain is switched between two values: zero and a constant gain, K. Hence, Jury's test may be used to analyse the situation where the user is in constant contact with the wall, where the gain is constant, but yields no insight into the stability of contact transitions since Jury's test is not applicable to time-varying systems.

Furthermore, Love and Book [22] only impose the unilateral constraint on the stiffness term, not the damping term. Thus, damping will be imposed outside the virtual wall, where it is generally undesirable. Outside the wall, it is preferable for the haptic device to emulate "free-space", offering the least resistance to the operator.

In contrast to the passivity-based approach of Colgate and Brown [7], which may yield conservative results, Tsai and Colgate [23] examine the same system of Figure 2.10 from an approach somewhat akin to describing function analysis. Conditions are considered under which sustained oscillations *cannot* exist for the sampled data system of Figure 2.10. It is assumed that limit cycles exist and then conditions are derived that establish a contradiction.

Theorem 2.2.2 *Limit Cycle Theorem [23]*

Consider Figure 2.10. Assume the operator applies a constant bias force to the haptic device, such that contact with the wall is achieved (although not necessarily maintained). Define:

$$O(s) = \textit{operator set corresponding to the operator in Figure 2.10}$$

$$m(\theta) = \frac{\sin\theta}{1-\cos\theta}$$

$$\hat{H}(z) = H(z)(1-z^{-1})\mathcal{Z}\left\{\frac{1}{ms^3+bs^2+s^2O(s)}\right\}$$

Then limit cycles will not occur if:

$$\max\Big\{\Re\mathrm{e}\{\hat{H}(e^{j\theta})\} + m(\theta)\Im\mathrm{m}\{\hat{H}(e^{j\theta})\},$$
$$\Re\mathrm{e}\{\hat{H}(e^{j\theta})\} - m(\theta)\Im\mathrm{m}\{\hat{H}(e^{j\theta})\}\Big\} > -1 \qquad \forall\theta\in[0,\pi]$$

2.2.5 Controller Designs

The above analyses use various assumptions about the human operator, some very general, to arrive at "stability" conditions for virtual wall implementations. However, human models are not usually directly incorporated into controller designs. Gillespie and Cutkosky [24] employ a user-specific second-order model of the human operator to develop improved controller designs. In particular, they focus on the destabilizing effects of the zero-order hold (ZOH) in the standard discrete-time implementation of virtual walls. The ZOH is destabilizing because it allows energy to be added to the system, destroying the passive nature of the simulated wall. In between sampling instants, the ZOH maintains output torque values at a constant level. However, the haptic end-effector may lose contact with the wall in between sampling instants. Hence, due to the ZOH, torque will continue to be applied even

outside the virtual wall, leading to unintended energy being added to the system. The ZOH also introduces an effective half-cycle delay.

Gillespie and Cutkosky [24] compensate for the delay introduced by the ZOH through design in the discrete domain. They account for the extra energy introduced at wall exit by a combination of model prediction and deadbeat control. User-specific model prediction is used to compute when the haptic end-effector will leave the surface of the virtual wall, and the desired state of the system at the first sampling instant outside the wall. Since the model employed is only second-order, deadbeat control is used in the last two timesteps prior to exiting the wall to drive the system to the desired state. Simulation results demonstrate the effectiveness of the technique, but while experimental results are cited as exhibiting a marked improvement over "conventional" techniques, unfortunately, no results are presented. Berkelman et al. [25] use digital state estimators on a magnetically-levitated wrist to compensate for sample-and-hold and computational delays. The results are verified experimentally. However, the forces or torques applied by the human operator are not included in the input to the estimators.

Ellis et al. [26] also confront the issues inherent in the ZOH. They arrive at a predictor-corrector scheme based on the impulse imparted by the virtual wall – the integral of force over time. Let:

$$T = \text{sampling period}$$

$$F_k = \text{force at time } kT$$

$$F(t) = \text{desired force at time } t$$

Over one sampling instant, kT to $(k+1)T$, the energy imparted by the wall is:

$$\int_{kT}^{(k+1)T} F(t)dt \approx TF_k$$

By considering predictor-correction integration methods, it is readily apparent that a better approximation to the above integral may be obtained if F_k is not simply the sampled value of $F(t)$, but more prudently chosen. Ellis et al. [26] chose F_k to minimize the difference between the desired continuous-time impulse and the impulse imparted by the discrete-time implementation. While experimental results demonstrated a marked improvement, small residual oscillations did remain.

2.3 Haptic Interfaces

Haptic man-machine interfaces provide feedback to the human operator through forces. Generally, in the literature, haptic refers to interfaces which provide gross force feedback capable of being sensed by the hand and arm, while the adjective tactile is applied to more sensitive force sensations, such as the perception of texture at the fingertip. There are numerous haptic interface designs found in the literature. The requirements for haptic interfaces can vary depending on the application. For example, the master hand controller in a telepresence system is essentially a haptic display. However, there can be advantages to making the master robot physically and kinematically similar to the slave robot, especially when real-time camera footage is provided as visual feedback. Since this thesis assumes virtual reality simulations will be used for visual feedback, master controllers used in telepresence applications will be discussed only in terms of their relevance as generic haptic displays.

Haptic interfaces may be classified into several categories: parallel platform hand controllers, suspension systems, robotic displays, hand displays and exoskeletons. Parallel platform hand controllers are essentially fully parallel or serial/parallel mechanisms akin to a Stewart platform [27]. Suspension systems use taut cables to

support and direct the operator's finger. Robotic displays are standard industrial robot configurations used to provide force feedback to the operator rather than to perform a task. Hand displays are haptic interfaces designed to interact with the fingers and palm of the human hand. Exoskeletons are manipulators which are mounted on the operator, such as on the arm and shoulder, and enhance the operator's capabilities.

2.3.1 Parallel Platform Hand Controllers

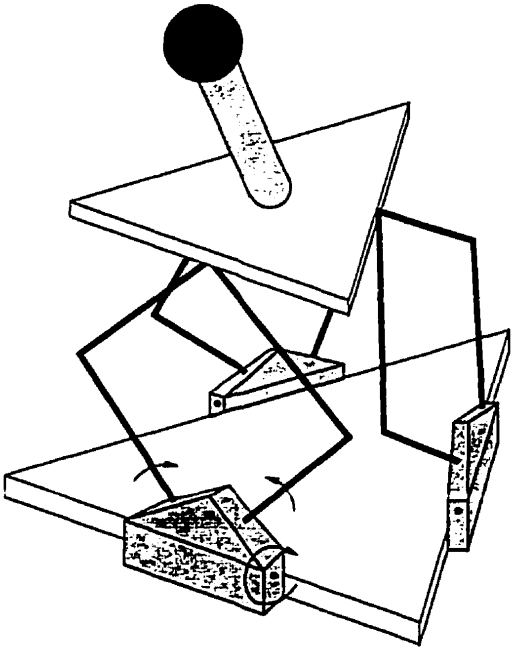


Figure 2.11: Parallel Platform [8]

Long and Collins of the University of California [8] developed the "pantograph linkage parallel platform master hand controller" illustrated in Figure 2.11. Similar mechanisms were developed by Noma and Iwata [28], Bryfogle et al. [29] and others. The design of Long and Collins will be used as an example.

Each pantograph linkage is connected to the top platform through a passive three degree-of-freedom ball-in-socket joint. The two links of each pantograph attached to the base platform are actuated by rotary motors. The point of attachment can also rotate (passively) along an axis tangential to the corresponding edge of the base platform. These degrees of freedom are indicated by arrows in Figure 2.11. The actuator mechanism includes "cable/pulley reducers and differentials" rather than

gears in order, as claimed by the authors, to "enhance transmission performance by minimizing friction, increasing efficiency, and eliminating backlash and torque ripple" [8].

The parallel platform mechanism of Bryfogle et al. [29] employs "direct-drive" actuation, although a cable mechanism is used for power transmission (but not gear reduction). The perceived advantages of parallel platform hand controllers is their compact size and high stiffness [8, 28]. Although earlier parallel platform designs suffered from a rather restricted workspace, the design of Long and Collins [8], and others' recent work, has extended the workspace [28]. Unfortunately, the kinematics of parallel platform manipulators can be quite complex. Indeed, the fully parallel force-reflecting hand controller of Bryfogle et al. [29] requires a Newton-Raphson iterative algorithm to solve the forward kinematics!

2.3.2 Suspension Systems

Hirata and Sato [9] designed a very simple haptic interface, dubbed SPIDAR (Space Interface Device for Artificial Reality) comprising four cables originating at the corners of a cube and joined in the centre. The configuration is depicted in Figure 2.12.

Each cable passes through an eyelet at the corner and then through a solenoid-controlled gap and over a rotary encoder. The tautness of the cable is maintained by a counterweight hung on the end of the cable. On-off pulsing of the solenoid provides friction and is used to control the forces exerted at the junction of the four cables, where the operator places their finger. A DC servomotor with integral encoder could easily replace the encoder/solenoid combination and provide more precise force sensations.

While the apparatus is fairly large, it has a number of advantages. First, since excessive forces are not required, nylon line could replace the cables - making the mechanism virtually transparent in the operator's field of view. The workspace of the haptic display is also quite large. However, the SPIDAR interface does not provide six degrees of freedom. It is limited to three degrees of freedom unless more cables and actuators are added. However, addition of more cables increases the likelihood that the cables will interfere with movement of the operator's hand and arm.

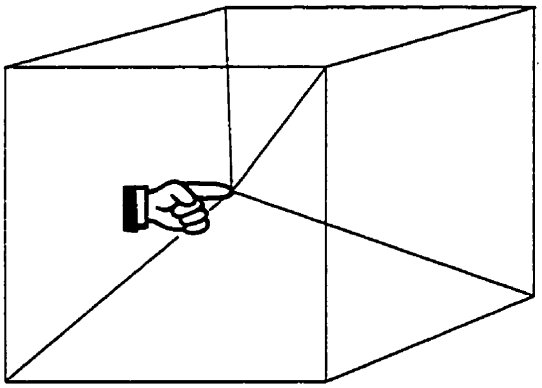


Figure 2.12: SPIDAR System [9]

A similar mechanism is employed by Morizono et al. [30] for a "virtual sports training system". Their system incorporates a head-mounted display, tension sensors on the suspension wires and a grip handle at the wire junction. The current application of the device, designed to demonstrate its high speed, is a "playing catch" simulation.

2.3.3 Robotic Displays

Most of the devices which use either commercially available or custom designed robotic manipulators as haptic displays are employed in teleoperation research, where a master device translates operator commands for a slave device and feedback from the slave is reflected back to the master. For example, Jansen, Kress and Babcock [31] use a Kraft KMC 9100-MC 6-D.O.F. manipulator, sold by Kraft Telerobotics Inc. of Overland Park, Kansas. To overcome Coulomb friction in the

manipulator joints, a preload function is used. A stiffness controller is used to control the steady-state manipulator stiffness.

Hill et al. [32] developed a 4-axis manipulator for use in endoscopic surgery that uses stiffness control to reflect forces between the master and slave manipulators. Salcudean and Yan [33] used magnetically levitated wrists in a microsurgery system. Magnetically levitated wrists were chosen because they are fast, frictionless, backdriveable, light and provide six degree-of-freedom motion. Sato et al. [34] developed a force-feedback "pencil", similar in concept to that of Lawrence and Chapel [6], for micro-teleoperation experiments.

Luecke and Chai [35] use a "force-reflecting exoskeleton" mounted on an industrial PUMA 560 robot as a haptic interface. The "exoskeleton" applies forces electromagnetically to one finger of the hand, while the PUMA 560 is used to generate gross torques and extend the workspace.

Hayward et al. [36] developed a two degree-of-freedom "Pantograph" device that the authors optimized according to size, workspace, inertia and other properties. Kelley and Salcudean [37] designed a force-feedback mouse in which standard actuators are not used. Instead, the device itself is essentially a *two degree-of-freedom actuator* with the mouse mounted on a moving plate bearing the actuation coils.

The Personal Haptic Interface Mechanism (PHANToM) of Massie and Salisbury [38] comprises a statically balanced mechanism with low mass and friction, high stiffness, good backdriveability and three degrees-of-freedom. According to the authors, the transformation matrix from the motor rotations to end-effector translations is nearly diagonal so that its three degrees-of-freedom are almost decoupled. Two of the motors actually act as counterbalances to achieve the gravity balancing. The PHANToM design also employs cables and pulleys to transmit the

motor torques to the lightweight aluminum links.

As a haptic device, Ching and Wang [10] employ a novel five-bar linkage manipulator which is both gravity balanced and *dynamically* decoupled. The robot uses direct-drive DC servomotors, without cables or pulleys for actuation. By using this innovative mechanical design, the dynamic equations are greatly simplified, making control strategies such as feedback linearization or inverse dynamics readily possible.

Hayward et al. [39] have developed a seven-degree of freedom haptic device called Freedom-7 with application to surgical training. The device is also balanced, but not dynamically decoupled. The seventh degree-of-freedom is employed to actuate the surgical tool which mounts on the end-effector, since tools such as scissors tend to have one degree-of-freedom.

2.3.4 Other Haptic Interfaces

There exists a great variety of haptic interfaces in the literature. While it is not the purpose of this chapter to enumerate all possible types of haptic devices, a brief survey of the other types of devices will be provided here for the benefit of the reader.

One fascinating class of haptic interface is the hand display. Gomez et al. [40] investigate the modelling of one such display, the "Rutgers Master II" haptic display. The Rutgers Master II display comprises a set of pneumatic actuators and linear position sensors mounted on four digits of the operator's hand. Other sensors are also employed. The device can sense hand gestures and provide forces to four digits, yet weighs only 80 grams.

Caldwell et al. [41] have developed another hand display they dub the "TactileGlove". The TactileGlove incorporate mechanoreceptive, thermoreceptive and proprioceptive feedback, all in one device. (Proprioception is the ability of the body to sense the orientation of its limbs, the movement of joints and resistance to movement, independent of surroundings. This information is provided by sensors in the muscles, tendons and joints. Mechanoreception involves the sense of pressure, touch, vibration, slip, etc. on the skin. Thermoreception entails the detection of temperature [41].) Air pressure is used to provide low-frequency mechanoreceptor feedback and piezo-electric actuators are used to induce medium and high-frequency mechanoreceptor feedback. Thermoreceptor stimulation is provided by Peltier elements, and the proprioceptor feedback by a force reflecting joystick.

Tzafestas and Coiffet [42] also have a hand display dubbed the "LRP Dextrous Hand Master". Tzafestas and Coiffet's work [42] centres on the computation of the contact forces required on each phalanx of the human hand to produce a realistic simulation of grasping a virtual object.

The impact of higher frequency feedback is less intuitive, but experiments by Wellman and Howe [43] demonstrate that high frequency feedback can impart a sense of surface stiffness and texture to a human operator. Wellman and Howe [43] refer to this higher frequency feedback as "vibrotactile feedback". Such vibrotactile feedback is used by Wellman and Howe [43] in conjunction with lower frequency force feedback to investigate its impact on human perception of surface stiffness and texture.

Submersive haptic displays are also presented in the literature. For example, the operator stands inside the "whole body kinesthetic display device" [44] of Roston and Peurach, with each foot on a six-degree-of-freedom actuated footpad. The actuated footpads allow presentation of arbitrary terrains. Also included in the

device are actuated "ground pads" to allow the operator to kneel, crawl, etc. A "vertical feature emulator" provides features such as windows, doors, etc.

Most haptic displays involve direct and continuous contact between the operator and the haptic device. Realistically, for such a device to provide a true sense of free motion when the operator is not in contact with a virtual object, the force exerted by the device on the operator must be below the operator's threshold of touch sensation. The human fingertip is extremely sensitive so giving the impression of "non-contact" while maintaining continuous contact with the device is quite difficult, if not impossible. Yoshikawa and Nagura [45] have developed a haptic interface which does *not* maintain contact with the operator. Instead, the operator puts their finger inside a ring, and the haptic control system tracks the user's finger to avoid contact except where the user impacts an object in the virtual environment.

Wannasuphoprasit et al. [46] developed a novel haptic interface by considering the design foremost from the perspective of operator safety. Hence, the "cobot" of Wannasuphoprasit et al. [46] is a mechanically passive device, in which device motion is provided entirely by the human operator. The cobot only provides constraint forces. For example, in the "unicycle cobot" of Wannasuphoprasit et al. [46], the unicycle wheel is actively steered, but not driven. While the inherently passive nature of a cobot places restrictions on its capabilities for simulating virtual environments, its novel passive design is suitable for applications such as the presentation of virtual "guides" in automotive and aircraft assembly, etc.

2.4 Flexible Structures

The five-bar linkage robot employed for the experimental research discussed in chapter 6 is a *flexible manipulator*. It is termed a "flexible" manipulator because

the last link is flexible. Joint flexibility is not considered. The flexible link is introduced into the robot design so that its deflection may be used to measure the forces being exerted by the user on the end-effector. Thus, the robot structure itself serves as the force sensor. However, link flexibility complicates the robot model since flexibility is generally represented by nonlinear partial differential equations. Furthermore, flexibility typically results in undesirable vibrations. For the haptic application considered in this work, eliminating such oscillation is particularly important. Hence, a survey of the literature in position control and force control of flexible structures is presented here.

2.4.1 Position-Control of Flexible Structures

A comparison of various position control strategies, both optimal and non-optimal, for the single flexible link is presented in [47]. The optimal control strategies employed were the rigid body linear quadratic regulator (LQR), the flexible body LQR, bounded-input H_2, one-parameter H_∞ and two-parameter H_∞.

The rigid body LQR controller is simply an LQR controller designed strictly for the rigid mode. Flexibility is ignored and the output variable is chosen to be the hub angle rather than the tip position. Since this controller ignores the link flexibility entirely, an impulse disturbance can lead to instability.

The flexible body LQR controller is an LQR controller based on the model of the flexible system. Hence, the output variable is the tip position rather than the hub angle. Obviously, by including some of the flexible modes, the order of the controller is higher. Cannon and Schmitz [48] designed a flexible body LQG controller for a single flexible link. In fact, Cannon and Schmitz are generally acknowledged as the first to study the control of a single flexible link experimentally.

The bounded-input H_2 technique attempts to minimize the mean-squared error between the actual tip position and the desired tip position, while maintaining a bounded torque command signal. Unfortunately, the technique requires the reference signal to be known and the design procedure is "very complex" [47].

The H_∞ techniques weight the uncertainty in the model in the frequency domain, with the goal of obtaining robustness to modelling errors and disturbances. In the H_∞ designs of [47], the higher order flexible modes that are not included in the plant model are treated as multiplicative uncertainty. The advantage of including the unmodelled modes as unstructured uncertainty is increased robustness to spillover effects. Spillover is the effect of the control on unmodelled higher order modes. Under certain conditions, the interaction of the controller with these higher order modes can lead to instability if the higher order modes are not included in the model for control design [49]. While the natural solution would be to increase the number of modes in the model, augmenting the model often leads to higher order controllers, which are undesirable.

Pieper [50] tackles the issue of controller order in H_2 and H_∞ designs by using order reduction techniques to reduce the order of the optimal controllers. However, he notes the impact of this order reduction on controller performance and concludes that there appear to be nominal bounds on the reduction before performance becomes unacceptable.

The non-optimal control strategies outlined by Wang in [47] are the rigid body proportional-derivative (PD) controller, the passive flexible controller and the closed-loop shaped input controller. The rigid body PD controller, like the rigid body LQR controller, treats the hub angle as the output variable and does not include any of the flexible modes in the model. The rigid body PD controller is the type of controller typically employed by industry.

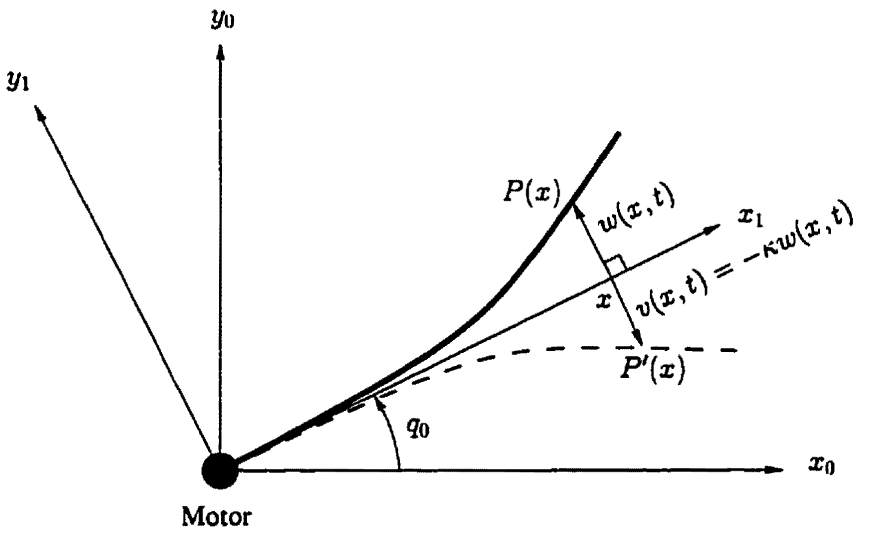


Figure 2.13: Passive, Minimum Phase Output

The passive flexible controller is based on an alternative output variable which includes the tip position but results in a minimum-phase, passive, transfer function between the input motor torque and the output. The normal tip position results in a non-minimum-phase transfer function, but by using the scaled, reflected tip position, $P'(x)$, as shown in Figure 2.13, a passive minimum-phase transfer function results, provided the link is sufficiently stiff. This result was demonstrated by Rossi, Zuo and Wang [51]. In the comparison of [47], a scaling factor of $\kappa = 1$ is used.

The closed-loop shaped input technique is a closed-loop version of the open-loop shaped input of Singer and Seering [52]. The shaped-input technique is best understood by a simple example. Let $\mu(t)$ be the unit step function. Suppose a sinusoid, $r_1(t) = A\sin(\omega t)\mu(t)$ is fed to a linear time-invariant system, P. The output will also be a sinusoid, $y_1(t) = B\sin(\omega t + a)\mu(t)$, due to the linearity of the system. Let $r_2(t)$ be the same input sinusoid delayed 180°. By adding the delayed

input to the actual input, i..e, $r(t) = r_1(t) + r_2(t)$, the output becomes:

$$y(t) = B\sin(\omega t + a)[\mu(t) - \mu(t - \tfrac{\pi}{\omega})]$$

Hence, the output goes to zero for all $t > \frac{\pi}{\omega}$. Due to superposition, the delayed input leads to complete destructive interference at the output. The same principle can be applied to cancelling vibration in a flexible beam. By adding more delays, the system can be made more robust to uncertainty. Zuo and Wang [53] further increased the controller robustness by putting the shaped-input controller into the closed-loop.

Recent work in improving the results of open-loop input shaping include the adaptive approach of Bodson [54] and the strategy of Magee, Cannon and Book, which combines input shaping and inertial damping [55].

Rattan and Feliu [56] examine discrete-time feedforward control in which the plant inverse is approximated using Wiener filter theory. The result is a more general version of the open-loop shaped-input controller of Singer and Seering [52].

Work has also been done on multiple flexible link manipulators, such as the two flexible link robot of Morris and Madani [57]. Morris and Madani [57] compare open-loop computed torque techniques to a quadratic optimal controller.

Also, Artega [58] develops a dynamic model of flexible link robot manipulators using Lagrange's equations and a serial chain approach. Artega highlights properties of the model resulting from physical attributes and the choice of flexible model.

An interesting variation on H_∞ control of a flexible link is presented by Tchernychev et al. [59], who investigates the application of *constrained* H_∞ control to tip position tracking. Constrained H_∞ control, according to the authors, involves optimizing the H_∞ norm under the constraints imposed by time-domain specifications.

The key point to note is that the time-domain specifications are not translated into the frequency domain, so the results are apparently much less conservative.

All of the control strategies discussed to this point have been linear time-invariant. Some non-linear or time-varying position control strategies have also been attempted. Kubica and Wang [60] designed a time-invariant fuzzy-logic controller that was highly successful at reducing vibration in the flexible link. Bayo et al. [61] investigated an open-loop "inverse dynamics" controller with limited success. Neural networks have also been applied to the control of flexible manipulators, such as the work of Talebi et al. [62]. Adaptive control strategies have also been applied. For example, Rokui and Khorasani [63] employ a multi-output recursive least-square algorithm for adaptive feedback linearization of a single flexible link.

Moallem et al. [64] use a non-linear observer to estimate the rates of change of the flexible modes. These rates of change are then used in an inverse dynamics-based control strategy to provide stable tip position tracking. (The non-linear observer is used to cancel particular nonlinear terms in the dynamics.) Moallem et al.'s work is expanded in [65].

Another non-linear approach is the "model-free approach" of Zhu and Ge [66]. Their controller is essentially a PD controller with an added non-linear integral term designed to enhance the controller's ability to suppress the link vibration. The added term allows for some generality in its definition and a Lyapunov stability result (based on energy concepts rather than specific linear models) is presented. Unfortunately, the results are currently only applied in simulation, although the PD nature of the controller augurs well for its implementation.

Also considered in the literature is the related problem of transportation of flexible payloads by a rigid manipulator. For example, Majors and Richards [67]

develop new real-time dynamic programming methods and apply them to the transportation of a flexible link, with the goal of minimizing the transportation time — including the time for link vibrations to cease. The real-time dynamic programming (RTDP) technique employed improves upon the work of Barto et al. [68]. By borrowing concepts from symmetric successive over-relaxation techniques for iteratively solving a system of linear equations, Majors and Richards manage to reduce the number of iterations required to solve the dynamic programming problem formulated for their transportation experiment by up to 50% over the methods of Barto et al [68].

2.4.2 Force-Control of Flexible Structures

Force control of flexible structures is still a nascent field, so it is only recently that a body of literature has begun to appear. Some early work was performed by Tilley et al. [69], who used a fast end-effector mechanism to improve the position and force response of a flexible manipulator. However, Tilley et al. only performed force control on the end-effector wrist; position control was employed to "slew the main arm into the wrist's operating envelope" [69]. Such a "dual" control system is limited in application because the position of the environment must be known to within the workspace of the end-effector wrist. For an environment, such as a human operator's hand, which is moving along an unknown trajectory, the dual control scheme is entirely inappropriate without some mechanism for detecting the environment location.

Matsuno et al. developed a "quasi-static hybrid position/force control" [70] algorithm for an elbow manipulator with both links flexible. The control scheme is called "quasi-static" because several technical assumptions had to be made in

order to make the problem tractable. Further assumptions were necessary to apply the theoretical results to their experimental apparatus. For details, refer to [70] and [71].

Further work by Matsuno, Umeyama and Kasai on force control of flexible structures may be found in [72]. There, they develop a robust H_∞ force control strategy for a flexible link with a symmetric rigid body fastened to the tip. Performance of the control strategy is validated by experimental results.

Yoshikawa et al. [73] also considered an elbow manipulator with both links flexible. However, like Tilley et al. [69], Yoshikawa et al. also added a fast, rigid "micro-manipulator" [73] with two degrees-of-freedom to the end of the elbow manipulator. Two control systems are developed in [73]. The first scheme employs "quasi-static" control in which only the position of the elbow manipulator is controlled, while force control is applied to the rigid micro-manipulator. In order for this first approach to remain valid, the dynamics of the "macro" elbow manipulator must be effectively uncoupled from the dynamics of the micro-manipulator. In their second scheme, simple proportional-derivative feedback control is applied to the joints of the elbow manipulator. However, the control algorithm for the micro-manipulator is designed to compensate for position and force errors in the flexible elbow manipulator. More recently, Lew [74] used the rigid micro-manipulator in a macro/micro manipulator experiment to actively damp out vibrations in the flexible macro-manipulator, while the contact force is regulated by a force-damping controller based on a combined model of the micro/macro- manipulator.

Chiou and Shahinpoor [75] applied a more conventional approach to force control of an elbow manipulator with both links flexible. Chiou et al. [75] derived the equations of motion for the elbow manipulator using a Lagrange formulation and then linearized the resulting non-linear equations about an operating point. Hybrid

force/position control was then employed, based on the linearized system. By examining the eigenvalues of the A-matrix in the state-space representation of the linearized system, for various link stiffnesses, Chiou et al. [75] observed that the flexibility of the first link in the elbow manipulator dominates the system stability. Hence, a manipulator with the first link rigid and second link flexible is "more stable" [75] than a manipulator whose first link is flexible and second link is rigid. Chiou et al. [75] also observed that the discontinuous contact of the end-effector with the environment resulted in limit cycles. Furthermore, variations in force sensor stiffness also affect system stability. Reducing the force sensor stiffness improves stability at the expense of positional accuracy [75].

Pfeiffer et al. [76] considered an n-link manipulator with all links flexible. For vibration suppression, a hybrid position/force controller was investigated, where a proportional-derivative (PD) controller was used for the position control and proportional control (P) for forces. Feedforward terms were also included, where the feedforward terms were calculated in two steps. First, the nominal actuator torques are calculated for the reference trajectory assuming the robot links are rigid. Second, the elastic deformations due to flexibility are computed, and corrections applied to the nominal actuator torques to compensate for elastic deformations. The actuator torques for the feedforward terms are calculated off-line for efficiency, while the PD and P controllers for position and force are applied on-line to compensate for modelling and feedforward errors, along with disturbances.

Kim et al. [77] also consider hybrid position/force control of multi-degree-of-freedom (multi-DOF) flexible manipulators. Kim et al. [77] note that due to the relationship between elastic deformations of a flexible link and the force exerted on the link tip, regulation of contact force also modulates link vibrations. Hence, Kim et al. [77] decouple, under certain assumptions, the dynamics of a multi-DOF

system into two subsystems: a "constrained motion subsystem" which is influenced by the contact force, and a "free motion subsystem" for which the impact of the contact force is negligible. The contact force is regulated for the constrained motion subsystem, but not vibrations, due to the observation above. Link vibration is only modulated for the free motion subsystem. The control outputs are based on velocity commands due to the velocity feedback servomotors of their experimental apparatus.

Chang and Fu [78] apply nonlinear adaptive control to a two-link planar flexible manipulator for the purpose of automated deburring. Again, hybrid position/force control is employed, with singular perturbation theory used to divide the dynamics into a slow and fast subsystem for control.

Recently, Chang and Chen [79] used the hub angle and deflection of a single flexible link to estimate contact force and so design a force controller for the single flexible link that does not require a force sensor. The controller structure entails a fast inner loop PD controller for the motor angle and a slower outer loop for the constraint force. The outer loop controller employs sliding mode control.

Compliance of flexible manipulators has long been touted as one of their advantages. Yet few studies exist which examine impacts between flexible structures and objects whose position is completely unknown. Moorehead and Wang [80] address this issue by using strain gauges readings to estimate forces exerted *at any point* along the flexible beam and hence to detect contact with an unknown object. Once contact has been ascertained, the controller switches from position control to force control in order to minimize the effect of the impact. Limited positive acceleration feedback is introduced to reduce the overshoot in the force response. The results have application to operator safety and robot protection since the techniques allow a flexible robot to detect unexpected collisions in its workspace.

Chapter 3

Modelling

3.1 Motivation

Two experimental apparatus are employed in this research – a simple linear mechanism, and a rigid five-bar linkage robot with an attached flexible link. Future research will likely focus on more degrees of freedom and other robotic configurations, so a general framework has been chosen for the model development. The model is based upon a flexible link which is free to vibrate in two directions. This link is clamped to an arbitrary robot. An assumed modes approach is taken to the modelling of the flexible link [81]. The base robot may be rigid or flexible. The mode shapes for the flexible beam are assumed to be orthonormal, and any number of modes may be used. The beam cross-section need not be square but an Euler-Bernoulli beam model [82] is assumed.

Because of the generality of the model, it may also be applied to a rigid robot picking up a flexible payload [83]. The model also attempts to separate the base robot terms and flexible link terms so that the effect of the flexibility on the robot

dynamics is readily apparent. In order to do this separation, the dynamics are expressed using Kronecker tensor products [84] and permutation matrices. The final dynamic equations are applied to the traditional single-flexible-link experiment to demonstrate their validity. The application of the model to the five-bar robot employed in this research is also investigated.

3.2 Preliminaries

Modelling of robotic manipulators typically involves operations such as rotations, translations, vector products and dot products. Yet the final dynamic equations are more conveniently expressed in matrix form. Hence, matrix representations of the above operations are defined to facilitate development of the general model in matrix form from the outset. Common operators, such as rotation matrices and skew-symmetric matrices, are defined in Appendix D, along with some useful properties. However, a matrix calculus is defined here so that the Euler-Lagrange equations may be applied in matrix form. In order to develop properties of the matrix calculus, the Kronecker tensor product is also discussed.

3.2.1 Kronecker Tensor Products [1]

A useful operator for expressing equations in matrix form is the Kronecker tensor product, denoted by $\otimes$. Let $A \in \Re^{n \times p}$ and $B \in \Re^{m \times s}$. Then the Kronecker tensor product of A and B is defined as:

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1p}B \\ \cdots & \ddots & \cdots \\ a_{n1}B & \cdots & a_{np}B \end{bmatrix} \in \Re^{nm \times sp} \tag{3.1}$$

Define a transformation matrix, T_{uv}, as:

$$T_{uv} = \begin{bmatrix} I_v \otimes \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \\ \vdots \\ I_v \otimes \underbrace{\begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}}_{u} \end{bmatrix} \in \Re^{uv \times uv} \tag{3.2}$$

The transformation is a permutation matrix. The definition provided here has a different form than the one in [1]. However, this definition is readily extended to the more general permutation matrix presented in section 3.2.2. The more general form is not discussed in [1].

The transformation of (3.2) is introduced here, as it is in [1], in order to allow the Kronecker product to be commuted. Being a permutation matrix, the matrix T_{uv} may be expressed as the product of elementary row or column operations. It simply exchanges rows or columns of the matrix on which it operates. While the transformation is introduced to commute Kronecker products, it may also be used to interleave matrices and its action is probably best understood in this capacity. For example, consider two matrices, A and B partitioned into column vectors $\mathbf{a}_i$ and $\mathbf{b}_i$ i.e.,

$$A = \begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_p \end{bmatrix} \in \Re^{n \times p}, \qquad B = \begin{bmatrix} \mathbf{b}_1 & \cdots & \mathbf{b}_p \end{bmatrix} \in \Re^{n \times p}$$

Post-multiplication by $T_{2,p}$ *interleaves* the columns as follows:

$$\begin{bmatrix} A & B \end{bmatrix} T_{2,p} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{b}_1 & \cdots & \mathbf{a}_p & \mathbf{b}_p \end{bmatrix}$$

The subscript 2 indicates that there are two matrices to be interleaved while the subscript p denotes the number of columns in A and B. Pre-multiplication may be used to interleave rows. The motivation for the transformation comes from the

column partitioning problem in matrix calculus, discussed in section 3.2.2. Refer to [1] for details concerning the T_{uv} permutation matrix and its application to Kronecker products. Refer to [85] for the more general permutation matrix and the application of permutation matrices to interleaving and matrix calculus.

Properties

The Kronecker tensor product has a number of useful properties. The properties enumerated below are particularly relevant for the ensuing derivation of the general dynamic equations. Define I_n to be the $n \times n$ identity matrix and let $A \in \Re^{n \times p}$ and $B \in \Re^{m \times s}$.

i. $A \otimes B = T_{nm}(B \otimes A)T_{sp}$ (3.3)

- If either A or B is a row vector then:

  $A \otimes B = (B \otimes A)T_{sp}$, since $T_{1v} = I_v$ and $T_{u1} = I_u$ (3.4)

- If either A or B is a column vector then:

  $A \otimes B = T_{nm}(B \otimes A)$, since $T_{1v} = I_v$ and $T_{u1} = I_u$ (3.5)

ii. $(A + B) \dot{\otimes} C = A \otimes C + B \otimes C$ (3.6)

iii. $(A \otimes B)^{\mathrm{T}} = A^{\mathrm{T}} \otimes B^{\mathrm{T}}$ (3.7)

iv. $(A \otimes B)(C \otimes D) = (AC \otimes BD)$ (3.8)

These properties are proven in [1].

3.2.2 Matrix Calculus

Since matrices are used throughout the modelling, a few fundamental matrix calculus relationships will be derived. The key concept here is the extension of the standard "vector" Jacobian to a matrix Jacobian. Let $\mathbf{x} = (x_1 \quad x_2 \quad \dots \quad x_m)^{\mathrm{T}}$ be a real-valued vector function of time, t, and let $f_{ij}(\mathbf{x}), i = 1, \dots, n, j = 1, \dots, p$, be real-valued differentiable functions. Let F denote the matrix function, $F(x) = [f_{ij}(\mathbf{x})]_{i=1,\dots,n,j=1,\dots,p}$. Then the *matrix Jacobian* is defined as the $n \times mp$ matrix function:

$$\frac{\partial F}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial F}{\partial x_1} & \cdots & \frac{\partial F}{\partial x_m} \end{bmatrix} \tag{3.9}$$

Note that if F is a column vector function then this definition reduces to the standard Jacobian[1]. Let $A \in \Re^{n \times p}$ and let $\mathbf{a}_i$ be a vector representing the ith column of A. Define:

$$\mathrm{vec}(A) = \begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_n \end{bmatrix} \in \Re^{np \times 1} \tag{3.10}$$

This *vectorizing* function, vec(A), could also be defined to arrange the matrix elements in row major order instead of column major order. The ordering is not important as long as it is applied consistently [86].

Application of the matrix Jacobian often leads to situations where an extended form of the transformation matrix T_{uv} is expedient. Define the *partitioning transformation*, $P_{s_1,\dots,s_r,v}$, as follows [85]:

$$P_{s_1,\dots,s_r,v} = \begin{bmatrix} I_v \otimes \begin{bmatrix} I_{s1} & 0 & \cdots & 0 \end{bmatrix} \\ \vdots \\ I_v \otimes \begin{bmatrix} 0 & \cdots & 0 & I_{sr} \end{bmatrix} \end{bmatrix}$$

[1]See Brogan [86] for a definition of the standard Jacobian matrix

The transformation, T_{uv}, of (3.2) interleaves the columns of matrices. The partitioning transformation, $P_{s_1,\ldots,s_r,v}$, is more general, and can interleave submatrices. For example, let $A \in \Re^{n\times vp}$ and partition A into submatrices, $A_i \in \Re^{n\times p}$ i.e.,

$$A = \begin{bmatrix} A_1 & \cdots & A_v \end{bmatrix} \in \Re^{n\times vp}$$

Similarly, let $B \in \Re^{n\times vs}$ and partition B into submatrices $B_i \in \Re^{n\times s}$:

$$B = \begin{bmatrix} B_1 & \cdots & B_v \end{bmatrix} \in \Re^{n\times vs}$$

Then $P_{s,p,v}$ interleaves the submatrices as follows:

$$\begin{bmatrix} A & B \end{bmatrix} P_{s,p,v} = \begin{bmatrix} A_1 & B_1 & \cdots & A_v & B_v \end{bmatrix}$$

Note that if A and B are partitioned into individual columns i.e., $p = 1$ and $s = 1$, then $P_{s,p,v}$ reduces to $T_{2,v}$. In general [85],

$$P_{\underbrace{1,\ldots,1}_{u},v} = T_{u,v} \tag{3.11}$$

Properties

Let $\mathbf{x} \in \Re^{m\times 1}$, $A(\mathbf{x}) \in \Re^{n\times p}$, $B(\mathbf{x}) \in \Re^{r\times s}$ and $F(\mathbf{x}) \in \Re^{n\times p}$. Let $(\cdot)'$ denote differentiation with respect to the vector $\mathbf{x}$ so that F' is defined as in equation (3.9) and let $\dot{()}$ represent differentiation with respect to time. Use I_m to denote the $m \times m$ identity matrix. Then[2]:

i. $$(A \otimes B)' = A' \otimes B + (A \otimes B')\underbrace{P_{s,\ldots,s,p}}_{m} \tag{3.12}$$

- If $\mathbf{x}$ is a scalar ($m = 1$) then

  $$(A \otimes B)' = A' \otimes B + A \otimes B'$$

[2]Properties are proven in [85]

- If A is not a function of $\mathbf{x}$ then

  $$(A \otimes B)' = (A \otimes B')P_{\underbrace{s,\ldots,s}_{m},p}$$

- If B is not a function of $\mathbf{x}$ then

  $$(A \otimes B)' = A' \otimes B$$

ii. $$(AB)' = A'(I_m \otimes B) + AB' \tag{3.13}$$

- If $\mathbf{x}$ is a scalar ($m = 1$) then

  $$(AB)' = A'B + AB' \tag{3.14}$$

- If A is not a function of $\mathbf{x}$ then

  $$(AB)' = AB' \tag{3.15}$$

- If B is not a function of $\mathbf{x}$ then

  $$(AB)' = A'(I_m \otimes B) \tag{3.16}$$

iii. $$\dot{F} = F'(\dot{\mathbf{x}} \otimes I_p) \tag{3.17}$$

- If F is a column vector ($p = 1$) then

  $$\dot{F} = F'\dot{\mathbf{x}}$$

iv. $$\text{vec}(\mathbf{x}) = \mathbf{x}, \quad \text{vec}(\mathbf{x}^{\mathrm{T}}) = \mathbf{x} \tag{3.18}$$

v. $$\mathbf{x}' = I_m, \quad (\mathbf{x}^{\mathrm{T}})' = \text{vec}(I_m)^{\mathrm{T}} \tag{3.19}$$

vi. If A and B are compatible for multiplication ($p = r$) then

- $\text{vec}(AB) = (B^{\mathrm{T}} \otimes A)\,\text{vec}(I_p)$
- $$\text{vec}(A)^{\mathrm{T}} = \text{vec}(I_p)^{\mathrm{T}}(I_p \otimes A^{\mathrm{T}}) \tag{3.20}$$

Partitioning

In the derivation of the dynamic equations, matrices are partitioned into submatrices which represent the base and flexible components and cross-coupling terms. Hence, it is convenient to define the effect of differentiation upon submatrices within a matrix, or differentiation by a vector which itself may be partitioned into base and flexible components. Let $\mathbf{x}$ be an m-vector and $F(\mathbf{x})$ be an $n \times p$ matrix.

i. *Vector Partitioning*

Let $\mathbf{x}$ be partitioned as follows: $\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}$. Then:

$$\frac{\partial F}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial F}{\partial \mathbf{x}_1} & \frac{\partial F}{\partial \mathbf{x}_2} \end{bmatrix} \tag{3.21}$$

ii. *Row Partitioning*

Let F be partitioned row-wise into two submatrices:

$$F = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

Then:

$$\frac{\partial F}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial F_1}{\partial \mathbf{x}} \\ \frac{\partial F_2}{\partial \mathbf{x}} \end{bmatrix} \tag{3.22}$$

iii. *Column Partitioning*

Let F be partitioned column-wise into r submatrices:

$$F = [F_1 \quad \cdots \quad F_r]$$

where F_i has s_i columns for $i = 1, \ldots, r$. Then:

$$\frac{\partial F}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial F_1}{\partial \mathbf{x}} & \cdots & \frac{\partial F_r}{\partial \mathbf{x}} \end{bmatrix} P_{s_1,\ldots,s_r,m} \tag{3.23}$$

Since the permutation matrix only operates upon the columns, row and column partitioning may be combined by partitioning the matrix F into rows prior to column partitioning.

3.3 Flexible Structures

The general model derived in this chapter is based upon a flexible link mounted on an arbitrary robot. However, before embarking on the derivation of the general dynamic equations, it is instructive to investigate the theory behind the modelling of flexible beams, since it is this vibration theory that is employed in the subsequent derivations. The theory is better described within the context of an application, so the traditional single flexible link experiment of Figure 3.1 will be considered [87, 47]. This single flexible link application will be revisited after the dynamics of the general model are derived in order to verify the results.

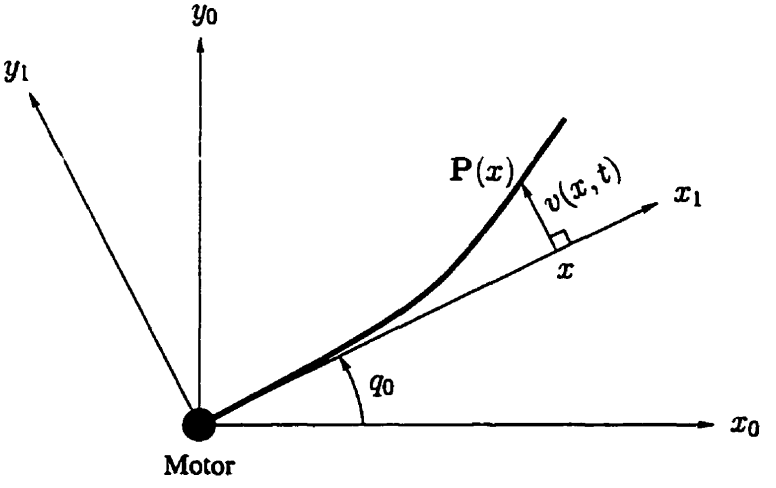


Figure 3.1: Single Flexible Link

The figure is a top view of the link. The link is typically an aluminum or steel beam of narrow cross-section. For this example, the link is mounted to a motor which is used to control the beam behaviour and the height of the beam is presumed to be much larger than the width so that only transverse vibrations are possible. Furthermore, the vibrations are assumed to occur in the horizontal plane so that gravitational forces may be ignored. In the general framework, transverse vibrations and gravitational forces are included in the dynamics, but, for simplicity, these factors are ignored in this example.

The frame x_1-y_1 is chosen such that the x_1-axis is tangent to the link at the motor (or hub). The frame x_1-y_1 is rotated from the base frame x_0-y_0 by an angle q_0. This frame is convenient because the angle q_0 is easily measured by an optical encoder or potentiometer mounted on the motor. The x_1-y_1 frame could also have been chosen to pass through the centre of gravity of the link, but Belleza et al. [87] demonstrated that either frame may be chosen. The results would only differ by a linear coordinate transformation. Although no payload mass is included in this derivation, Belleza's results [87] hold even when a payload (with both mass and inertia modelled) is added to the end of the flexible beam.

The displacement of each point along the neutral axis of the beam from the x_1-y_1 frame as a function of time is denoted by $v(x,t)$. The neutral axis is the axis of the beam that experiences zero stress [88]. For example, consider the beam element depicted in grey in Figure 3.2. As the beam is bent upward, the portion of the beam element above the neutral axis experiences compressive stress while the portion below the neutral axis undergoes tensile stress [88].

Let l be the length of the beam. Since the neutral axis does not experience any stress, the arc length of the neutral axis in the bent beam is also l. Thus, the x-coordinate of the end of the neutral axis, x_e, is less than l and the displacement

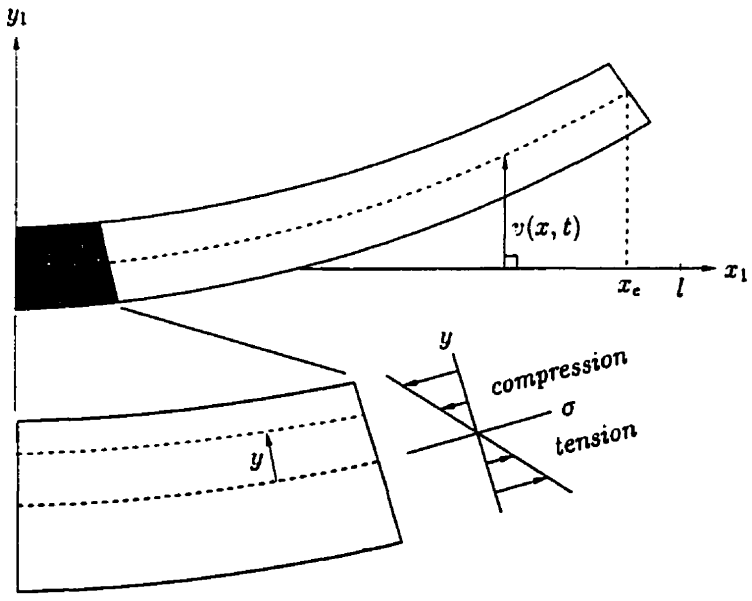


Figure 3.2: Beam Element in Bending

of the end of the beam over time is $v(x_e, t)$, not $v(l, t)$. However, for small tip deflections, $x_e \approx l$ and $v(x_e, t) \approx v(l, t)$. In the subsequent analysis of the single flexible link, borrowed from Wang [47], it will be assumed that tip deflections are "small". Furthermore, damping in the beam will be ignored. Assume a uniform beam with linear mass density, ρ.

Consider a point $\mathbf{P}(x)$ on the beam, as shown in Figure 3.1. The dependence on time of q_0 has been omitted for clarity. The coordinates of $\mathbf{P}(x, t)$ in the x_0-y_0 frame are:

$$\mathbf{P}(x,t) = \begin{bmatrix} x\cos(q_0) - v(x,t)\sin(q_0) \\ x\sin(q_0) + v(x,t)\cos(q_0) \end{bmatrix}$$

Since a Lagrangian approach will be used to develop the dynamic equations of the single flexible link, the kinetic and potential energies of the beam must be derived.

Let $\dot{()}$ denote the time derivative of the quantity in parentheses. Trivially, the kinetic energy of the flexible link, assuming $x_e \approx l$ is:

$$T_b = \frac{1}{2}\int \dot{\mathbf{P}}^{\mathrm{T}}\dot{\mathbf{P}}dm = \frac{1}{2}\rho\int_0^l \dot{\mathbf{P}}^{\mathrm{T}}\dot{\mathbf{P}}dx$$

Let I_h represent the inertia of the base motor i.e., the hub inertia. Then the kinetic energy of the hub is:

$$T_h = \frac{1}{2}I_h\dot{q}_0^2$$

yielding a total kinetic energy of:

$$T = \frac{1}{2}I_h\dot{q}_0^2 + \frac{1}{2}\rho\int_0^l \dot{\mathbf{P}}^{\mathrm{T}}\dot{\mathbf{P}}dx \tag{3.24}$$

Calculation of the potential energy of the flexible beam is more involved. It is derived in detail in Appendix A. Let $(\cdot)'$ denote differentiation with respect to x, or the *spatial* derivative. From (A.4), the potential energy of the flexible beam is:

$$V = \frac{1}{2}\int_0^l EI(v'')^2dx$$

Clearly, in order to compute the Lagrangian using the kinetic and potential energy expressions, the displacement of each point on the beam, $v(x,t)$, must be known. For control purposes, it would be preferable to have a semi-discretized representation (an ordinary differential equation), resulting in a finite-dimensional system rather than the infinite dimensional system that arises from the partial differentiation of $v(x,t)$. For this purpose, the assumed modes approach of Meirovitch [81] will be employed. In this method, it is assumed that the variables may be separated such that the beam deflection takes the form:

$$v(x,t) = \Phi(x)Q(t)$$

The function $Q(t)$ is a time-varying function that modulates the modal shape function, $\Phi(x)$. A modal shape function which approximates the exact solution is chosen. The Euler-Lagrange dynamic equations are then used to solve for $Q(t)$. Such an approach mirrors the Rayleigh Method referred to in [82]. The result should provide a good approximation to the actual beam dynamics (in an L_2 sense).

Mode shape functions for the Euler-Bernoulli beam model and clamped-free boundary conditions are derived in Appendix B. The Euler-Bernoulli beam model assumes that plane faces remain plane within a beam element and factors such as rotary inertia and shearing deformations are ignored [82]. These assumptions are standard and are appropriate for the flexible link employed in this work. While particular mode shapes are not assumed in the general model of this chapter, torsional vibration and shearing deformation are ignored, consistent with an Euler-Bernoulli beam model. However, an important observation from the derivation in Appendix B is that the separation of variables assumption typically leads to an infinite number of particular solutions, $\phi_i(x)$ and $q_i(t), i = 1 \ldots \infty$. The total solution is the sum of the particular solutions, whence:

$$v(x,t) = \sum_{i=1}^{\infty} \phi_i(x) q_i(t) \tag{3.25}$$

A convenient property of the mode shapes for the clamped-free beam that will be referred to throughout this chapter is orthogonality i.e. that for $i \neq j$:

$$\int_0^l \phi_i(x)\phi_j(x)dx = 0$$

$$\text{and} \quad \int_0^l \phi_i''(x)\phi_j''(x)dx = 0$$

From (B.14) and (B.15) in Appendix B, it is clear that these two "orthogonality" relations hold if the beam has uniform mass density, EI is constant and the

boundary conditions for the beam satisfy:

$$\phi_i'''(l)\phi_j(l) - \phi_i'''(0)\phi_j(0) - \phi_i''(l)\phi_j'(l) + \phi_i''(0)\phi_j'(0) = 0 \quad \forall i, j$$

This relationship is valid for a variety of boundary conditions, such as clamped-free, clamped-pinned and clamped-clamped. Also, as discussed in Appendix B, orthogonal mode shapes, $\phi_i(x)$, may be "normalized" such that:

$$\int_0^l \phi_i(x)^2 dx = 1 \quad \forall i$$

Kinetic Energy for the Single Flexible Link

The displacement, v, for an unforced clamped-free beam takes the form of equation (3.25), where the modal shape functions, $\phi_i(x)$, are defined by equations (B.8)-(B.9). In the case of the single flexible link shown in Figure 3.1, the hub actuator exerts a force on the link. However, a reasonably accurate model of the single flexible link behaviour can be obtained by assuming that the modal shape functions are the same as those of the unforced clamped-free beam, while the time-varying functions, $q_i(t)$, are treated as the generalized coordinates. Thus, instead of using $q_i(t) = e^{j\omega_i t}$, the Euler-Lagrange formulation is employed to determine the governing equations for each $q_i(t)$. Note that by using the eigenfunctions for the clamped-free beam, the orthogonality and normalization relationships hold. Furthermore, for practical purposes, let the displacement $v(x,t)$ be approximated using a finite number of modes, n. Thus,

$$v(x,t) \approx \sum_{i=1}^{n} \phi_i(x) q_i(t) \tag{3.26}$$

Now, consider the kinetic energy of the single flexible link, as defined by equation (3.24). The time-derivative of the position, $\mathbf{P}$, is:

$$\dot{\mathbf{P}} = \begin{bmatrix} -x\dot{q}_0 \sin(q_0) - v\dot{q}_0 \cos(q_0) + \dot{v}\sin(q_0) \\ x\dot{q}_0 \cos(q_0) - v\dot{q}_0 \sin(q_0) + \dot{v}\cos(q_0) \end{bmatrix}$$

Thus,

$$\begin{aligned} \dot{\mathbf{P}}^T\dot{\mathbf{P}} &= x^2\dot{q}_0^2 \sin^2(q_0) + v^2\dot{q}_0^2 \cos^2(q_0) + \dot{v}^2 \sin^2(q_0) + 2vx\dot{q}_0^2 \sin(q_0)\cos(q_0) \\ &\quad + 2\dot{v}x\dot{q}_0 \sin^2(q_0) + 2v\dot{v}\dot{q}_0 \cos(q_0)\sin(q_0) \\ &\quad + x^2\dot{q}_0^2 \cos^2(q_0) + v^2\dot{q}_0^2 \sin^2(q_0) + \dot{v}^2 \cos^2(q_0) - 2vx\dot{q}_0^2 \sin(q_0)\cos(q_0) \\ &\quad + 2\dot{v}x\dot{q}_0 \cos^2(q_0) - 2v\dot{v}\dot{q}_0 \cos(q_0)\sin(q_0) \\ &= x^2\dot{q}_0^2 + v^2\dot{q}_0^2 + \dot{v}^2 + 2\dot{v}x\dot{q}_0 \end{aligned}$$

Substituting the above expression into (3.24), the kinetic energy, T, is:

$$T = \tfrac{1}{2}I_h\dot{q}_0^2 + \tfrac{1}{2}\rho\dot{q}_0^2 \int_0^l (x^2 + v^2)dx + \tfrac{1}{2}\rho \int_0^l \dot{v}^2 dx + \dot{q}_0\rho \int_0^l \dot{v}x dx \qquad (3.27)$$

Now, from equation (3.26):

$$\begin{aligned} v &\approx \sum_{i=1}^{n} \phi_i q_i \\ \therefore v^2 &\approx \sum_{i=1}^{n}\sum_{j=1}^{n} \phi_i\phi_j q_i q_j \\ \therefore \int_0^l v^2 dx &\approx \sum_{i=1}^{n}\sum_{j=1}^{n} \int_0^l \phi_i\phi_j dx q_i q_j \\ &= \sum_{i=1}^{n} \int_0^l \phi_i^2 dx q_i^2 \qquad \text{by orthogonality} \\ &= \sum_{i=1}^{n} q_i^2 \qquad \text{by normalization} \end{aligned}$$

Similarly,

$$\begin{aligned}\int_0^l \dot{v}^2 dx &\approx \sum_{i=1}^n \int_0^l \phi_i^2 dx \dot{q}_i^2 && \text{by orthogonality}\\ &= \sum_{i=1}^n \dot{q}_i^2 && \text{by normalization}\end{aligned}$$

Substituting this result into equation (3.27), the kinetic energy in terms of the eigenfunctions and generalized coordinates is:

$$T = \tfrac{1}{2}\left(I_h + \tfrac{1}{3}\rho l^3 + \rho \sum_{i=1}^n q_i^2\right)\dot{q}_0^2 + \tfrac{1}{2}\rho \sum_{i=1}^n \dot{q}_i^2 + \rho \dot{q}_0 \sum_{i=1}^n \dot{q}_i \int_0^l x\phi_i dx$$

Potential Energy for the Single Flexible Link

As derived previously, the potential energy for the single flexible link is:

$$V = \tfrac{1}{2}\int_0^l EI(v'')^2 dx$$

Let λ_i and ω_i be defined as in Appendix B for $i = 1 \ldots n$. From equation (3.26):

$$\begin{aligned}v'' &\approx \sum_{i=1}^n \phi_i'' q_i\\ \therefore (v'')^2 &= \sum_{i=1}^n \sum_{j=1}^n \phi_i'' \phi_j'' q_i q_j\\ \therefore \int_0^l (v'')^2 dx &\approx \sum_{i=1}^n \sum_{j=1}^n q_i q_j \int_0^l \phi_i'' \phi_j'' dx\\ &= \sum_{i=1}^n q_i^2 \int_0^l (\phi_i'')^2 dx && \text{by orthogonality}\\ &= \sum_{i=1}^n \lambda_i q_i^2 && \text{by normalization and (B.16)}\end{aligned}$$

Thus:

$$V \approx \tfrac{1}{2} EI \sum_{i=1}^{n} \lambda_i q_i^2$$
$$= \tfrac{1}{2} \rho \sum_{i=1}^{n} \omega_i^2 q_i^2 \qquad \text{by (B.11)}$$

Euler-Lagrange Equations for the Single Flexible Link

Combining the expressions for the potential and kinetic energies, the Lagrangian of the system, $L = T - V$, is:

$$L = \tfrac{1}{2} \left(I_h + \tfrac{1}{3}\rho l^3 + \rho \sum_{i=1}^{n} q_i^2 \right) \dot{q}_0^2 + \tfrac{1}{2}\rho \sum_{i=1}^{n} \dot{q}_i^2 + \rho \dot{q}_0 \sum_{i=1}^{n} \dot{q}_i \int_0^l x \phi_i dx - \tfrac{1}{2}\rho \sum_{i=1}^{n} \omega^2 q_i^2$$

Let τ_i be the generalized force acting upon the generalized coordinate, q_i. The Euler-Lagrange equations stipulate, for each generalized coordinate:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \tau_i, \quad i = 0, \ldots, n \tag{3.28}$$

Now, for the rigid body mode, corresponding to generalized coordinate, q_0:

$$\frac{\partial L}{\partial q_0} = 0$$
$$\frac{\partial L}{\partial \dot{q}_0} = \left(I_h + \tfrac{1}{3}\rho l^3 + \rho \sum_{i=1}^{n} q_i^2 \right) \dot{q}_0 + \rho \sum_{i=1}^{n} \dot{q}_i \int_0^l x \phi_i dx$$

Hence, if τ is the torque exerted by the motor at the hub, then by equation (3.28):

$$\left(I_h + \tfrac{1}{3}\rho l^3 + \rho \sum_{i=1}^{n} q_i^2 \right) \ddot{q}_0 + 2\rho \sum_{i=1}^{n} q_i \dot{q}_i \dot{q}_0 + \rho \sum_{i=1}^{n} \ddot{q}_i \int_0^l x \phi_i dx = \tau \tag{3.29}$$

For the kth flexible mode, corresponding to generalized coordinate, q_k:

$$\frac{\partial L}{\partial q_k} = -\rho q_k \omega_k^2 + \rho q_k \dot{q}_0^2$$
$$\frac{\partial L}{\partial \dot{q}_k} = \rho \dot{q}_k + \rho \dot{q}_0 \int_0^l x \phi_k dx$$

Since there is no generalized force acting upon the flexible modes, $\tau_k = 0$. Hence:

$$\rho\ddot{q}_k + \rho\ddot{q}_0 \int_0^l x\phi_k dx - \rho q_k \dot{q}_0^2 + \rho q_k \omega_k^2 = 0 \tag{3.30}$$

Equations (3.29) and (3.30) define the dynamic equations for the single flexible link shown in Figure 3.1. Define:

$$\beta_i = \int_0^l x\phi_i(x)dx$$

$$\mathbf{q} = [q_0 \quad \cdots \quad q_n]^{\mathrm{T}}$$

$$\mathbf{T} = [\tau \quad 0 \quad \cdots \quad 0]^{\mathrm{T}}$$

$$M(\mathbf{q}) = \begin{bmatrix} I_h + \frac{1}{3}ml^2 + \rho\sum_{i=1}^n q_i^2 & \rho\beta_1 & \cdots & \rho\beta_n \\ \rho\beta_1 & \rho & & \\ \vdots & & \ddots & \\ \rho\beta_n & & & \rho \end{bmatrix}$$

$$\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) = \begin{bmatrix} 2\rho\left(\sum_{i=1}^n q_i\dot{q}_i\right)\dot{q}_0 \\ -\rho q_1\dot{q}_0^2 \\ \vdots \\ -\rho q_n\dot{q}_0^2 \end{bmatrix}$$

$$K = \begin{bmatrix} 0 & & & \\ & \rho\omega_1^2 & & \\ & & \ddots & \\ & & & \rho\omega_n^2 \end{bmatrix} = \begin{bmatrix} 0 & & & \\ & EI\lambda_1 & & \\ & & \ddots & \\ & & & EI\lambda_n \end{bmatrix}$$

Then equations (3.29) and (3.30) may be written in matrix form as:

$$M(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) + K\mathbf{q} = \mathbf{T} \tag{3.31}$$

Note that the inertia matrix is symmetric.

Only $M(\mathbf{q})$ and $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ are functions of the generalized coordinates. Define:

$$\overline{M} = \begin{bmatrix} I_h + \frac{1}{3}ml^2 & \rho\beta_1 & \cdots & \rho\beta_n \\ \rho\beta_1 & \rho & & \\ \vdots & & \ddots & \\ \rho\beta_n & & & \rho \end{bmatrix}$$

It is easily shown that by linearizing (3.31) about the stationary operating point $\mathbf{q} = 0$ and $\dot{\mathbf{q}} = 0$, the inertia matrix becomes $\overline{M}$ and the centripetal and Coriolis terms disappear. To put the linearized equations into state-space form, define the state vector:

$$\mathbf{x} = [q_0 \quad \cdots \quad q_n \quad \dot{q}_0 \quad \cdots \quad \dot{q}_n]^{\mathrm{T}}$$

Let the output, y, be defined in traditional fashion as the arc length from the base frame to the tip of the beam:

$$\begin{aligned} y(t) &= lq_0(t) + v(l,t) \\ &\approx lq_0(t) + \sum_{i=1}^{n} \phi_i(l) q_i(t) \end{aligned}$$

Since small deflections have been assumed, the distance $v(l,t)$ may be used to represent the tip deflection arc. The arc length is a natural displacement for the rotary single flexible link and its use avoids the trigonometric relationships inherent to a Cartesian displacement. Let $\Psi = [l \quad \phi_1(l) \quad \cdots \quad \phi_n(l)]$. Then, the dynamic equations in state-space are:

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{bmatrix} 0 & I_n \\ -\overline{M}^{-1}K & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ \overline{M}^{-1} \end{bmatrix} \mathbf{T} \\ y &= \begin{bmatrix} \Psi & 0 \end{bmatrix} \mathbf{x} \end{aligned}$$

3.4 Forward Kinematics

The dynamic equations for a simple single flexible link experiment have been derived in the preceding section. The full dynamic equations for a flexible link mounted upon an arbitrary base robot will now be considered, using the same assumed modes approach of Meirovitch [81] for modelling the flexibility that was employed in the preceding section. The flexible link attached to a base robot is shown schematically in Figure 3.3. Let:

x_0-y_0-z_0 = base frame of the base robot

x_e-y_e-z_e = frame of attached end of flexible link

l = length of flexible beam

m = degrees of freedom of base robot

$\mathbf{q}(t)$ = generalized coordinate vector of base robot, $\mathbf{q}(t) \in \Re^m$

$H_0^e(\mathbf{q}, t)$ = homogeneous transformation relating base frame to flexible frame

$v(x, t)$ = deflection of flexible beam in y_e direction

$w(x, t)$ = deflection of flexible beam in z_e direction

The undeflected beam lies along the x_e axis. The beam is assumed to be a line. Hence, there are no rotational inertia effects. It should be noted that in specific circumstances, this assumption can lead to misleading results. In particular, suppose the base robot consists of only a rotary motor and the flexible beam is mounted such that its neutral axis corresponds to the axis of rotation. Assume the actuator has no rotational inertia. Since the beam is also modelled as having no rotational inertia, the kinetic energy of the beam will be zero and the resulting inertia matrix will be singular. Such singularity is not possible for a real physical system. However, by accounting for the rotational inertia of the actuator, the singularity

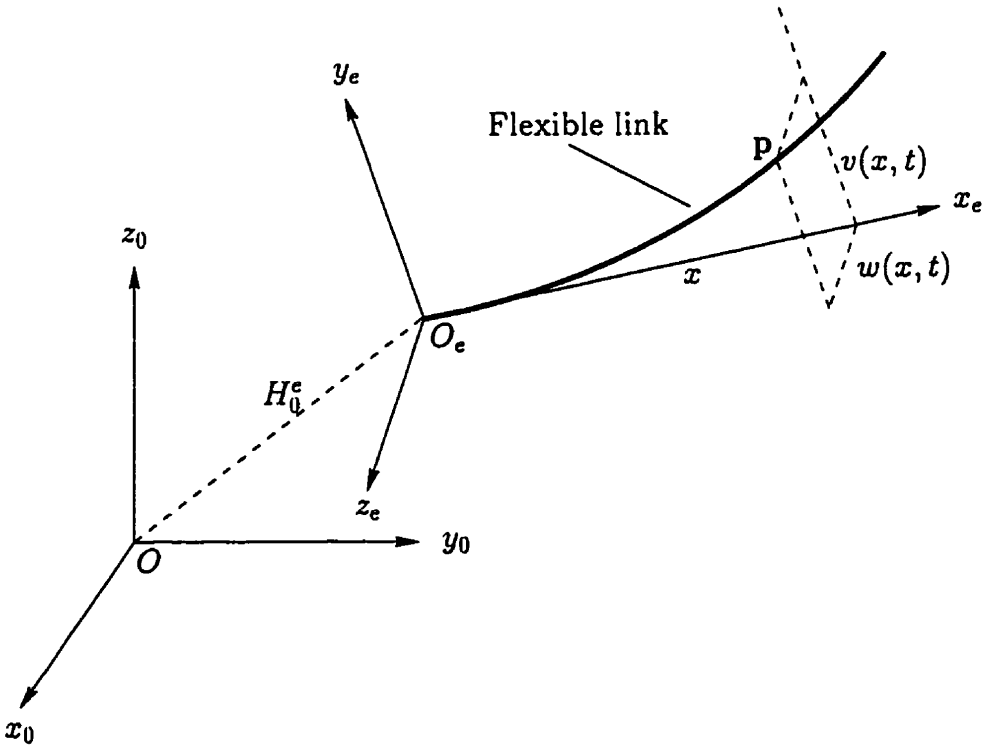


Figure 3.3: Flexible Beam Mounted on a Base Robot

problem disappears and the model is more realistic. Generally, a base robot has non-zero inertia with respect to each axis so the assumption that the beam has no rotational inertia is reasonable and will not result in a singular inertia matrix.

It is further assumed that beam deflections are small and extension or compression of the beam along the x_e axis is negligible. Hence, the point p expressed in frame e coordinates is simply:

$$\mathbf{p}(x,t) = [x \quad v(x,t) \quad w(x,t)]^{\mathrm{T}} \tag{3.32}$$

The assumed modes approach will be used for modelling the flexible beam. Define,

for $i = 1 \ldots \infty$:

$\varphi_i(x)$ = mode shapes of the beam in the y_e direction

$\nu_i(t)$ = generalized coordinates of the beam in the y_e direction

$\psi_i(x)$ = mode shapes of the beam in the z_e direction

$\eta_i(t)$ = generalized coordinates of the beam in the z_e direction

Thus, the beam deflections $v(x,t)$ and $w(x,t)$ may be written:

$$v(x,t) = \sum_{i=1}^{\infty} \varphi_i(x)\nu_i(t) \tag{3.33}$$

$$w(x,t) = \sum_{i=1}^{\infty} \psi_i(x)\eta_i(t) \tag{3.34}$$

For convenience, define the n-vectors:

$$\boldsymbol{\varphi}(x) \triangleq [\varphi_1(x) \quad \cdots \quad \varphi_n(x)]^{\mathrm{T}} \in \Re^{n\times 1} \tag{3.35}$$

$$\boldsymbol{\psi}(x) \triangleq [\psi_1(x) \quad \cdots \quad \psi_n(x)]^{\mathrm{T}} \in \Re^{n\times 1} \tag{3.36}$$

$$\boldsymbol{\nu}(t) \triangleq [\nu_1(t) \quad \cdots \quad \nu_n(t)]^{\mathrm{T}} \in \Re^{n\times 1} \tag{3.37}$$

$$\boldsymbol{\eta}(t) \triangleq [\eta_1(t) \quad \cdots \quad \eta_n(t)]^{\mathrm{T}} \in \Re^{n\times 1} \tag{3.38}$$

and write the base robot generalized coordinates as:

$$\mathbf{q}(t) = [q_0(t) \quad \cdots \quad q_{m-1}(t)]^{\mathrm{T}} \in \Re^{m\times 1} \tag{3.39}$$

Hence, by (3.33) and (3.34):

$$v(x,t) \approx \boldsymbol{\varphi}(x)^{\mathrm{T}}\boldsymbol{\nu}(t) \tag{3.40}$$

$$w(x,t) \approx \boldsymbol{\psi}(x)^{\mathrm{T}}\boldsymbol{\eta}(t) \tag{3.41}$$

3.4.1 Translation

The homogeneous transformation H_0^e of Figure 3.3 may be expressed in terms of a rotation matrix, $R = R_0^e(\mathbf{q}, t) \in SO(3)$ in conjunction with a translation vector, $\mathbf{d} = \mathbf{d}_0^e(\mathbf{q}, t) \in \Re^{3\times 1}$, as follows [12]:

$$H_0^e = \begin{bmatrix} R & \mathbf{d} \\ 0 & 1 \end{bmatrix} \tag{3.42}$$

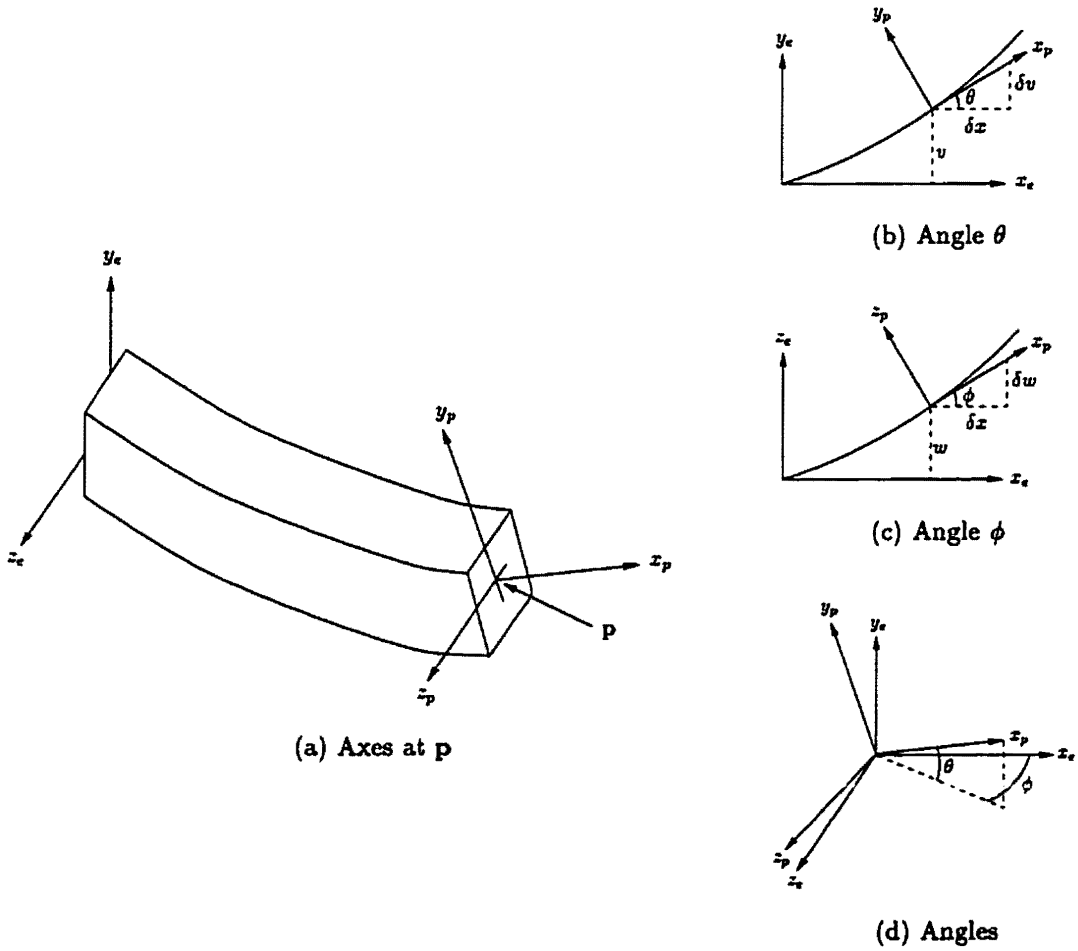
Similarly, the coordinates of the point $\mathbf{p}$ on the flexible beam, expressed in *base frame* coordinates, $\mathbf{p}_0$, represents a translation from the base frame to the point $\mathbf{p}$ on the beam. Clearly by (3.40)-(3.42):

$$\mathbf{p}_0 = H_0^e \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix} \approx R \begin{bmatrix} x \\ \boldsymbol{\varphi}^{\mathrm{T}}\boldsymbol{\nu} \\ \boldsymbol{\psi}^{\mathrm{T}}\boldsymbol{\eta} \end{bmatrix} + \mathbf{d} \tag{3.43}$$

3.4.2 Orientation

In order to have a complete homogeneous transformation from the base frame to a point on the flexible beam, the orientation of the beam at the point, $\mathbf{p}$, must also be considered. Observe the frame attached to point, $\mathbf{p}$, in Figure 3.4(a). The x_p axis of the frame is tangential to the neutral axis of the beam. The y_p-z_p plane lies in a planar cross-section of the beam at point, $\mathbf{p}$.

The tip deflection of the beam is modelled by two displacements: v and w, in the y_e and z_e directions respectively. Rotation about the x_e axis, or roll, is not considered so the orientation of the frame x_p-y_p-z_p may be expressed by two angles, θ and ϕ, representing pitch and yaw respectively. These two angles are illustrated individually in Figures 3.4(b) and 3.4(c), and together in Figure 3.4(d).



(a) Axes at **p**

(b) Angle θ

(c) Angle ϕ

(d) Angles

Figure 3.4: Flexible Beam Orientation at Point, **p**, Along the Beam

It is clear from Figure 3.4(d) that the transformation from frame e to frame p may be accomplished by first rotating frame e through an angle ϕ about the y_e axis to get a new frame, frame e'. Denote this new frame by x'_e-y'_e-z'_e. Rotate frame e' about its z'_e axis to get frame p. Since small deflections are assumed, the same result may be obtained by rotating about the z_e axis first and then about the y'_e axis.

Let $R_{y,\phi}$ be the rotation through angle ϕ about the y_e-axis and $R_{z,\theta}$ be the rotation by θ about the z'_e-axis. Then the net rotation matrix, R^p_e, from frame e to frame p is [12]:

$$R^p_e = R_{y,\phi}R_{z,\theta} = \begin{bmatrix} \cos(\theta)\cos(\phi) & -\sin(\theta)\cos(\phi) & \sin(\phi) \\ \sin(\theta) & \cos(\theta) & 0 \\ -\cos(\theta)\sin(\phi) & \sin(\theta)\sin(\phi) & \cos(\phi) \end{bmatrix} \tag{3.44}$$

For convenience, define:

$$(\cdot)' = \frac{\partial(\cdot)}{\partial x}$$

From Figures 3.4(b) and 3.4(c):

$$\theta = \tan^{-1}(v'), \qquad \sin(\theta) = \frac{v'}{\sqrt{1+(v')^2}}, \qquad \cos(\theta) = \frac{1}{\sqrt{1+(v')^2}} \tag{3.45}$$

$$\phi = \tan^{-1}(w'), \qquad \sin(\phi) = \frac{w'}{\sqrt{1+(w')^2}}, \qquad \cos(\phi) = \frac{1}{\sqrt{1+(w')^2}} \tag{3.46}$$

Assuming small deflections, let $(v')^2 = (w')^2 = v'w' \approx 0$. Hence, from (3.44)-(3.46):

$$R^p_e \approx \begin{bmatrix} 1 & -v' & w' \\ v' & 1 & 0 \\ -w' & 0 & 1 \end{bmatrix}$$

Thus, by (3.40) and (3.41), the overall rotation between the base frame, x_0-y_0-z_0,

and frame x_p-y_p-z_p is:

$$R_0^p = RR_e^p \approx R \begin{bmatrix} 1 & -\nu^T\varphi' & \eta^T\psi' \\ \nu^T\varphi' & 1 & 0 \\ -\eta^T\psi' & 0 & 1 \end{bmatrix} \tag{3.47}$$

Equations (3.43) and (3.47) define the homogeneous transformation between the base frame and the frame x_p-y_p-z_p. The position and orientation of the tip of the beam are found by evaluating equations (3.43) and (3.47) at $x = l$.

3.5 Velocity Kinematics

The preceding sections derived the position and orientation of a point, $\mathbf{p}$, on the flexible beam with respect to the base frame. In order to determine the kinetic energy of the system, the velocity of each point on the beam must also be known. Recall the position, $\mathbf{p}_0$, of a point $\mathbf{p}$ on the beam from (3.43).

$$\mathbf{p}_0 = R\mathbf{p} + \mathbf{d} \qquad \text{where} \quad \mathbf{p} = \begin{bmatrix} x \\ \varphi^T\nu \\ \psi^T\eta \end{bmatrix}$$

Let:

ω = angular velocity of frame e expressed in *base frame* coordinates

Then by the chain rule of differentiation:

$$\begin{aligned} \dot{\mathbf{p}}_0 &= R\dot{\mathbf{p}} + \dot{R}\mathbf{p} + \dot{\mathbf{d}} \\ &= R\dot{\mathbf{p}} + \mathrm{S}(\omega)R\mathbf{p} + \dot{\mathbf{d}} && \text{using (D.7)} \\ &= R\dot{\mathbf{p}} + R\,\mathrm{S}(R^T\omega)\mathbf{p} + \dot{\mathbf{d}} && \text{using (D.6)} \end{aligned}$$

But $R^{\mathrm{T}}\omega$ is simply the angular velocity of frame e expressed in frame e coordinates. Define:

$$\Omega \triangleq R^{\mathrm{T}}\omega \tag{3.48}$$

Hence:

$$\begin{aligned} \dot{\mathrm{p}}_0 &= R\dot{\mathrm{p}} + R\mathrm{S}(\Omega)\mathrm{p} + \dot{\mathrm{d}} \\ &= R\dot{\mathrm{p}} - R\mathrm{S}(\mathrm{p})\Omega + \dot{\mathrm{d}} \qquad \text{by (D.4)} \end{aligned} \tag{3.49}$$

Note that R, d and Ω are strictly functions of the base robot generalized coordinates, q, while p is strictly a function of the flexible coordinates, ν and η.

Now, the angular velocity, Ω, may always be expressed as:

$$\Omega = J^{\Omega}\dot{\mathrm{q}} \qquad \text{where} \quad J^{\Omega}(\mathrm{q}) \in \Re^{3\times m} \tag{3.50}$$

This fact is easily shown by considering the inverse of the skew-symmetric operator $\mathrm{S}(\cdot)$. Let $A = [a_{ij}]$ be an arbitrary matrix in $SS(3)$. Define the *inverse skew-symmetric operator*, $\mathrm{S}^{-1}(A) : SS(3) \mapsto \Re^3$, by:

$$\mathrm{S}^{-1}(A) \triangleq \begin{bmatrix} a_{32} \\ a_{13} \\ a_{21} \end{bmatrix} \tag{3.51}$$

The inverse skew-symmetric operator is clearly linear. From (D.6)-(D.7) and (3.48):

$$\begin{aligned} \mathrm{S}(\Omega) &= R^{\mathrm{T}}\dot{R} \\ \therefore \Omega &= \mathrm{S}^{-1}(R^{\mathrm{T}}\dot{R}) && \text{using (3.51)} \\ &= \mathrm{S}^{-1}\left(R^{\mathrm{T}}\sum_{i=0}^{m-1}\frac{\partial R}{\partial q_i}\dot{q}_i\right) && \text{using the chain rule} \\ &= \sum_{i=0}^{m-1}\mathrm{S}^{-1}\left(R^{\mathrm{T}}\frac{\partial R}{\partial q_i}\right)\dot{q}_i && \text{by linearity of } \mathrm{S}^{-1}(\cdot) \\ &= \left[\mathrm{S}^{-1}\left(R^{\mathrm{T}}\tfrac{\partial R}{\partial q_0}\right) \;\cdots\; \mathrm{S}^{-1}\left(R^{\mathrm{T}}\tfrac{\partial R}{\partial q_{m-1}}\right)\right]\dot{\mathrm{q}} \end{aligned}$$

Hence,

$$J^{\Omega} = \left[\mathrm{S}^{-1}\left(R^{\mathrm{T}} \frac{\partial R}{\partial q_0}\right) \quad \cdots \quad \mathrm{S}^{-1}\left(R^{\mathrm{T}} \frac{\partial R}{\partial q_{m-1}}\right) \right] \tag{3.52}$$

Now, define the $3 \times m$ matrix:

$$J^{v} \triangleq \frac{\partial \mathbf{d}}{\partial \mathbf{q}} \tag{3.53}$$

Using (3.15), (3.22), (3.50) and (3.53) in (3.49):

$$\begin{aligned} \dot{\mathbf{p}}_0 &= R\dot{\mathbf{p}} - R\,\mathrm{S}(\mathbf{p})J^{\Omega}\dot{\mathbf{q}} + J^{v}\dot{\mathbf{q}} \\ &= R[0 \quad \dot{v} \quad \dot{w}]^{\mathrm{T}} + (J^{v} - R\,\mathrm{S}(\mathbf{p})J^{\Omega})\dot{\mathbf{q}} && \text{since } \dot{x} = 0 \\ &\approx R[0 \quad \boldsymbol{\varphi}^{\mathrm{T}}\dot{\boldsymbol{\nu}} \quad \boldsymbol{\psi}^{\mathrm{T}}\dot{\boldsymbol{\eta}}]^{\mathrm{T}} + (J^{v} - R\,\mathrm{S}(\mathbf{p})J^{\Omega})\dot{\mathbf{q}} && \text{from (3.40) and (3.41)} \end{aligned} \tag{3.54}$$

Let:

$$\Phi(x) \triangleq \begin{bmatrix} 0 & \boldsymbol{\varphi}(x) & 0 \\ 0 & 0 & \boldsymbol{\psi}(x) \end{bmatrix} \in \Re^{2n \times 3} \quad \text{and} \quad \mathbf{Q}(t) \triangleq \begin{bmatrix} \mathbf{q}(t) \\ \boldsymbol{\nu}(t) \\ \boldsymbol{\eta}(t) \end{bmatrix} \in \Re^{(m+2n)\times 1} \tag{3.55}$$

and define the $3 \times (m + 2n)$ Jacobian matrix:

$$J \triangleq \left[(J^{v} - R\,\mathrm{S}(\mathbf{p})J^{\Omega}) \quad R\Phi^{\mathrm{T}} \right] \tag{3.56}$$

Note that only $\mathrm{S}(\mathbf{p})$ and Φ are functions of x. Combining equations (3.54) and (3.56):

$$\dot{\mathbf{p}}_0 = J \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\boldsymbol{\nu}} \\ \dot{\boldsymbol{\eta}} \end{bmatrix} = J\dot{\mathbf{Q}} \tag{3.57}$$

This equation provides the relationship between the generalized coordinate derivatives and the velocity of each point on the flexible link.

3.6 Force-Torque Relationship

Of obvious interest for force control is the relationship between the torques exerted by the robot actuators and the force at the end of the flexible beam. Let:

$\mathbf{F}$ = generalized force exerted at a point x along the beam

$\mathbf{X}$ = displacement of the beam element at x

$\mathbf{dX}$ = incremental displacement of the beam element at x

τ = generalized forces exerted by the robot actuators

$\mathbf{dQ}$ = incremental change in the generalized coordinates

By the Principle of Virtual Work [12]:

$$\mathbf{F}^{\mathrm{T}}\mathbf{dX} = \tau^{\mathrm{T}}\mathbf{dQ}$$

But from (3.57), $\dot{\mathbf{X}} = J\dot{\mathbf{Q}}$. Hence $\mathbf{dX} \approx J\mathbf{dQ}$, whence:

$$\mathbf{F}^{\mathrm{T}}J\mathbf{dQ} = \tau^{\mathrm{T}}\mathbf{dQ}$$

$$\tau = J^{\mathrm{T}}\mathbf{F} \tag{3.58}$$

Thus, given a desired end-point force, or force anywhere along the flexible beam, the required joint torques may be readily calculated using (3.58). Note, however, that since equation (3.57) relates the coordinates of the end-effector in *base frame coordinates* to the generalized coordinates, then the force, $\mathbf{F}$, in (3.58) is also in base frame coordinates, *not* end-effector coordinates.

3.7 Kinetic Energy

In order to derive the dynamic equations using the Euler-Lagrange approach, the kinetic energy is required. Since no restrictions have been placed on the base robot

up to this point, only the kinetic energy of the flexible beam will be considered. The kinetic energy of the base robot may be derived separately since energies are additive. Assuming the beam has uniform linear mass density, let:

$$K_b = \text{kinetic energy of the base robot}$$

$$K_f = \text{kinetic energy of the flexible beam}$$

$$m_f = \text{mass of the flexible beam}$$

$$\rho = \text{linear mass density of the flexible beam}$$

From standard vibration theory:

$$K_f = \tfrac{1}{2}\int \dot{\mathbf{p}}_0^{\mathrm{T}}\dot{\mathbf{p}}_0 dm \tag{3.59}$$

Substituting (3.57) into (3.59):

$$K_f = \tfrac{1}{2}\rho\int_0^l \dot{\mathbf{Q}}^{\mathrm{T}} J^{\mathrm{T}} J\dot{\mathbf{Q}} dx$$

But $\mathbf{Q}$ is independent of x, hence:

$$K_f = \tfrac{1}{2}\rho\dot{\mathbf{Q}}^{\mathrm{T}}\int_0^l J^{\mathrm{T}} J dx\dot{\mathbf{Q}} \tag{3.60}$$

Thus, a quantity of interest is the symmetric matrix, $J^{\mathrm{T}}J$. By equation (3.56):

$$\begin{aligned} J^{\mathrm{T}}J &= \left[(J^v - R\,\mathrm{S}(\mathbf{p})J^\Omega) \quad R\Phi^{\mathrm{T}}\right]^{\mathrm{T}}\left[(J^v - R\,\mathrm{S}(\mathbf{p})J^\Omega) \quad R\Phi^{\mathrm{T}}\right] \\ &= \begin{bmatrix} (J^v - R\,\mathrm{S}(\mathbf{p})J^\Omega)^{\mathrm{T}}(J^v - R\,\mathrm{S}(\mathbf{p})J^\Omega) & (J^v - R\,\mathrm{S}(\mathbf{p})J^\Omega)^{\mathrm{T}}R\Phi^{\mathrm{T}} \\ \Phi R^{\mathrm{T}}(J^v - R\,\mathrm{S}(\mathbf{p})J^\Omega) & \Phi R^{\mathrm{T}}R\Phi^{\mathrm{T}} \end{bmatrix} \end{aligned}$$

For convenience, define the following quantities:

$$\chi_1 \triangleq \mathrm{S}(\mathbf{p}) \qquad \chi_2 \triangleq \Phi\,\mathrm{S}(\mathbf{p})$$

$$\chi_3 \triangleq \mathrm{S}(\mathbf{p})^{\mathrm{T}}\mathrm{S}(\mathbf{p}) \qquad \chi_4 \triangleq \Phi\Phi^{\mathrm{T}}$$

and

$$\varpi \triangleq \Phi R^{\mathrm{T}} J^{v} - \chi_2 J^{\Omega}$$

whence, using the orthogonality of rotation matrices:

$$J^{\mathrm{T}} J = \begin{bmatrix} \left((J^{v})^{\mathrm{T}} J^{v} - (J^{\Omega})^{\mathrm{T}} \chi_1^{\mathrm{T}} R^{\mathrm{T}} J^{v} - (J^{v})^{\mathrm{T}} R \chi_1 J^{\Omega} + (J^{\Omega})^{\mathrm{T}} \chi_3 J^{\Omega}\right) & \varpi^{\mathrm{T}} \\ \varpi & \chi_4 \end{bmatrix} \tag{3.61}$$

Only χ_1 through χ_4 and ϖ are functions of x in this equation. Indeed, in ϖ, only Φ and χ_2 are functions of x. Hence, to solve for the integral in equation (3.60), the following quantities must be evaluated:

$$\alpha_v \triangleq \int_0^l \varphi dx \qquad \alpha_w \triangleq \int_0^l \psi dx \tag{3.62}$$

and the integrals:

$$\Xi_1 \triangleq \int_0^l \mathrm{S}(\mathbf{p}) dx \qquad \Xi_2 \triangleq \int_0^l \Phi\, \mathrm{S}(\mathbf{p}) dx \tag{3.63}$$

$$\Xi_3 \triangleq \int_0^l \mathrm{S}(\mathbf{p})^{\mathrm{T}} \mathrm{S}(\mathbf{p}) dx \qquad \Xi_4 \triangleq \int_0^l \Phi \Phi^{\mathrm{T}} dx \tag{3.64}$$

Now consider each of the integrals, Ξ_1 through Ξ_4. Each of the terms has particular significance that will be discussed in the ensuing subsections.

3.7.1 Centre of Mass

The quantity Ξ_1 is related to the centre of mass of the flexible link with respect to the clamped frame. For, by definition of the centre of mass, $\mathbf{p}_{cm}$:

$$\mathbf{p}_{cm} = \frac{1}{m_f} \int \mathbf{p} dm = \frac{1}{l} \int_0^l \mathbf{p} dx$$

$$\therefore \mathrm{S}(\mathbf{p}_{cm}) = \frac{1}{l} \int_0^l \mathrm{S}(\mathbf{p}) dx, \quad \text{by linearity of S}$$

from which:

$$\Xi_1 = l\ \mathrm{S}(\mathbf{p}_{cm}) \tag{3.65}$$

Since the generalized coordinates ν and η are independent of x, Ξ_1 may be expressed in block matrix form, using (D.1), as:

$$\Xi_1 \approx \begin{bmatrix} 0 & -\alpha_w^{\mathrm{T}}\eta & \alpha_v^{\mathrm{T}}\nu \\ \alpha_w^{\mathrm{T}}\eta & 0 & -\frac{1}{2}l^2 \\ -\alpha_v^{\mathrm{T}}\nu & \frac{1}{2}l^2 & 0 \end{bmatrix} \tag{3.66}$$

Note that α_v and α_w are constant, so Ξ_1 is a linear function of the generalized flexible coordinates, ν and η. Also, by setting $\nu = 0$ and $\eta = 0$, the centre of mass of a rigid beam may be extracted. In this manner, Ξ_1 may be separated into a "rigid" component and flexible "correction" terms. Also, it is clear from (3.66) that:

$$\mathbf{p}_{cm} = \tfrac{1}{l}[\tfrac{1}{2}l^2 \quad \nu^{\mathrm{T}}\alpha_v \quad \eta^{\mathrm{T}}\alpha_w]^{\mathrm{T}} \tag{3.67}$$

3.7.2 Cross-Coupling

The quantity Ξ_2 is part of the cross-coupling terms in the $J^{\mathrm{T}}J$ matrix. In particular, it will be shown that Ξ_2 includes the coupling terms between the two modes of vibration. Also, it should be noted that Ξ_2 contains no contributions from the base robot. It is purely a flexible term.

From equations (D.1) and (3.55):

$$\begin{aligned} \Phi\,\mathrm{S}(\mathbf{p}) &\approx \begin{bmatrix} \varphi w & 0 & -\varphi x \\ -\psi v & \psi x & 0 \end{bmatrix} \\ &= \begin{bmatrix} \varphi\psi^{\mathrm{T}}\eta & 0 & -\varphi x \\ -\psi\varphi^{\mathrm{T}}\nu & \psi x & 0 \end{bmatrix} \end{aligned} \tag{3.68}$$

Define the constant $n \times 1$ vectors β_v and β_w, and the constant $n \times n$ matrix, γ, as follows:

$$\beta_v \triangleq \int_0^l \varphi x dx \tag{3.69}$$

$$\beta_w \triangleq \int_0^l \psi x dx \tag{3.70}$$

$$\gamma \triangleq \int_0^l \varphi \psi^{\mathrm{T}} dx \tag{3.71}$$

Hence:

$$\Xi_2 = \int_0^l \Phi \, \mathrm{S}(\mathbf{p}) dx \approx \begin{bmatrix} \gamma\eta & 0 & -\beta_v \\ -\gamma^{\mathrm{T}}\nu & \beta_w & 0 \end{bmatrix} \tag{3.72}$$

The quantity γ is worth further consideration. Consider each element of the matrix γ:

$$\gamma_{ij} = \int_0^l \varphi_i \psi_j dx, \quad i, j = 1, \ldots, n$$

γ clearly represents all the cross-coupling terms between the two modes of vibration. However, consider the factors which determine the mode shapes. From equation (B.8) in Appendix B, when an Euler-Bernoulli beam model is assumed the mode shapes take the same form regardless of the physical parameters:

$$\varphi_i(x) = C_1 \cos(p_i x) + C_2 \sin(p_i x) + C_3 \cosh(p_i x) + C_4 \sinh(p_i x)$$

This mode shape is the general solution to the unforced Euler-Bernoulli beam equation. The constants are solved by substituting the appropriate boundary conditions, as well as the modulus of elasticity, moment area of inertia, and linear mass density. The frequencies, p_i, of the mode shapes also depend on these boundary conditions. For example, for the clamped-free mode shapes in Appendix B, the modal frequencies are determined by the characteristic equation (B.9):

$$\cos(pl)\cosh(pl) = -1$$

The clamped-free mode shapes are also orthonormal, another property that often arises. Hence, if the boundary conditions and moment areas of inertia are the same for both directions of vibration (for a uniform beam), then the mode shapes, φ and ψ, will be identical. In this case, assuming orthonormality of the mode shapes, $\gamma = I$.

3.7.3 Moment of Inertia of the Flexible Beam

Consider the elements of the matrix χ_3:

$$\chi_3 = \mathrm{S(p)}^{\mathrm{T}}\,\mathrm{S(p)} = (\mathrm{p}^{\mathrm{T}}\mathrm{p})I - \mathrm{pp}^{\mathrm{T}} = \begin{bmatrix} v^2+w^2 & -xv & -xw \\ -xv & x^2+w^2 & -vw \\ -xw & -vw & x^2+v^2 \end{bmatrix}$$

$$= \begin{bmatrix} \nu^{\mathrm{T}}\varphi\varphi^{\mathrm{T}}\nu + \eta^{\mathrm{T}}\psi\psi^{\mathrm{T}}\eta & -x\varphi^{\mathrm{T}}\nu & -x\psi^{\mathrm{T}}\eta \\ -x\varphi^{\mathrm{T}}\nu & x^2+\eta^{\mathrm{T}}\psi\psi^{\mathrm{T}}\eta & -\nu^{\mathrm{T}}\varphi\psi^{\mathrm{T}}\eta \\ -x\psi^{\mathrm{T}}\eta & -\nu^{\mathrm{T}}\varphi\psi^{\mathrm{T}}\eta & x^2+\nu^{\mathrm{T}}\varphi\varphi^{\mathrm{T}}\nu \end{bmatrix}$$

From the form of χ_3, clearly $\rho\Xi_3 = \int \chi_3 dm$ is the mass moment of inertia. Assuming orthonormality of the mode shapes:

$$\int_0^l \varphi\varphi^{\mathrm{T}}dx = I \qquad \text{and} \qquad \int_0^l \psi\psi^{\mathrm{T}}dx = I$$

Hence:

$$\Xi_3 = \int_0^l \mathrm{S(p)}^{\mathrm{T}}\,\mathrm{S(p)}dx = \begin{bmatrix} \nu^{\mathrm{T}}\nu + \eta^{\mathrm{T}}\eta & -\beta_v^{\mathrm{T}}\nu & -\beta_w^{\mathrm{T}}\eta \\ -\beta_v^{\mathrm{T}}\nu & \frac{1}{3}l^3 + \eta^{\mathrm{T}}\eta & -\nu^{\mathrm{T}}\gamma\eta \\ -\beta_w^{\mathrm{T}}\eta & -\nu^{\mathrm{T}}\gamma\eta & \frac{1}{3}l^3 + \nu^{\mathrm{T}}\nu \end{bmatrix} \tag{3.73}$$

Note once again that if the mode shapes are the same for each direction and are orthonormal, then $\gamma = I$. Also observe that Ξ_3 is quadratic in the generalized coordinates.

3.7.4 Purely Flexible Component

The quantity Ξ_4 represents the purely flexible component of the kinetic energy term. From the definition of Φ in equation (3.55), the integrand of Ξ_4 may be expressed in block matrix form as:

$$\Phi\Phi^{\mathrm{T}} = \begin{bmatrix} \varphi\varphi^{\mathrm{T}} & 0 \\ 0 & \psi\psi^{\mathrm{T}} \end{bmatrix}$$

Assuming orthonormality of the mode shapes, clearly:

$$\Xi_4 = \int_0^l \Phi\Phi^{\mathrm{T}} dx = I \tag{3.74}$$

3.7.5 Inertia Matrix

Now that the integrals Ξ_1 through Ξ_4 have been evaluated, define:

$$\mathcal{A} \triangleq \begin{bmatrix} 0 & \alpha_v & 0 \\ 0 & 0 & \alpha_w \end{bmatrix}, \qquad \Pi \triangleq \mathcal{A}R^{\mathrm{T}}J^v - \Xi_2 J^\Omega \tag{3.75}$$

Then the flexible inertia matrix, D_f, is:

$$\begin{aligned} D_f &\approx \rho \int_0^l J^{\mathrm{T}} J dx \\ &= \rho \begin{bmatrix} \left((J^v)^{\mathrm{T}}J^v l - (J^\Omega)^{\mathrm{T}}\Xi_1^{\mathrm{T}}R^{\mathrm{T}}J^v - (J^v)^{\mathrm{T}}R\Xi_1 J^\Omega + (J^\Omega)^{\mathrm{T}}\Xi_3 J^\Omega\right) & \Pi^{\mathrm{T}} \\ \Pi & I \end{bmatrix} \end{aligned} \tag{3.76}$$

Thus, the kinetic energy contribution of the flexible beam may be written:

$$K_f = \tfrac{1}{2}\dot{\mathbf{Q}}^{\mathrm{T}} D_f \dot{\mathbf{Q}} \tag{3.77}$$

Consider the individual terms in (3.76). The term $\rho(J^v)^{\mathrm{T}}J^v l = m_f(J^v)^{\mathrm{T}}J^v$ represents the purely translational kinetic energy of the flexible beam based on the

motion at the clamped end of the beam. The term $(J^v)^T R\Xi_1 J^\Omega$ and its transpose represent the correction terms due to the fact that the clamped end of the beam was used as a reference rather than the centre of mass. This fact is readily apparent from the relationship between Ξ_1 and the centre of mass of the flexible beam in (3.65). The centre of mass was not used as the reference point in the derivations because it is constantly changing for the flexible beam, while the clamped end is always fixed.

Finally, the term $(J^\Omega)^T \Xi_3 J^\Omega$ is clearly the purely rotational component of the kinetic energy since $\rho\Xi_3$ is the moment of inertia of the flexible beam. The cross-coupling terms of Ξ_2 are naturally in the off-diagonal matrices of the inertia matrix. Note that the inertia matrix is symmetric as expected.

3.8 Potential Energy

The potential energy is also needed to derive the dynamic equations using the Euler-Lagrange approach. Both the gravitational potential energy and the elastic strain energy must be examined. Let:

$$V_{fg} = \text{gravitational potential energy of the flexible link}$$

$$V_{fe} = \text{elastic strain energy of the flexible link}$$

3.8.1 Gravitational Potential Energy

The gravitational potential energy is readily computed using the centre of mass. Using (3.67), the height, h, of the centre of mass, in base frame coordinates, is:

$$h = [0 \quad 0 \quad 1](R\mathbf{p}_{cm} + \mathbf{d})$$

$$\therefore h = [0 \quad 0 \quad \tfrac{1}{l}] \left(R[\tfrac{1}{2}l^2 \quad \nu^{\mathrm{T}}\alpha_v \quad \eta^{\mathrm{T}}\alpha_w]^{\mathrm{T}} + l\mathrm{d} \right) \tag{3.78}$$

The gravitational potential energy is:

$$V_{fg} = m_f g h \tag{3.79}$$

3.8.2 Strain Energy

Define:

$$E = \text{modulus of elasticity of the beam}$$

$$I_v = \text{area moment of inertia in the } v\text{-direction}$$

$$I_w = \text{area moment of inertia in the } w\text{-direction}$$

From standard vibration theory (see (A.4) in Appendix A), the elastic strain energy of the flexible link is:

$$\begin{aligned} V_{fs} &= \tfrac{1}{2}\int_0^l EI_v v''(x,t)^2 dx + \tfrac{1}{2}\int_0^l EI_w w''(x,t)^2 dx \\ &= \tfrac{1}{2}\int_0^l EI_v \nu(t)^{\mathrm{T}} \varphi''(x)\varphi''(x)^{\mathrm{T}} \nu(t) dx + \tfrac{1}{2}\int_0^l EI_v \eta(t)^{\mathrm{T}} \psi''(x)\psi''(x)^{\mathrm{T}} \eta(t) dx \end{aligned}$$

However, assuming orthonormality of the mode shapes and an Euler-Bernoulli beam model, there exists $\lambda_i^v, i = 1, \ldots, n$ such that:

$$\int_0^l \varphi_i''(x)\varphi_j''(x)^{\mathrm{T}} dx = \begin{cases} \lambda_i^v, & i = j \\ 0, & i \neq j \end{cases}$$

This fact is clear from Appendix B by combining (B.16) and (B.20). Note that the λ_i^v are a function of the natural frequencies of the beam. A similar relationship holds in the w-direction, with corresponding constants $\lambda_i^w, i = 1, \ldots, n$. Define the

constant diagonal matrices:

$$\Lambda_v \triangleq \begin{bmatrix} \lambda_1^v & & 0 \\ & \ddots & \\ 0 & & \lambda_n^v \end{bmatrix} \qquad \text{and} \qquad \Lambda_w \triangleq \begin{bmatrix} \lambda_1^w & & 0 \\ & \ddots & \\ 0 & & \lambda_n^w \end{bmatrix}$$

Hence, the elastic strain energy may be written:

$$V_{fs} = \tfrac{1}{2} EI_v \nu^{\mathrm{T}} \Lambda_v \nu + \tfrac{1}{2} EI_w \eta^{\mathrm{T}} \Lambda_w \eta \tag{3.80}$$

3.9 Dynamic Equations

3.9.1 General Equations

Before addressing the detailed expressions for the dynamic equations, it is instructive to examine the Euler-Lagrange formulation in a general form so that the results may be applied to the rigid robot as well. Hence, consider any robot which satisfies the conditions for the Principle of Virtual Work and for which the kinetic and potential energies take the following form:

$$\mathbf{Q} = \text{generalized coordinates of the robot, } \mathbf{Q}(t) = [Q_i(t)] \in \Re^{N \times 1}$$
$$K = \tfrac{1}{2} \dot{\mathbf{Q}}^{\mathrm{T}} D(\mathbf{Q}) \dot{\mathbf{Q}}$$
$$V = V(\mathbf{Q})$$

where $D(\mathbf{Q})$ is a symmetric, positive-definite inertia matrix.

Most conventional robot manipulators satisfy these conditions. Indeed, from the previous section, it is clear that the flexible beam energies also satisfy these

assumptions. Now, let:

$\mathbf{F}_{ext}$ = the net external force acting on the robot end-effector, $\mathbf{F}_{ext} \in C^{3\times 1}$

$J(\mathbf{Q})$ = the Jacobian mapping joint velocities to end-effector velocities

τ = the torque exerted by the robot motors

and define the Lagrangian:

$$L \triangleq K - V$$

The Euler-Lagrange equations stipulate [12], in vector form, that:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{Q}}}\right)^{\mathrm{T}} - \left(\frac{\partial L}{\partial \mathbf{Q}}\right)^{\mathrm{T}} = \tau - J(\mathbf{Q})^{\mathrm{T}}\mathbf{F}_{ext}$$

Examine the second term in the Euler-Lagrange formulation:

$$\frac{\partial L}{\partial \mathbf{Q}} = \frac{\partial K}{\partial \mathbf{Q}} - \frac{\partial V}{\partial \mathbf{Q}}$$

From (3.15), (3.16) and the form of K:

$$\frac{\partial K}{\partial \mathbf{Q}} = \tfrac{1}{2}\frac{\partial}{\partial \mathbf{Q}}\left(\dot{\mathbf{Q}}^{\mathrm{T}}(D\dot{\mathbf{Q}})\right)$$
$$\therefore \frac{\partial K}{\partial \mathbf{Q}} = \tfrac{1}{2}\dot{\mathbf{Q}}^{\mathrm{T}}\frac{\partial D}{\partial \mathbf{Q}}(I_N \otimes \dot{\mathbf{Q}})$$

Similarly, from (3.13), (3.15), (3.18), (3.19) (3.20) and the fact that $D(\mathbf{Q})$ is symmetric and only a function of $\mathbf{Q}$:

$$\frac{\partial L}{\partial \dot{\mathbf{Q}}} = \frac{\partial K}{\partial \dot{\mathbf{Q}}} = \dot{\mathbf{Q}}^{\mathrm{T}}D$$

Thus, from (3.14) and (3.17)

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{Q}}}\right) = \ddot{\mathbf{Q}}^{\mathrm{T}}D + \dot{\mathbf{Q}}^{\mathrm{T}}\frac{\partial D}{\partial \mathbf{Q}}(\dot{\mathbf{Q}} \otimes I_N)$$

For convenience, define:

$$\mathbf{g}(\mathbf{Q}) \triangleq \left(\frac{\partial V}{\partial \mathbf{Q}}\right)^{\mathrm{T}} \qquad \text{and} \qquad D'(\mathbf{Q}) \triangleq \frac{\partial D}{\partial \mathbf{Q}}$$

Then from the symmetry of $D(\mathbf{Q})$ and (3.7), the equations of motion are:

$$D(\mathbf{Q})\ddot{\mathbf{Q}} + [(\dot{\mathbf{Q}}^{\mathrm{T}} \otimes I_N) - \tfrac{1}{2}(I_N \otimes \dot{\mathbf{Q}}^{\mathrm{T}})]D'(\mathbf{Q})^{\mathrm{T}}\dot{\mathbf{Q}} + \mathbf{g}(\mathbf{Q}) = \tau - J(\mathbf{Q})^{\mathrm{T}}\mathbf{F}_{ext}$$

To simplify further, use the transformation of (3.2) and define the $N^2 \times N^2$ constant matrix:

$$Z \triangleq T_{NN} - \tfrac{1}{2} I_{N^2}$$

and let:

$$C(\mathbf{Q}) \triangleq Z D'(\mathbf{Q})^{\mathrm{T}} \tag{3.81}$$

Thus, application of (3.4) yields the general dynamic equations:

$$D(\mathbf{Q})\ddot{\mathbf{Q}} + (I_N \otimes \dot{\mathbf{Q}}^{\mathrm{T}})C(\mathbf{Q})\dot{\mathbf{Q}} + \mathbf{g}(\mathbf{Q}) = \tau - J(\mathbf{Q})^{\mathrm{T}}\mathbf{F}_{ext} \tag{3.82}$$

The matrix $C(\mathbf{Q})$ lies in $\Re^{N^2 \times N}$. Partition $C(\mathbf{Q})$ into submatrices $C_k(\mathbf{Q}) \in \Re^{N \times N}$ such that:

$$C(\mathbf{Q}) = \begin{bmatrix} C_1(\mathbf{Q}) \\ \vdots \\ C_N(\mathbf{Q}) \end{bmatrix}$$

The expression $(I_N \otimes \dot{\mathbf{Q}}^{\mathrm{T}})C(\mathbf{Q})\dot{\mathbf{Q}}$ may be rewritten in the form:

$$(I_N \otimes \dot{\mathbf{Q}}^{\mathrm{T}})C(\mathbf{Q})\dot{\mathbf{Q}} = \begin{bmatrix} \dot{\mathbf{Q}}^{\mathrm{T}} C_1(\mathbf{Q})\dot{\mathbf{Q}} \\ \vdots \\ \dot{\mathbf{Q}}^{\mathrm{T}} C_N(\mathbf{Q})\dot{\mathbf{Q}} \end{bmatrix} \in \Re^{N \times 1}$$

Let d_{ij} denote the (i,j)-th element of the inertia matrix $D(\mathbf{Q})$ and let c_{ijk} represent the (i,j)-th element of $C_k(\mathbf{Q})$. It may be shown that:

$$c_{ijk} = \frac{1}{2}\left\{\frac{\partial d_{kj}}{\partial Q_i} + \frac{\partial d_{ki}}{\partial Q_j} - \frac{\partial d_{ij}}{\partial Q_k}\right\}$$

The terms c_{ijk} are known as the *Christoffel symbols of the first kind* [12]. Hence, $C(\mathbf{Q})$ in (3.81) is actually a matrix of all the Christoffel symbols. Furthermore, since $c_{ijk} = c_{jik}$, each submatrix C_k is symmetric i.e., $C_k = C_k^{\mathrm{T}}$. This fact can also be deduced from (3.81), (3.9), (3.2) and the symmetry of $D(\mathbf{Q})$.

3.9.2 Flexible Dynamics

Since expressions for the kinetic and potential energies have been derived in terms of the mode shapes, φ and ψ, and the generalized coordinates, ν and η, the Euler-Lagrange equations for our particular configuration may now be formulated. Define the Lagrangians:

$$L_b \triangleq K_b - V_b$$
$$L_f \triangleq K_f - V_f$$
$$L \triangleq L_b + L_f$$

where subscript b denotes the base robot components, and subscript f denotes the flexible link components. Note that while K_f is a function of $\mathbf{Q}$ and $\dot{\mathbf{Q}}$, as defined in (3.77), V_f is only a function of $\mathbf{Q}$. Clearly, K_f and V_f satisfy the assumptions of the preceding section. Assume the base robot also satisfies these conditions. Let:

$$N \triangleq n + 2m$$

Then due to the additive nature of the Lagrangians, the dynamic equations for the overall system may be written:

$$\boxed{\begin{aligned} D_b(\mathbf{q})\ddot{\mathbf{q}} + D_f(\mathbf{Q})\ddot{\mathbf{Q}} + (I_N \otimes \dot{\mathbf{q}}^{\mathrm{T}})C_b(\mathbf{q})\dot{\mathbf{q}} + (I_N \otimes \dot{\mathbf{Q}}^{\mathrm{T}})C_f(\mathbf{Q})\dot{\mathbf{Q}} \\ + \mathbf{g}_b(\mathbf{q}) + \mathbf{g}_f(\mathbf{Q}) = \tau - J(\mathbf{Q})^{\mathrm{T}}\mathbf{F}_{ext} \end{aligned}} \tag{3.83}$$

Note that the terms involving only the dynamics of the base robot have been separated from the terms comprising the dynamic contributions of the flexible link. Furthermore, the structure is hierarchical. The base robot terms are a function of only the base robot generalized coordinates, $\mathbf{q}(t) \in \Re^{m\times 1}$. The flexible terms are a function of both the base robot coordinates, $\mathbf{q}$, and the flexible coordinates, ν and η. Also note that the derivatives of the generalized coordinates have been extracted from the centripetal and Coriolis terms, so that the quadratic nature of these terms is clearly evident.

Now consider each of the flexible link components. From (3.79) and (3.80), the total potential energy for the flexible link is:

$$V_f = m_f g h + \tfrac{1}{2} E I_v \nu^{\mathrm{T}} \Lambda_v \nu + \tfrac{1}{2} E I_w \eta^{\mathrm{T}} \Lambda_w \eta$$

Note that h is a function of $\mathbf{Q}$. By (3.15):

$$\frac{\partial V_f}{\partial \mathbf{Q}} = m_f g \frac{\partial h}{\partial \mathbf{Q}} + \tfrac{1}{2} E I_v \frac{\partial}{\partial \mathbf{Q}} \left(\nu^{\mathrm{T}} \Lambda_v \nu\right) + \tfrac{1}{2} E I_w \frac{\partial}{\partial \mathbf{Q}} \left(\eta^{\mathrm{T}} \Lambda_w \eta\right) \tag{3.84}$$

But by (3.21), (3.13), (3.18), (3.19), (3.20) and symmetry of Λ_v:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{Q}}(\nu^{\mathrm{T}} \Lambda_v \nu) &= \left[\tfrac{\partial}{\partial \mathbf{q}}(\nu^{\mathrm{T}} \Lambda_v \nu) \quad \tfrac{\partial}{\partial \nu}(\nu^{\mathrm{T}} \Lambda_v \nu) \quad \tfrac{\partial}{\partial \eta}(\nu^{\mathrm{T}} \Lambda_v \nu)\right] \\ &= \left[0 \quad \tfrac{\partial}{\partial \nu}(\nu^{\mathrm{T}} \Lambda_v \nu) \quad 0\right] \\ &= \left[0 \quad 2\nu^{\mathrm{T}} \Lambda_v \quad 0\right] \end{aligned}$$

Similarly,

$$\frac{\partial \eta^{\mathrm{T}} \Lambda_w \eta}{\partial \mathbf{Q}} = \begin{bmatrix} 0 & 0 & 2\eta^{\mathrm{T}} \Lambda_w \end{bmatrix}$$

Substituting these results back into (3.84) yields:

$$\mathbf{g}_f(\mathbf{Q})^{\mathrm{T}} = \frac{\partial V_f}{\partial \mathbf{Q}} = m_f g \frac{\partial h}{\partial \mathbf{Q}} + E \begin{bmatrix} 0 & \nu^{\mathrm{T}} I_v \Lambda_v & \eta^{\mathrm{T}} I_w \Lambda_w \end{bmatrix} \tag{3.85}$$

Now, by (3.21):

$$\frac{\partial h}{\partial \mathbf{Q}} = \begin{bmatrix} \frac{\partial h}{\partial \mathbf{q}} & \frac{\partial h}{\partial \nu} & \frac{\partial h}{\partial \eta} \end{bmatrix}$$

and by (3.78) and the matrix calculus properties (3.15), (3.16) and (3.22):

$$\frac{\partial h}{\partial \mathbf{q}} = [0 \quad 0 \quad \tfrac{1}{l}] \left(\frac{\partial R}{\partial \mathbf{q}} (I_m \otimes [\tfrac{1}{2} l^2 \quad \nu^{\mathrm{T}} \alpha_v \quad \eta^{\mathrm{T}} \alpha_w])^{\mathrm{T}} + l J^v \right) \tag{3.86}$$

$$\frac{\partial h}{\partial \nu} = [0 \quad 0 \quad \tfrac{1}{l}] R [0 \quad \alpha_v \quad 0]^{\mathrm{T}} \tag{3.87}$$

$$\frac{\partial h}{\partial \eta} = [0 \quad 0 \quad \tfrac{1}{l}] R [0 \quad 0 \quad \alpha_w]^{\mathrm{T}} \tag{3.88}$$

The flexible inertia matrix, D_f has already been defined in (3.76), but for convenience, define its submatrices as:

$$D_{11} = (J^v)^{\mathrm{T}} J^v l - (J^\Omega)^{\mathrm{T}} \Xi_1^{\mathrm{T}} R^{\mathrm{T}} J^v - (J^v)^{\mathrm{T}} R \Xi_1 J^\Omega + (J^\Omega)^{\mathrm{T}} \Xi_3 J^\Omega \tag{3.89}$$

$$D_{21} = \Pi = \mathcal{A} R^{\mathrm{T}} J^v - \Xi_2 J^\Omega \tag{3.90}$$

$$D_{22} = I \tag{3.91}$$

such that:

$$D_f = \rho \begin{bmatrix} D_{11} & D_{21}^{\mathrm{T}} \\ D_{21} & D_{22} \end{bmatrix}$$

By (3.22) and (3.23):

$$\frac{\partial D_f}{\partial \mathbf{Q}} = \rho \begin{bmatrix} \frac{\partial D_{11}}{\partial \mathbf{Q}} & \frac{\partial (D_{21}^{\mathrm{T}})}{\partial \mathbf{Q}} \\ \frac{\partial D_{21}}{\partial \mathbf{Q}} & \frac{\partial D_{22}}{\partial \mathbf{Q}} \end{bmatrix} P_{m,2n,N} \tag{3.92}$$

and by (3.21):

$$\frac{\partial D_{ij}}{\partial \mathbf{Q}} = \rho \left[\frac{\partial D_{ij}}{\partial \mathbf{q}} \quad \frac{\partial D_{ij}}{\partial \boldsymbol{\nu}} \quad \frac{\partial D_{ij}}{\partial \boldsymbol{\eta}} \right], \quad i,j = 1,\ldots,2 \tag{3.93}$$

Thus, consider each of the matrix Jacobians in equations (3.92) and (3.93):

$$\begin{aligned}\frac{\partial D_{11}}{\partial \mathbf{q}} = {} & l\frac{\partial((J^v)^{\mathrm{T}})}{\partial \mathbf{q}}(I_m \otimes J^v) + l(J^v)^{\mathrm{T}}\frac{\partial J^v}{\partial \mathbf{q}} - \frac{\partial((J^\Omega)^{\mathrm{T}})}{\partial \mathbf{q}}(I_m \otimes \Xi_1^{\mathrm{T}} R^{\mathrm{T}} J^v) \\ & - (J^\Omega)^{\mathrm{T}}\Xi_1^{\mathrm{T}}\frac{\partial(R^{\mathrm{T}})}{\partial \mathbf{q}}(I_m \otimes J^v) - (J^\Omega)^{\mathrm{T}}\Xi_1^{\mathrm{T}} R^{\mathrm{T}}\frac{\partial J^v}{\partial \mathbf{q}} \\ & - \frac{\partial((J^v)^{\mathrm{T}})}{\partial \mathbf{q}}(I_m \otimes R\Xi_1 J^\Omega) - (J^v)^{\mathrm{T}}\frac{\partial R}{\partial \mathbf{q}}(I_m \otimes \Xi_1 J^\Omega) - (J^v)^{\mathrm{T}} R\Xi_1\frac{\partial J^\Omega}{\partial \mathbf{q}} \\ & + \frac{\partial((J^\Omega)^{\mathrm{T}})}{\partial \mathbf{q}}(I_m \otimes \Xi_3 J^\Omega) + (J^\Omega)^{\mathrm{T}}\Xi_3\frac{\partial J^\Omega}{\partial \mathbf{q}}\end{aligned} \tag{3.94}$$

$$\begin{aligned}\frac{\partial D_{11}}{\partial \boldsymbol{\nu}} = {} & -(J^\Omega)^{\mathrm{T}}\frac{\partial(\Xi_1^{\mathrm{T}})}{\partial \boldsymbol{\nu}}(I_n \otimes R^{\mathrm{T}} J^v) - (J^v)^{\mathrm{T}} R\frac{\partial \Xi_1}{\partial \boldsymbol{\nu}}(I_n \otimes J^\Omega) \\ & + (J^\Omega)^{\mathrm{T}}\frac{\partial \Xi_3}{\partial \boldsymbol{\nu}}(I_n \otimes J^\Omega)\end{aligned} \tag{3.95}$$

$$\begin{aligned}\frac{\partial D_{11}}{\partial \boldsymbol{\eta}} = {} & -(J^\Omega)^{\mathrm{T}}\frac{\partial(\Xi_1^{\mathrm{T}})}{\partial \boldsymbol{\eta}}(I_n \otimes R^{\mathrm{T}} J^v) - (J^v)^{\mathrm{T}} R\frac{\partial \Xi_1}{\partial \boldsymbol{\eta}}(I_n \otimes J^\Omega) \\ & + (J^\Omega)^{\mathrm{T}}\frac{\partial \Xi_3}{\partial \boldsymbol{\eta}}(I_n \otimes J^\Omega)\end{aligned} \tag{3.96}$$

$$\frac{\partial D_{21}}{\partial \mathbf{q}} = \mathcal{A}\left(\frac{\partial(R^{\mathrm{T}})}{\partial \mathbf{q}}(I_m \otimes J^v) + R^{\mathrm{T}}\frac{\partial J^v}{\partial \mathbf{q}}\right) - \Xi_2\frac{\partial J^\Omega}{\partial \mathbf{q}} \tag{3.97}$$

$$\frac{\partial D_{21}}{\partial \boldsymbol{\nu}} = -\frac{\partial \Xi_2}{\partial \boldsymbol{\nu}}(I_n \otimes J^\Omega) \tag{3.98}$$

$$\frac{\partial D_{21}}{\partial \boldsymbol{\eta}} = -\frac{\partial \Xi_2}{\partial \boldsymbol{\eta}}(I_n \otimes J^\Omega) \tag{3.99}$$

$$\frac{\partial(D_{21}^{\mathrm{T}})}{\partial \mathbf{q}} = \frac{\partial((J^v)^{\mathrm{T}})}{\partial \mathbf{q}}(I_m \otimes R\mathcal{A}) + (J^v)^{\mathrm{T}}\frac{\partial R}{\partial \mathbf{q}}(I_m \otimes \mathcal{A}^{\mathrm{T}}) - \frac{\partial J^\Omega}{\partial \mathbf{q}}(I_m \otimes \Xi_2^{\mathrm{T}}) \tag{3.100}$$

$$\frac{\partial(D_{21}^{\mathrm{T}})}{\partial \boldsymbol{\nu}} = -(J^\Omega)^{\mathrm{T}}\frac{\partial(\Xi_2^{\mathrm{T}})}{\partial \boldsymbol{\nu}} \tag{3.101}$$

$$\frac{\partial(D_{21}^{\mathrm{T}})}{\partial \boldsymbol{\eta}} = -(J^\Omega)^{\mathrm{T}}\frac{\partial(\Xi_2^{\mathrm{T}})}{\partial \boldsymbol{\eta}} \tag{3.102}$$

$$\frac{\partial D_{22}}{\partial \mathbf{Q}} = 0 \tag{3.103}$$

Applying the matrix calculus properties, and the partitioning equations (3.22), (3.23) and (3.11), the partial derivatives of the flexible terms $\Xi_i, i = 1, \ldots, 3$ are:

$$\frac{\partial \Xi_1}{\partial \nu} = \begin{bmatrix} 0 & 0 & \alpha_v^T \\ 0 & 0 & 0 \\ -\alpha_v^T & 0 & 0 \end{bmatrix} T_{3,n} \tag{3.104}$$

$$\frac{\partial \Xi_1}{\partial \eta} = \begin{bmatrix} 0 & -\alpha_w^T & 0 \\ \alpha_w^T & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} T_{3,n} \tag{3.105}$$

$$\frac{\partial ((\Xi_1)^T)}{\partial \nu} = \begin{bmatrix} 0 & 0 & -\alpha_v^T \\ 0 & 0 & 0 \\ \alpha_v^T & 0 & 0 \end{bmatrix} T_{3,n} \tag{3.106}$$

$$\frac{\partial ((\Xi_1)^T)}{\partial \eta} = \begin{bmatrix} 0 & \alpha_w^T & 0 \\ -\alpha_w^T & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} T_{3,n} \tag{3.107}$$

$$\frac{\partial \Xi_2}{\partial \nu} = \begin{bmatrix} 0 & 0 & 0 \\ -\gamma^T & 0 & 0 \end{bmatrix} T_{3,n} \tag{3.108}$$

$$\frac{\partial \Xi_2}{\partial \eta} = \begin{bmatrix} \gamma & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} T_{3,n} \tag{3.109}$$

$$\frac{\partial ((\Xi_2)^T)}{\partial \nu} = \begin{bmatrix} 0 & -\operatorname{vec}(\gamma^T)^T \\ 0 & 0 \\ 0 & 0 \end{bmatrix} P_{n,n,n} \tag{3.110}$$

$$\frac{\partial ((\Xi_2)^T)}{\partial \eta} = \begin{bmatrix} \operatorname{vec}(\gamma)^T & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} P_{n,n,n} \tag{3.111}$$

$$\frac{\partial \Xi_3}{\partial \nu} = \begin{bmatrix} 2\nu^{\mathrm{T}} & -\beta_v^{\mathrm{T}} & 0 \\ -\beta_v^{\mathrm{T}} & 0 & -\eta^{\mathrm{T}}\gamma^{\mathrm{T}} \\ 0 & -\eta^{\mathrm{T}}\gamma^{\mathrm{T}} & 2\nu^{\mathrm{T}} \end{bmatrix} T_{3,n} \tag{3.112}$$

$$\frac{\partial \Xi_3}{\partial \eta} = \begin{bmatrix} 2\eta^{\mathrm{T}} & 0 & -\beta_w^{\mathrm{T}} \\ 0 & 2\eta^{\mathrm{T}} & -\nu^{\mathrm{T}}\gamma \\ -\beta_w^{\mathrm{T}} & -\nu^{\mathrm{T}}\gamma & 0 \end{bmatrix} T_{3,n} \tag{3.113}$$

3.9.3 State-Space Representation

Consider the dynamic equations in section 3.9.1. To put these general dynamic equations into state-space form, let:

$$\begin{aligned} \mathbf{X}_1 &\triangleq \mathbf{Q} \\ \mathbf{X}_2 &\triangleq \dot{\mathbf{Q}} \\ \mathbf{X} &\triangleq [\mathbf{X}_1^{\mathrm{T}} \quad \mathbf{X}_2^{\mathrm{T}}]^{\mathrm{T}} \\ \mathbf{U} &\triangleq [\tau^{\mathrm{T}} \quad \mathbf{F}_{ext}^{\mathrm{T}}]^{\mathrm{T}} \end{aligned}$$

Thus, from equation (3.82):

$$D(\mathbf{X}_1)\dot{\mathbf{X}}_2 + (I_N \otimes \mathbf{X}_2^{\mathrm{T}})C(\mathbf{X}_1)\mathbf{X}_2 + \mathbf{g}(\mathbf{X}_1) = [I_N \quad -J(\mathbf{X}_1)^{\mathrm{T}}]\mathbf{U}$$

or in state-space form:

$$\begin{bmatrix} \dot{\mathbf{X}}_1 \\ \dot{\mathbf{X}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{X}_2 \\ -D(\mathbf{X}_1)^{-1}\left((I_N \otimes \mathbf{X}_2^{\mathrm{T}})C(\mathbf{X}_1)\mathbf{X}_2 + \mathbf{g}(\mathbf{X}_1)\right) \end{bmatrix} + \begin{bmatrix} 0 \\ D(\mathbf{X}_1)^{-1}[I_N \quad -J(\mathbf{X}_1)^{\mathrm{T}}] \end{bmatrix} \mathbf{U} \tag{3.114}$$

3.9.4 Linearization

Once the non-linear state-space equations are defined, it is a simple matter to derive a linearized state-space representation. Indeed, from (3.114), it is clear that the general dynamic equations are already linear in $\mathbf{U}$. For convenience, let:

$$\mathbf{P}(\mathbf{X}) \triangleq (I_N \otimes \mathbf{X}_2^{\mathrm{T}})C(\mathbf{X}_1)\mathbf{X}_2 + \mathbf{g}(\mathbf{X}_1)$$

and define:

$$\mathbf{F}(\mathbf{X},\mathbf{U}) = \begin{bmatrix} \mathbf{F}_1(\mathbf{X},\mathbf{U}) \\ \mathbf{F}_2(\mathbf{X},\mathbf{U}) \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{X}_2 \\ -D(\mathbf{X}_1)^{-1}\mathbf{P}(\mathbf{X}) + D(\mathbf{X}_1)^{-1}[I_N \quad -J(\mathbf{X}_1)^{\mathrm{T}}]\mathbf{U} \end{bmatrix}$$

To linearize (3.114) about an operating point, $(\mathbf{X}^0, \mathbf{U}^0)$, compute constant matrices A and B:

$$A \triangleq \left.\frac{\partial \mathbf{F}}{\partial \mathbf{X}}\right|_{(\mathbf{X}^0,\mathbf{U}^0)} \qquad \text{and} \qquad B \triangleq \left.\frac{\partial \mathbf{F}}{\partial \mathbf{U}}\right|_{(\mathbf{X}^0,\mathbf{U}^0)}$$

and define the incremental signals:

$$\Delta\mathbf{X} \triangleq \mathbf{X} - \mathbf{X}^0 \qquad \text{and} \qquad \Delta\mathbf{U} \triangleq \mathbf{U} - \mathbf{U}^0$$

so that:

$$\Delta\dot{\mathbf{X}} \approx A\Delta\mathbf{X} + B\Delta\mathbf{U}$$

Now from (3.21) and (3.22):

$$\frac{\partial \mathbf{F}}{\partial \mathbf{X}} = \begin{bmatrix} \frac{\partial \mathbf{F}_1}{\partial \mathbf{X}_1} & \frac{\partial \mathbf{F}_1}{\partial \mathbf{X}_2} \\ \frac{\partial \mathbf{F}_2}{\partial \mathbf{X}_1} & \frac{\partial \mathbf{F}_2}{\partial \mathbf{X}_2} \end{bmatrix} \qquad \text{and} \qquad \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \begin{bmatrix} \frac{\partial \mathbf{F}_1}{\partial \mathbf{U}} \\ \frac{\partial \mathbf{F}_2}{\partial \mathbf{U}} \end{bmatrix}$$

Consider each of the partial derivatives in turn:

$$\frac{\partial \mathbf{F}_1}{\partial \mathbf{X}_1} = 0$$

$$\frac{\partial \mathbf{F}_1}{\partial \mathbf{X}_2} = I_N$$

$$\frac{\partial \mathbf{F}_2}{\partial \mathbf{X}_1} = -\frac{\partial (D(\mathbf{X}_1)^{-1})}{\partial \mathbf{X}_1}(I_N \otimes \mathbf{P}(\mathbf{X})) - D(\mathbf{X}_1)^{-1}\frac{\partial \mathbf{P}}{\partial \mathbf{X}_1} + \frac{\partial (D(\mathbf{X}_1)^{-1})}{\partial \mathbf{X}_1}\left(I_N \otimes [I_N \quad -J(\mathbf{X}_1)^{\mathrm{T}}]\mathbf{U}\right) + D(\mathbf{X}_1)^{-1}\begin{bmatrix} 0 & -\frac{\partial (J(\mathbf{X}_1)^{\mathrm{T}})}{\partial \mathbf{X}_1} \end{bmatrix} P_{N,3,N}(I_N \otimes \mathbf{U})$$

$$\frac{\partial \mathbf{F}_2}{\partial \mathbf{X}_2} = -D(\mathbf{X}_1)^{-1}\frac{\partial \mathbf{P}}{\partial \mathbf{X}_2}$$

where the partial derivatives of $\mathbf{P}(\mathbf{X})$ are:

$$\frac{\partial \mathbf{P}}{\partial \mathbf{X}_1} = (I_N \otimes \mathbf{X}_2^{\mathrm{T}})\frac{\partial C}{\partial \mathbf{X}_1}(I_N \otimes \mathbf{X}_2) + \frac{\partial \mathbf{g}}{\partial \mathbf{X}_1}$$

$$\frac{\partial \mathbf{P}}{\partial \mathbf{X}_2} = \left(I_N \otimes \mathrm{vec}(I_N)^{\mathrm{T}}\right) \underbrace{P_{N,\ldots,N,N}}_{N} C(\mathbf{X}_1)\mathbf{X}_2 + \left(I_N \otimes \mathbf{X}_2^{\mathrm{T}}\right) C(\mathbf{X}_1)$$

Now, typically the operating point is chosen to be stationary with zero input, i.e.,

$$\mathbf{X}_2^0 = 0$$

$$\mathbf{U}^0 = 0$$

Substituting this partial operating point into the partial derivatives yields the state matrices:

$$A = \begin{bmatrix} 0 & I_N \\ \frac{\partial (D(\mathbf{X}_1)^{-1})}{\partial \mathbf{X}_1}(I_N \otimes \mathbf{g}(\mathbf{X}_1)) - D(\mathbf{X}_1)^{-1}\frac{\partial \mathbf{g}}{\partial \mathbf{X}_1} & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ D(\mathbf{X}_1)^{-1}[I_N \quad -J(\mathbf{X}_1)^{\mathrm{T}}] \end{bmatrix}$$

Note that for any system for which $\mathbf{g}(\mathbf{X}_1) \equiv 0$, such as a gravity-balanced rigid manipulator, the A matrix reduces to a constant matrix independent of $\mathbf{X}_1^0$:

$$A = \begin{bmatrix} 0 & I_N \\ 0 & 0 \end{bmatrix}$$

While the A matrix will not be independent of $\mathbf{X}_1^0$ once damping is introduced, it is straightforward to add a damping term $B(\mathbf{X}_1)\mathbf{X}_2$ to the model.

Similarly, for any system with a constant inertia matrix, $D(\mathbf{X}_1) = M$, such as the single flexible link example of section 3.3, the A matrix reduces to:

$$A = \begin{bmatrix} 0 & I_N \\ -M^{-1}\dfrac{\partial \mathbf{g}}{\partial \mathbf{X}_1} & 0 \end{bmatrix}$$

The general model derived here is a full matrix representation of the Euler-Lagrange dynamic equations. Hence, it is readily converted to state-space form and coupling terms, etc. are easily discerned, and the quadratic nature of the centripetal and Coriolis terms is clearer. Indeed, the Christoffel symbols are isolated into a single matrix, $C(\mathbf{Q})$ and a simple expression for $C(\mathbf{Q})$ in terms of the matrix Jacobian of the inertia matrix, $D(\mathbf{Q})$, is presented. The model may be applied to a rigid (or flexible) manipulator picking up a flexible payload. Separation of the flexible terms aids in understanding the impact of flexibility on the dynamic equations. In particular, it may be beneficial for understanding the effect of flexible payloads on the dynamics of the manipulator. The model is hierarchical and may be applied recursively to consider manipulators with multiple flexible links.

3.10 Rigid Links

The equations of motion were derived based on a flexible link. However, the same derivation could equally well be applied to a rigid link. Indeed, the equations are greatly simplified because in the case of a rigid link:

$n = 0$	i.e., with no modes, matrix dimensions are reduced
$v = w = 0$	i.e. there is no link deflection

As a result of these simplifications, the equations in the derivation are modified as follows:

- in (3.32), $\mathbf{p} = [x \;\; 0 \;\; 0]^{\mathrm{T}}$
- in (3.43), $\mathbf{p}_0 = R[x \;\; 0 \;\; 0]^{\mathrm{T}} + \mathbf{d}$
- in (3.47), $R_0^p = R$
- in (3.56), $J = J^v - R\,\mathrm{S}(\mathbf{p})J^\Omega \in \Re^{3\times m}$
- in (3.57), $\dot{\mathbf{p}}_0 = J\dot{\mathbf{q}}$
- in (3.66), $\Xi_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{2}l^2 \\ 0 & \frac{1}{2}l^2 & 0 \end{bmatrix}$
- in (3.67), $\mathbf{p}_{cm} = [\frac{1}{2}l \;\; 0 \;\; 0]^{\mathrm{T}}$
- in (3.72), no longer relevant
- in (3.73), $\Xi_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{3}l^3 & 0 \\ 0 & 0 & \frac{1}{3}l^3 \end{bmatrix}$
- in (3.74), no longer relevant
- in (3.76), $D_f = \rho\left((J^v)^{\mathrm{T}}J^v l - (J^\Omega)^{\mathrm{T}}\Xi_1^{\mathrm{T}}R^{\mathrm{T}}J^v - (J^v)^{\mathrm{T}}R\Xi_1 J^\Omega + (J^\Omega)^{\mathrm{T}}\Xi_3 J^\Omega\right)$
- in (3.78), $h = [0 \;\; 0 \;\; 1]\left(R[\frac{1}{2}l \;\; 0 \;\; 0]^{\mathrm{T}} + \mathbf{d}\right)$
- in (3.80), $V_{fs} = 0$ since there is no strain
- in (3.85), $\mathbf{g}_f(\mathbf{q})^{\mathrm{T}} = m_f g \dfrac{\partial h}{\partial \mathbf{q}}$
- in (3.86), $\dfrac{\partial h}{\partial \mathbf{q}} = [0 \;\; 0 \;\; 1]\left(\dfrac{\partial R}{\partial \mathbf{q}}(I_m \otimes [\frac{1}{2}l \;\; 0])^{\mathrm{T}} + J^v\right)$

3.11 Force Sensing

One objective in using flexible structures in robotics is to employ the robot structure itself as a force sensor. Since strain gauges are inexpensive yet fairly accurate, such an approach has evident cost advantages. To use a flexible link as a force sensor, consider the Euler-Bernoulli beam model. From standard beam theory (see (B.2) and (B.4)), the shear force acting on the beam due to bending, in the v-direction, is:

$$\begin{aligned} F_v(x,t) &= -EI_v v'''(x,t) \\ &= -(EI_v h^{-1})\epsilon_v'(x,t) \quad \text{from (C.1)} \end{aligned}$$

While the derivative of strain is not measured directly, it can be approximated from strain measurements using a first-order Euler difference, whence for two strain gauge positions, x_1 and x_2:

$$F_v(x_2,t) \approx -(EI_v h^{-1}) \left(\frac{\epsilon_v(x_2,t) - \epsilon_v(x_1,t)}{x_2 - x_1} \right)$$

For small deflections, a crude approximation is obtained by considering $x_1 = 0$ and $x_2 = l$. For a clamped-free beam, the strain at the tip of the beam is always zero, so this approximation yields:

$$F_v(l,t) \approx \left(\frac{EI_v}{hl} \right) \epsilon_v(0,t) \tag{3.115}$$

Hence, the shear force on the beam is approximately proportional to the strain measured at the clamped end. In practice, as seen in section 5.2, this approximation ends up being quite effective. Note that a more accurate result may be obtained by using multiple strain gauges. The reader is referred to [80] for such a case.

3.12 Single Flexible Link Revisited

In order to demonstrate the validity of the derived equations, the general model will be applied to the traditional single-flexible link experiment depicted in Figure 3.1. The apparatus consists of a flexible beam attached rigidly to a motor shaft at one end, and free to vibrate at the other end. The homogeneous transformation relating the base frame to frame e consists of the rotation:

$$R = \begin{bmatrix} \cos(q_0) & -\sin(q_0) & 0 \\ \sin(q_0) & \cos(q_0) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and a translation:

$$\mathbf{d} = [0 \quad 0 \quad 0]^{\mathrm{T}}$$

There is only one base generalized coordinate, q_0. Hence, $m = 1$. It is easily seen that:

$$J^{v} = [0 \quad 0 \quad 0]^{\mathrm{T}} \tag{3.116}$$

$$J^{\Omega} = [0 \quad 0 \quad 1]^{\mathrm{T}} \tag{3.117}$$

Substituting these equations into (3.56):

$$J = \begin{bmatrix} -x\sin(q_0) - \varphi^{\mathrm{T}}\nu\cos(q_0) & -\sin(q_0)\varphi^{\mathrm{T}} & 0 \\ x\cos(q_0) - \varphi^{\mathrm{T}}\nu\sin(q_0) & \cos(q_0)\varphi^{\mathrm{T}} & 0 \\ 0 & 0 & \psi^{\mathrm{T}} \end{bmatrix} \in \Re^{3\times(1+2n)}$$

This Jacobian provides both the velocity kinematics and the relationship between the motor torque and the resulting force anywhere along the length of the beam.

Now consider the flexible inertia matrix, D_f. Substituting (3.116) into (3.89) - (3.91):

$$D_{11} = (J^{\Omega})^{\mathrm{T}} \Xi_3 J^{\Omega}$$
$$D_{21} = -\Xi_2 J^{\Omega}$$
$$D_{22} = I$$

Note that D_{11} contains only the purely flexible term Ξ_3 because there is no translation of the clamped end of the beam - it is attached to the hub. Substituting (3.117) into these terms:

$$D_{11} = \tfrac{1}{3} l^3 + \boldsymbol{\nu}^{\mathrm{T}} \boldsymbol{\nu}$$
$$D_{21} = [\boldsymbol{\beta}_v^{\mathrm{T}} \quad 0]^{\mathrm{T}}$$

Hence, the flexible inertia matrix has the very simple form:

$$D_f = \rho \begin{bmatrix} \frac{1}{3} l^3 + \boldsymbol{\nu}^{\mathrm{T}} \boldsymbol{\nu} & \boldsymbol{\beta}_v^{\mathrm{T}} & 0 \\ \boldsymbol{\beta}_v & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

Knowledge of the flexible inertia matrix allows the kinetic energy in the system due to the flexible beam to be calculated via equation (3.77). The inertia matrix is also present in the final dynamic equations. Now, from the above Jacobians:

$$\frac{\partial J^v}{\partial \mathbf{q}} = [0 \quad 0 \quad 0]^{\mathrm{T}}$$
$$\frac{\partial J^{\Omega}}{\partial \mathbf{q}} = [0 \quad 0 \quad 0]^{\mathrm{T}}$$

Thus, from (3.94)-(3.103), the partial derivatives of the inertia matrix elements are:

$$\frac{\partial D_{11}}{\partial \mathbf{q}} = 0$$

$$\frac{\partial D_{11}}{\partial \boldsymbol{\nu}} = (J^{\Omega})^{\mathrm{T}} \frac{\partial \Xi_3}{\partial \boldsymbol{\nu}} (I_n \otimes J^{\Omega})$$
$$\frac{\partial D_{11}}{\partial \boldsymbol{\eta}} = (J^{\Omega})^{\mathrm{T}} \frac{\partial \Xi_3}{\partial \boldsymbol{\eta}} (I_n \otimes J^{\Omega})$$
$$\frac{\partial D_{21}}{\partial \mathbf{q}} = 0$$
$$\frac{\partial D_{21}}{\partial \boldsymbol{\nu}} = -\frac{\partial \Xi_2}{\partial \boldsymbol{\nu}} (I_n \otimes J^{\Omega})$$
$$\frac{\partial D_{21}}{\partial \boldsymbol{\eta}} = -\frac{\partial \Xi_2}{\partial \boldsymbol{\eta}} (I_n \otimes J^{\Omega})$$
$$\frac{\partial D_{22}}{\partial \mathbf{Q}} = 0$$

Note that:

$$T_{3,n}(I_n \otimes J^{\Omega}) = [0 \quad 0 \quad I_n]^{\mathrm{T}}$$

Hence, substituting (3.117) into these equations, along with the partial derivatives of Ξ_2 and Ξ_3 yields:

$$\frac{\partial D_{11}}{\partial \boldsymbol{\nu}} = 2\boldsymbol{\nu}^{\mathrm{T}}$$
$$\frac{\partial D_{11}}{\partial \boldsymbol{\eta}} = 0$$
$$\frac{\partial D_{21}}{\partial \boldsymbol{\nu}} = 0$$
$$\frac{\partial D_{21}}{\partial \boldsymbol{\eta}} = 0$$

whence:

$$\frac{\partial D}{\partial \mathbf{Q}} = \left[\begin{array}{ccc|ccc} 0 & 2\boldsymbol{\nu}^{\mathrm{T}} & 0 & 0 & \cdots & 0 \\ \hline \underbrace{0}_{1} & \underbrace{0}_{n} & \underbrace{0}_{n} & \underbrace{0}_{2n} & \cdots & \underbrace{0}_{2n} \end{array}\right] P_{1,2n,1+2n}$$

Evaluating the centripetal and Coriolis terms using this expression and (3.81) leads to:

$$(I_{1+2n} \otimes \dot{\mathbf{Q}}^{\mathrm{T}}) C(\mathbf{Q}) \dot{\mathbf{Q}} = \rho \begin{bmatrix} 2\boldsymbol{\nu}^{\mathrm{T}} \dot{\boldsymbol{\nu}} \dot{q}_0 \\ -\boldsymbol{\nu} \dot{q}_0^2 \\ 0 \end{bmatrix}$$

Also, looking at the rotation, R and equations (3.86)-(3.88):

$$\begin{aligned}\frac{\partial h}{\partial \mathbf{q}} &= 0\\ \frac{\partial h}{\partial \boldsymbol{\nu}} &= 0\\ \frac{\partial h}{\partial \boldsymbol{\eta}} &= \tfrac{1}{l}\boldsymbol{\alpha}_w^{\mathrm{T}}\end{aligned}$$

Let I_h be the hub inertia of the motor. Then substituting the above results into the general dynamic equation (3.83) and (3.85):

$$\begin{bmatrix} I_h + \frac{1}{3}m_f l^2 + \rho\boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{\nu} & \rho\boldsymbol{\beta}_v^{\mathrm{T}} & 0 \\ \rho\boldsymbol{\beta}_v & \rho I & 0 \\ 0 & 0 & \rho I \end{bmatrix} \begin{bmatrix} \ddot{q}_0 \\ \ddot{\boldsymbol{\nu}} \\ \ddot{\boldsymbol{\eta}} \end{bmatrix} + \begin{bmatrix} 0 & & \\ & EI_v\Lambda_v & \\ & & EI_w\Lambda_w \end{bmatrix} \begin{bmatrix} q_0 \\ \boldsymbol{\nu} \\ \boldsymbol{\eta} \end{bmatrix}$$
$$+ \begin{bmatrix} 2\rho\boldsymbol{\nu}^{\mathrm{T}}\dot{\boldsymbol{\nu}}\dot{q}_0 \\ -\rho\boldsymbol{\nu}\dot{q}_0^2 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \rho g\boldsymbol{\alpha}_w \end{bmatrix} = \begin{bmatrix} \tau_h \\ 0 \\ 0 \end{bmatrix} - J(\mathbf{Q})^{\mathrm{T}}\mathbf{F}_{ext}$$

Comparison with the single flexible link example in section 3.3 verifies that the general dynamic equations have yielded the correct result[3]. Note that the extra rows corresponding to $\boldsymbol{\eta}$ are present because deflections in the vertical direction are included in this more general model. The term $\frac{1}{3}m_f l^2$ represents the rotary inertia of an equivalent rigid beam. Hence, the extra factor $\rho\boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{\nu}$ accounts for the deflection of the flexible beam in the horizontal plane. It is also clear from the off-diagonal elements of the inertia matrix that there will be coupling between the hub angle and horizontal vibrations, as expected[4].

The model predicts that at equilibrium (presuming $\mathbf{F}_{ext} = \mathbf{0}$), there will be a

[3] $\boldsymbol{\nu}$ here is equivalent to $[q_1 \ \cdots \ q_n]$ earlier

[4] There is no interaction with the vertical modes, $\boldsymbol{\eta}$, because axial extension and compression of the beam has not been modelled

sag in the beam due to gravity such that:

$$\lim_{t\to\infty} \eta(t) = -\rho g (EI_w \Lambda_w)^{-1} \alpha_w$$

3.13 Five-Bar Linkage Robot

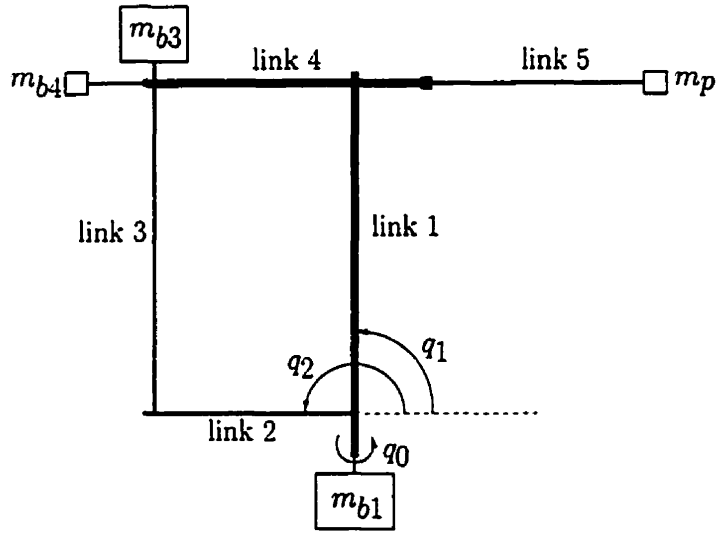


Figure 3.5: Five-Bar Linkage Robot

The model derived in this chapter applies to a flexible link attached to an arbitrary robot. The experimental setup used for most of the experiments in this research comprises a five-bar linkage robot with all links rigid except the last link, which is flexible. A simplified schematic of the robot is depicted in Figure 3.5. The first four links are rigid and the fifth link represents the flexible link. There are three adjustable counterbalances whose purposes are described in section 4.2. The "payload" mass m_p actually represents a strain gauge unit used to measure axial forces.

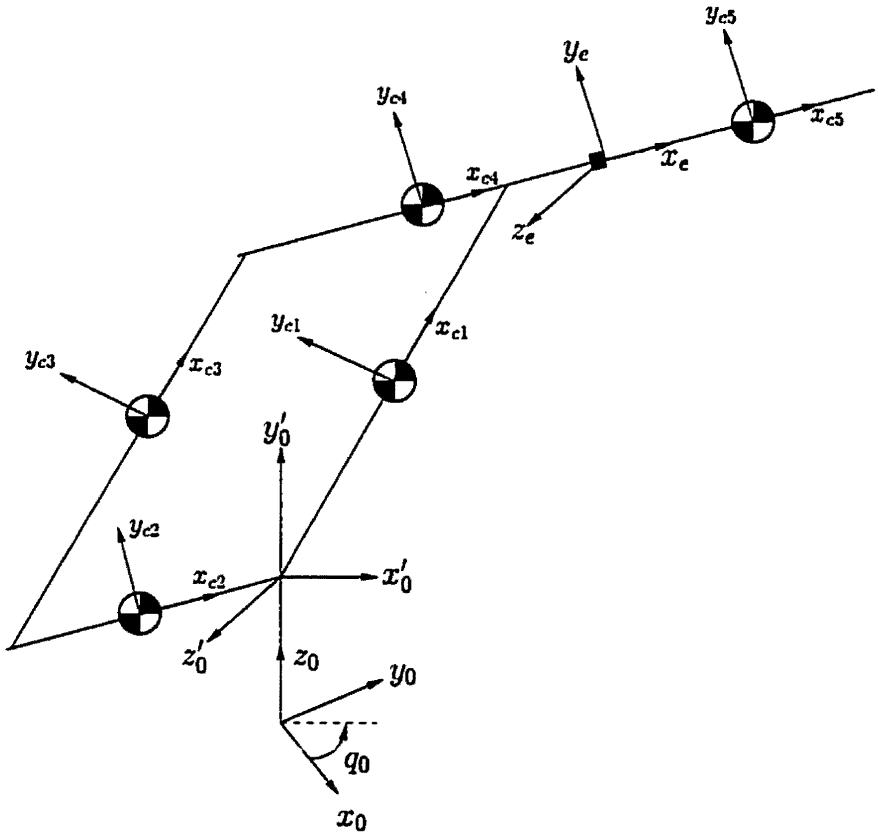


Figure 3.6: Reference Frames for the Five-Bar Robot

In order to apply the modelling results of this chapter to such an apparatus, the homogeneous transformation, $H_0^e(\mathbf{q}, t)$, mapping the base frame to the flexible frame must be determined. For the purposes of this derivation, the Denavit-Hartenberg (DH) convention [12] will be followed. Note that the counterbalances will be treated as constituting part of the associated link. Figure 3.6 illustrates the various reference frames involved in the analysis. Each link has a centre of mass with an associated reference frame x_c-y_c-z_c. The base frame is x_0-y_0-z_0 and the "end-effector frame" for the purposes of this section is x_e-y_e-z_e. The reference frame for each link is considered to have the same orientation as the frame at the centre of mass, but having its origin at the end of the link.

Since the frames are chosen using the DH convention, it is possible to define the homogeneous transformation mapping the reference frame of one link to the reference frame of the next link in the kinematic chain in terms of two rotations and two translations. Consider the transformation from link i to link $i+1$. Rotate frame i about its z-axis through an angle θ such that the x_i and x_{i+1} axes are parallel. Then translate a distance d along the z-axis to make the x_i and x_{i+1} axes collinear. Next, translate along the resulting x-axis to make the origins coincident. Finally, rotate about the x-axis through an angle ϕ to make the frames identical. This series of four operations is equivalent to the homogeneous transformation:

$$A(\theta, d, a, \alpha) = \begin{bmatrix} \cos(\theta) & -\sin(\theta)\cos(\alpha) & \sin(\theta)\sin(\alpha) & a\cos(\theta) \\ \sin(\theta) & \cos(\theta)\cos(\alpha) & -\cos(\theta)\sin(\alpha) & a\sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The four transformation parameters for the mapping from each frame to the next frame are listed in Table 3.1. The mapping from frame e to the centre of mass of link 5 assumes that link 5 is rigid. It is included for the purposes of section 4.2.

Mapping	θ	d	a	ϕ
$0 \rightarrow 0'$	q_0	0	0	$\frac{\pi}{2}$
$0' \rightarrow c2$	$q_2 - \pi$	0	$-l_{c2}$	0
$0' \rightarrow 2$	$q_2 - \pi$	0	$-l_2$	0
$2 \rightarrow c3$	$\pi - (q_2 - q_1)$	0	l_{c3}	0
$2 \rightarrow 3$	$\pi - (q_2 - q_1)$	0	l_1	0
$0' \rightarrow c1$	q_1	0	l_{c1}	0
$0' \rightarrow 1$	q_1	0	l_1	0
$1 \rightarrow c4$	$(q_2 - q_1) - \pi$	0	l_{c4}	0
$1 \rightarrow e$	$(q_2 - q_1) - \pi$	0	l_4	0
$e \rightarrow c5$	0	0	l_{c5}	0

Table 3.1: Homogeneous Transformation Parameters

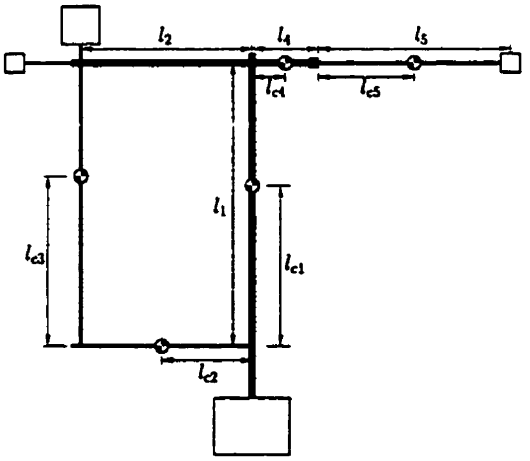


Figure 3.7: Link Lengths for the Five-Bar Robot

The distances to each centre of mass and the link lengths cited in the table are depicted in Figure 3.7.

The mapping from one coordinate frame to any other can be determined simply by multiplying the coordinate transformations together in the appropriate order. For example, using C_i to denote $\cos(q_i)$ and S_i to represent $\sin(q_i)$, the homogeneous transformation from the base frame to frame e is simply:

$$\begin{aligned} H_0^e &= H_0^{0'} H_{0'}^1 H_1^e \\ &= A(q_0, 0, 0, \tfrac{\pi}{2}) A(q_1, 0, l_1, 0) A(q_2 - q_1 - \pi, 0, l_4, 0) \\ &= \left[\begin{array}{ccc|c} -C_0C_2 & C_0S_2 & S_0 & (l_1C_1 - l_4C_2)C_0 \\ -S_0C_2 & S_0S_2 & -C_0 & (l_1C_1 - l_4C_2)S_0 \\ -S_2 & -C_2 & 0 & l_1S_1 - l_4S_2 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \qquad (3.118) \\ &\equiv \begin{bmatrix} R & \mathbf{d} \\ 0 & 1 \end{bmatrix} \qquad (3.119) \end{aligned}$$

Hence, using (3.42), (3.52) and (3.53):

$$J^v = \begin{bmatrix} -(l_1C_1 - l_4C_2)S_0 & -l_1S_1C_0 & l_4S_2C_0 \\ (l_1C_1 - l_4C_2)C_0 & -l_1S_1S_0 & l_4S_2S_0 \\ 0 & l_1C_1 & -l_4C_2 \end{bmatrix} \qquad (3.120)$$

$$J^\Omega = \begin{bmatrix} -S_2 & 0 & 0 \\ -C_2 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (3.121)$$

Let:

$$l_x \triangleq l_4 + x \qquad (3.122)$$

Substituting J^v, J^Ω and l_x from (3.120) - (3.122) into (3.56) yields:

$$J(\mathbf{Q},x) = \begin{bmatrix} -(l_1C_1 - l_xC_2 + vS_2)S_0 + wC_0 & -l_1S_1C_0 & (vC_2 + l_xS_2)C_0 & \varphi^{\mathrm{T}}S_2C_0 & \psi^{\mathrm{T}}S_0 \\ (l_1C_1 - l_xC_2 + vS_2)C_0 + wS_0 & -l_1S_1S_0 & (vC_2 + l_xS_2)S_0 & \varphi^{\mathrm{T}}S_2S_0 & -\psi^{\mathrm{T}}C_0 \\ 0 & l_1C_1 & vS_2 - l_xC_2 & -\varphi^{\mathrm{T}}C_2 & 0 \end{bmatrix}$$

Recall that $v \approx \varphi^{\mathrm{T}}\nu$ and $w \approx \psi^{\mathrm{T}}\eta$. Hence, in the product $J\dot{\mathbf{Q}}$, the φ^{T} and ψ^{T} terms above will multiply $\dot{\nu}$ and $\dot{\eta}$ respectively to yield $\dot{v}$ and $\dot{w}$.

Now consider the flexible inertia matrix, D_f, as defined in (3.76). Ξ_1, Ξ_2 and Ξ_3 are defined in (3.66), (3.72) and (3.73) respectively. The Jacobians J^v and J^Ω for the five-bar robot are described by (3.120) and (3.121). Finally, the rotation R of frame e with respect to the base frame is evident from (3.118) and (3.119) above. Before substituting these equations into (3.76), consider each of the terms in (3.76). For brevity, let C_{2-1} denote $\cos(q_2 - q_1)$ and S_{2-1} represent $\sin(q_2 - q_1)$.

$$(J^v)^{\mathrm{T}}J^v l_5 = \begin{bmatrix} (l_1C_1 - l_4C_2)^2 l_5 & 0 & 0 \\ 0 & l_1^2 l_5 & -l_1 l_4 l_5 C_{2-1} \\ 0 & -l_1 l_4 l_5 C_{2-1} & l_4^2 l_5 \end{bmatrix}$$

$$(J^v)^{\mathrm{T}}R\Xi_1 J^\Omega = \begin{bmatrix} (l_1C_1 - l_4C_2)(\frac{1}{2}l_5^2 C_2 - \alpha_v^{\mathrm{T}}\nu) & 0 & 0 \\ l_1\alpha_w^{\mathrm{T}}\eta S_1 & 0 & l_1(\frac{1}{2}l_5^2 C_{2-1} - \alpha_v^{\mathrm{T}}\nu S_{2-1}) \\ -l_4\alpha_w^{\mathrm{T}}\eta S_2 & 0 & -\frac{1}{2}l_4 l_5^2 \end{bmatrix}$$

$$(J^\Omega)^{\mathrm{T}}\Xi_3 J^\Omega = \begin{bmatrix} \eta^{\mathrm{T}}\eta + \nu^{\mathrm{T}}\nu S_2^2 - 2\beta_v^{\mathrm{T}}\nu S_2 C_2 + \frac{1}{3}l_5^3 C_2^2 & 0 & (\beta_w^{\mathrm{T}}S_2 + \nu^{\mathrm{T}}\gamma C_2)\eta \\ 0 & 0 & 0 \\ (\beta_w^{\mathrm{T}}S_2 + \nu^{\mathrm{T}}\gamma C_2)\eta & 0 & \frac{1}{3}l_5^3 + \nu^{\mathrm{T}}\nu \end{bmatrix}$$

Recall that $(J^v)^{\mathrm{T}}J^v l_5$ is proportional to the translational kinetic energy of the flexible beam based on the position of the clamped end, while $(J^\Omega)^{\mathrm{T}}\Xi_1 J^\Omega$ is the

rotational kinetic energy of the beam. The term $(J^v)^T R \Xi_1 J^\Omega$ and its transpose account for the fact that the reference frame for the beam is at the clamped end rather than at the beam's centre of mass.

The coupling between the rigid robot and flexible beam is represented by Π as defined in (3.75). Substituting for the five-bar robot in particular:

$$\Pi = \begin{bmatrix} \gamma\eta S_2 & -l_1\alpha_v C_{2-1} & l_4\alpha_v + \beta_v \\ -(l_1 C_1 - l_4 C_2)\alpha_w - \gamma\nu S_2 + \beta_w C_2 & 0 & 0 \end{bmatrix}$$

These terms may be substituted into (3.76) to obtain the final flexible inertia matrix, D_f. The Coriolis and centripetal terms for the flexible beam are then computed by taking the matrix Jacobian of this inertia matrix, as described by (3.92)-(3.113). Since the equations are quite complex, they are not included here.

In order to calculate the gravitational potential energy of the beam, the height of the beam centre of mass is required. From (3.78), (3.118) and (3.119):

$$h = l_1 S_1 - (l_4 + \tfrac{1}{2} l_5) S_2 - \tfrac{1}{l_5} \alpha_v^T \nu C_2$$

As expected, h is independent of the horizontal modal coordinates, η, of the flexible beam. The role of the flexibility in shifting the location of the centre of mass is also clear from this equation.

The strain energy is independent of the base robot so (3.80) does not require any substitutions for base robot parameters.

To obtain the full dynamic equations as described in section 3.9.2 requires all the partial derivatives mentioned above. While many terms have been eliminated in the equations of that section, the final dynamic equations are extremely complicated and will not be included here. The algebra is mechanical and the purpose of this example is only to highlight certain aspects of the model, particularly the Jacobian,

J, and flexible inertia matrix, D_f. Derivation of the full dynamic equations requires the dynamics of the rigid base robot as well. The dynamics of the rigid robot are considered in greater detail in the discussion on decoupling and balancing of the robot in section 4.2.

Chapter 4

Apparatus

It is difficult to assess the empirical results of the literature on haptic interfaces and force control without performing experiments to ascertain the "feel" provided by a haptic interface and the associated problems, such as high frequency limit cycles present in the implementation of virtual walls [7]. Hence, a simple one-dimensional experimental apparatus was employed to validate existing results in the literature and explore new control algorithms.

However, the key to a mechatronics approach is the balance of mechanical structure and control. While the first apparatus was used for some preliminary experiments, it exhibited significant friction and was limited to one degree-of-freedom. For more sophisticated control algorithms, a better device was required. Hence, a modified version of the three degree-of-freedom robot of Ching and Wang [10] is employed for the more advanced control algorithms in this work. In [10], the robot is rigid and specifically designed to be gravity-balanced and dynamically decoupled in order to simplify the control problem. In this work, the last link of the robot is replaced by a "flexible" link. Strain gauges on the link allow the robot

structure itself to be employed as a three degree-of-freedom force sensor and these measurements are utilized for control.

4.1 One D.O.F. Experiment

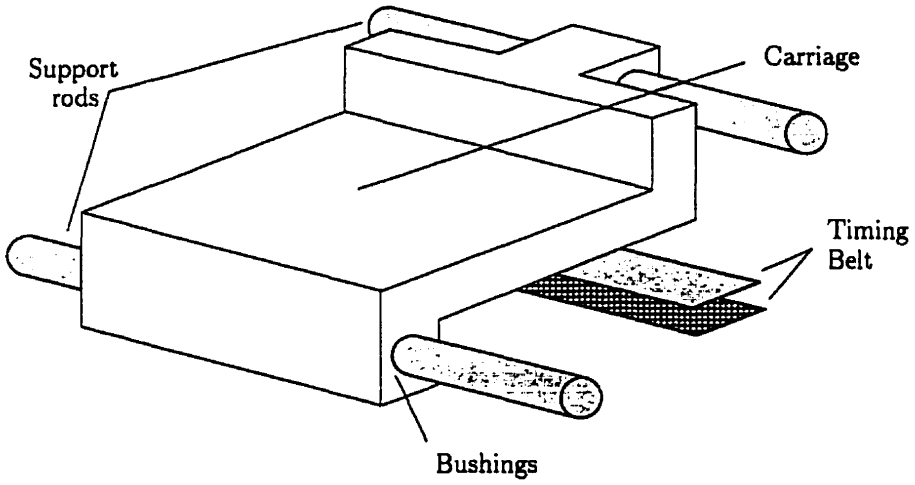


Figure 4.1: One D.O.F. Apparatus

The first experimental apparatus[1] consists of a one degree of freedom manipulator with approximately 42 cm of travel. An optical encoder is used to measure the end-effector position with an accuracy of about 40 μm. The end-effector rides along two parallel metal rods, driven by a timing belt connected to a pulley on the motor shaft. The apparatus is illustrated in Figure 4.1. The motor is a DC servomotor. Due to the use of bushings instead of bearings, there is a fair amount of friction in the system. The motor has a 4:1 gear reduction and the pulley radius

[1]constructed with the assistance of Kevin Krauel, a Research Associate at the University of Waterloo

is approximately 0.5 cm. Define:

$$r = \text{the gear ratio}$$

$$J = \text{motor and gear inertia}$$

$$B = \text{motor and gear damping}$$

$$\tau = \text{motor torque}$$

$$\tau_l = \text{load torque}$$

$$\theta = \text{motor shaft angle}$$

Then the motor dynamics may be modelled as:

$$J\ddot{\theta} + B\dot{\theta} = \tau - r\tau_l$$

Now, let:

$$p = \text{pulley radius}$$

$$M_l = \text{mass of carriage}$$

$$B_l = \text{damping of carriage}$$

$$f = \text{friction of bushings}$$

$$x = \text{carriage position}$$

Then the load presented to the gear shaft by the carriage is:

$$\tau_l = p(M_l\ddot{x} + B_l\dot{x} + f)$$

Using the relationship between x and θ, $x = rp\theta$, then the overall dynamics of the system are:

$$\left(\frac{J}{rp} + rpM_l\right)\ddot{x} + \left(\frac{B}{rp} + rpB_l\right)\dot{x} = \tau - rpf$$

Let F denote the motor torque reflected onto the linear travel of the carriage i.e. $\tau = rpF$. Then the above equation can be rewritten as:

$$\left(\frac{J}{r^2p^2} + M_l\right)\ddot{x} + \left(\frac{B}{r^2p^2} + B_l\right)\dot{x} = F - f$$

Thus, neglecting friction, it is easily seen that the haptic device has a simple second-order transfer function of the form:

$$P(s) = \frac{X(s)}{F(s)} = \frac{1}{as^2 + bs} \quad \text{or} \quad P'(s) = \frac{X(s)}{\tau(s)} = \frac{rp}{as^2 + bs} \tag{4.1}$$

where a and b are defined appropriately.

4.2 Direct-Drive Robot

Most of the experiments are performed using a unique five-bar robot designed and built at the University of Waterloo. The robot uses direct-drive motors so issues such as the non-linear effects of gear backlash or harmonic drives are not present. Furthermore, a novel counterbalancing scheme developed by David Wang and Mennas Ching [10] allows the robot to be gravity balanced and dynamically decoupled. Hence, this direct-drive robot is ideally suited for research into force control and haptic devices.

A schematic of the five-bar robot is illustrated in Figure 4.2. The robot consists of five links. Links 1 through 4 form the traditional parallelogram structure of a five-bar robot. Link 5 is considered a "flexible" link. All the links comprise square aluminum tubing in order to reduce the weight of the links while maintaining structural rigidity.

The joints are referenced according to the links they join. For example, the joint between links 2 and 3 is referred to as joint 2-3. Thus, there are five joints: 1-2,

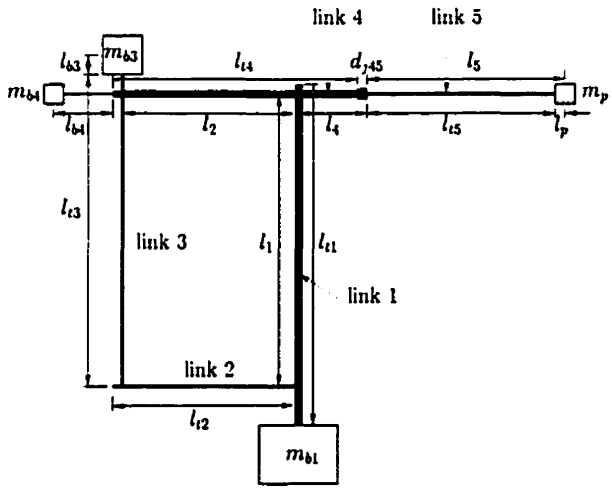


Figure 4.2: Five-Bar Robot Apparatus

2-3, 3-4, 4-5 and 1-4. All joints but joint 4-5 are rotational joints with bearings. Joint 4-5 is simply a clamp binding links 4 and 5 together. Note that the aluminum tubing typically extends beyond the joint in order to provide room for the bearings to attach to the link. For example, the length of the link 2 tubing, l_{t2}, is longer than l_2, which is traditionally considered to be the length of link 2 in a five-bar configuration. All variables associated with aluminum tubing are demarcated with a subscript t, followed by the link number. Variables associated with joints are denoted using a subscript j, followed by the numbers of the links being joined. For example, m_{j23} is the mass of the bearings, etc. constituting joint 2-3.

There are three adjustable counterbalances on the robot, which are numbered according to the link upon which they are fastened. Counterbalances 1 and 4 are strictly for balancing the robot. Counterbalance 3 is used to dynamically decouple the robot. All variables associated with counterbalances are denoted with a subscript b, followed by the link number. Positioning of the counterbalances will be

explained in greater detail in subsequent sections.

4.2.1 Balancing

Balancing the robot against gravity may be accomplished by adjusting the various counterbalances. The balancing equations may be extracted from the dynamic equations by equating the gravitational terms to zero, but this section presents the derivation of the balancing equations in a more intuitive fashion. It should be noted however, that balancing can only be accomplished if link 5 is assumed to be rigid. This observation can be proven from the general dynamic equations when flexibility is incorporated, but it will become clear from the ensuing discussion. Let:

m_1 = mass of link 1, including link 1 counterbalance, m_{b1}

m_2 = mass of link 2

m_3 = mass of link 3, including link 3 counterbalance, m_{b3}

m_r = mass of link 4 and 5, including counterbalance and "payload"

l_{c1} = centre of mass of link 1, including counterbalance

l_{c2} = centre of mass of link 2

l_{c3} = centre of mass of link 3, including counterbalance

l_{cr} = centre of mass of links 4 and 5, including counterbalance and "payload"

Consider the robot in the home position as depicted in Figure 4.3. For the robot to be balanced, the net moment about the fulcrum (shown as a large caret) must be zero. Hence a condition for balance is:

$$m_2 l_{c2} + m_3 l_2 - m_r l_{cr} = 0 \tag{4.2}$$

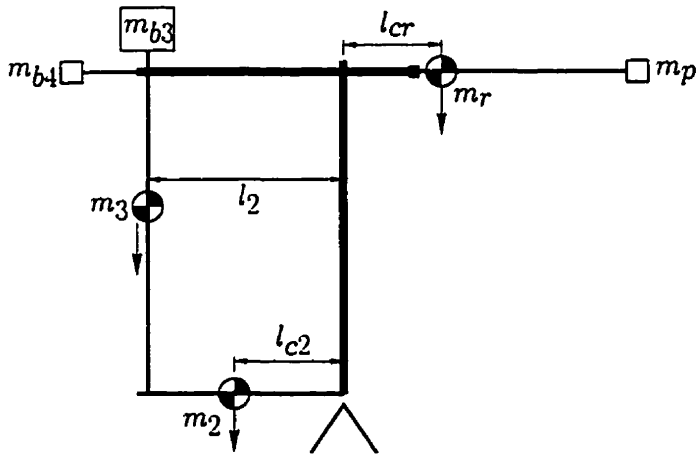


Figure 4.3: Horizontal Moment Balance

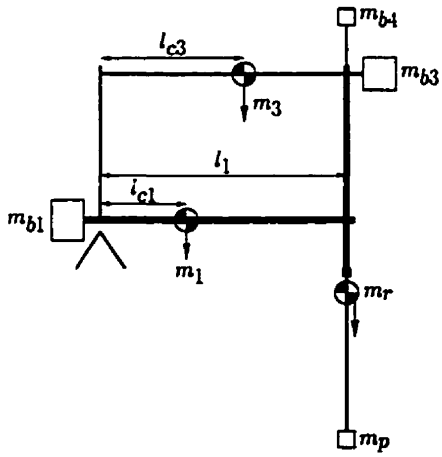


Figure 4.4: Vertical Moment Balance

Now consider the robot in the rotated position depicted in Figure 4.4. For the robot to be balanced, the net moment about the fulcrum must again be zero, yielding the following condition for balance:

$$m_1 l_{c1} + m_r l_{cr} + m_3 l_{c3} = 0 \tag{4.3}$$

In turns out that these are the only conditions for the robot to be gravity balanced. In other words, by satisfying these two conditions, the centre of mass of the robot is positioned at the origin (the fulcrum position). Furthermore, this centre of mass is independent of the joint angles. Since the origin is fixed in space, being in the centre of joint 1-2, no moment acts upon the robot due to gravity. Thus all gravitational terms are reduced to zero, *independent of robot configuration.* Thus, the non-linear effects of gravity have been completely eliminated by the novel mechanical design.

It should also be noted that the key quantities are mostly *moments*, rather than individual masses or distances. Hence, when designing the robot, it is possible to solve for the moments associated with each counterbalance without specifying a particular counterbalance mass. Thus, the counterbalance mass can be chosen, for example, such that the counterbalance will be positioned in the centre of its adjustable range. Then, if there are any errors in the computations, there is more room left for adjusting the counterbalance positions.

Now consider the flexibility of link 5. The vertical moment balance is unchanged, but the link will sag under the influence of gravity when the robot is in the home position of Figure 4.3. This sag will shift the centre of mass and affect the robot moment balance. Even if the robot is balanced in the home position, accounting for the link flexibility, it will not be balanced in an intermediate position where the sag will differ. Hence, flexibility destroys the perfect gravity compensation theoretically

offered by the counterbalances. However, in the case of the experimental apparatus, the deflection of link 5 under gravity is imperceptible to the naked eye and the effect of the flexibility on the centre of mass is negligible. Let:

m = mass of flexible link

g = acceleration due to gravity (9.81 m/s^2)

l = length of link

w = width of square tubing

d = thickness of tube wall

E = modulus of elasticity

I = area moment of inertia

For a cantilever beam under uniform gravitational load, the maximum deflection of the tip, v_{max}, is [89]:

$$v_{max} = \frac{mgl^3}{8EI} \tag{4.4}$$

The parameters of the flexible link [90] used with the experimental apparatus are:

$$m = 0.077 \text{ kg}$$

$$l = 47.35 \text{ cm}$$

$$w = 1.27 \text{ cm}$$

$$d = 0.127 \text{ cm}$$

$$E = 69 \times 10^5 \text{ N-cm}^2$$

$$I = \tfrac{1}{12}w^4 - \tfrac{1}{12}(w - 2d)^4 = 0.128 \text{ cm}^4$$

Substituting these parameters into (4.4), the maximum deflection of the flexible link for this apparatus is only $v_{max} = 0.0114$ cm or 0.1 mm. Hence, for all intents and purposes, the experimental apparatus is gravity balanced.

4.2.2 Decoupling

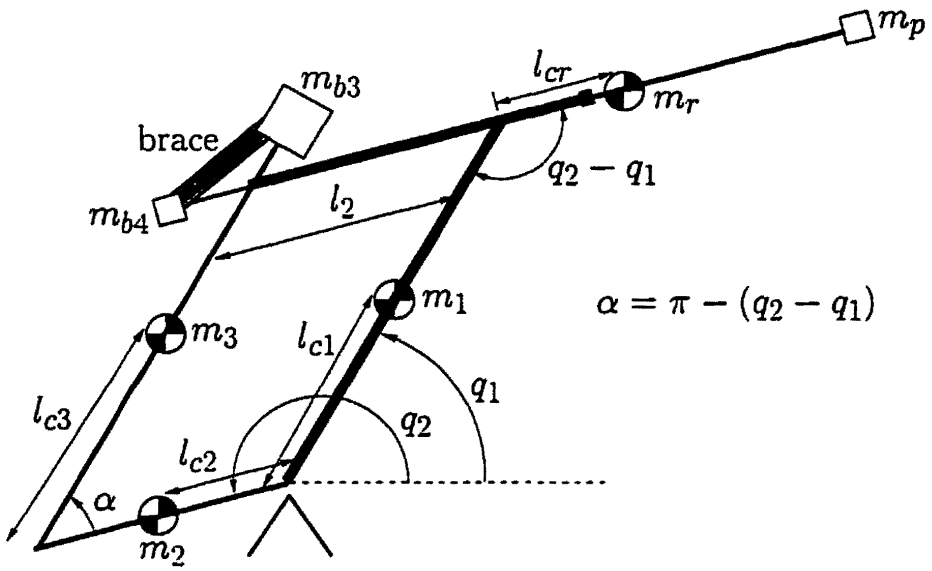


Figure 4.5: Decoupling

The condition for dynamic decoupling is less readily apparent. For the dynamic equations to be decoupled, the inertia matrix must be diagonal and the entries in the inertia matrix should preferably be constant. Hence, consider the robot schematic in Figure 4.5, which shows the centre of mass of each link. Since link 5 is essentially rigid, let the combination of link 4 and link 5 be called link r, with centre of mass, m_{cr}.

Suppose the robot is restricted as depicted in the figure, so that $q_2 - q_1$ is constant, yet q_1 is free to rotate. Let the inertia of link i about its centre of mass be denoted $I_{ci}, i = 1, 2, 3, r$. Define the inertia of link i with respect to the fulcrum axis by I_i. By the parallel axis theorem, the inertias of links 1 and 2 are constant,

being:

$$I_1 = I_{c1} + m_1 l_{c1}^2$$

$$I_2 = I_{c2} + m_2 l_{c2}^2$$

Define $\alpha = \pi - (q_2 - q_1)$. Applying the cosine law along with the parallel axis theorem, the inertias of the other two links are:

$$I_3 = I_{c3} + m_3(l_2^2 + l_{c3}^2 - 2 l_2 l_{c3} \cos(\alpha))$$

$$I_r = I_{cr} + m_r(l_1^2 + l_{cr}^2 - 2 l_1 l_{cr} \cos(q_2 - q_1))$$

Hence, the total inertia, I, of the robot is:

$$I = I_{c1} + I_{c2} + I_{c3} + I_{cr} + m_1 l_{c1}^2 + m_2 l_{c2}^2 + m_3(l_2^2 + l_{c3}^2) + m_r(l_1^2 + l_{cr}^2) + 2(m_3 l_2 l_{c3} - m_r l_1 l_{cr}) \cos(q_2 - q_1) \quad (4.5)$$

The only term in the inertia that depends on the robot configuration is the term in $\cos(q_2 - q_1)$. However, this term may be eliminated by carefully adjusting the position of the link 3 counterbalance such that

$$m_3 l_{c3} l_2 - m_r l_{cr} l_1 = 0 \quad (4.6)$$

Satisfying (4.6) results in a total inertia that is constant. In fact, $I_3 + I_r$ is constant, so there exists a position of the counterbalance for which the sum of the link 3 and link r inertias is independent of the angle, $q_2 - q_1$, between the two links. As one link's centre of mass moves away from the pivot, the other link's centre of mass moves toward the pivot in exactly the right proportion to maintain a constant moment of inertia.

It turns out that (4.6) is the condition required to dynamically decouple joints q_1 and q_2, not only in terms of inertia, but also in the Coriolis and centripetal components of the dynamics. This intriguing characteristic of the robot will be investigated more thoroughly in the next section.

4.2.3 Robot Dynamics

The previous three sections explore the balancing and decoupling of the five-bar robot from an intuitive perspective. The full dynamics of the robot are calculated in this section in order to rigorously defend the analysis of the preceding sections.

Reconsider Figures 3.6 and 3.7, which depict the frames associated with each link and its centre of mass. For this analysis, it will be assumed that link 5 is rigid, in which case it may simply be considered an extension of link 4. Hence, as before, let the combination of link 4 and link 5 be called link r, with centre of mass, m_{cr}. Treat the links as lines and the counterbalances as point masses. Thus, a scalar, I_i, may be used to represent the inertia of link i with respect to the z-axis through its centre of mass. Using the DH parameters defined in Table 3.1 then, it is relatively straightforward to derive the inertia matrix of the robot. Define the constant coefficients:

$$\alpha_1 = (I_1 + I_3 + m_1 l_{c1}^2 + m_3 l_{c3}^2 + m_r l_1^2) \tag{4.7}$$

$$\alpha_2 = (I_2 + I_r + m_2 l_{c2}^2 + m_r l_{cr}^2 + m_3 l_2^2) \tag{4.8}$$

$$\alpha_3 = (m_3 l_{c3} l_2 - m_r l_{cr} l_1) \tag{4.9}$$

For convenience, assume the following notation for $i = 1, 2$:

$$C_i \triangleq \cos(q_i), \quad S_i \triangleq \sin(q_i), \quad C_{2-1} \triangleq \cos(q_2 - q_1); \quad \mathbf{Q} = [q_0 \quad q_1 \quad q_2]^{\mathrm{T}}$$

Then the inertia matrix for the five-bar robot is:

$$D(\mathbf{Q}) = \begin{bmatrix} \alpha_1 C_1^2 + \alpha_2 C_2^2 + 2\alpha_3 C_1 C_2 & 0 & 0 \\ 0 & \alpha_1 & \alpha_3 C_{2-1} \\ 0 & \alpha_3 C_{2-1} & \alpha_2 \end{bmatrix} \tag{4.10}$$

The only off-diagonal term is $\alpha_3 C_{2-1}$. By adjusting the link 3 counterbalance such that $\alpha_3 = (m_3 l_{c3} l_2 - m_r l_{cr} l_1) = 0$, the inertia matrix becomes diagonal. Notice that this expression is the decoupling condition derived in section 4.2.2.

Now consider the Coriolis and centripetal terms when $\alpha_3 = 0$. From (3.81), the Coriolis and centripetal terms are completely characterized by $\frac{\partial D}{\partial \dot{\mathbf{Q}}}$. Evaluating (3.81) yields Coriolis and centripetal terms of:

$$(I_3 \otimes \dot{\mathbf{Q}}^{\mathrm{T}}) C(\mathbf{Q}) \dot{\mathbf{Q}} = \begin{bmatrix} -2(\alpha_1 C_1 S_1 \dot{q}_1 + \alpha_2 C_2 S_2 \dot{q}_2)\dot{q}_0 \\ \alpha_1 C_1 S_1 \dot{q}_0^2 \\ \alpha_2 C_2 S_2 \dot{q}_0^2 \end{bmatrix}$$

The second and third rows correspond to the q_1 and q_2 joints respectively. Yet the second row does not depend on q_2, nor does the third row depend on q_1. Hence, there is no coupling between q_1 and q_2 even in the Coriolis and centripetal terms! Thus, adjusting the link 3 counterbalance to satisfy the decoupling condition *dynamically* decouples q_1 and q_2. There are, of course, interactions with the base joint which cannot be eliminated.

Now consider the gravitational terms. According to section 4.2.1, it is possible to completely eliminate all gravitational terms from the dynamics of the robot. It is relatively straightforward to show that the potential energy of the rigid robot is:

$$V = g\left((m_1 l_{c1} + m_3 l_{c3} + m_r l_1) S_1 + (m_2 l_{c2} + m_3 l_2 - m_r l_{cr}) S_2\right)$$

The potential energy is clearly zero for all robot configurations if the coefficients of S_1 and S_2 are zero. Hence, the robot is dynamically balanced if:

$$m_1 l_{c1} + m_3 l_{c3} + m_r l_1 = 0$$

$$\text{and} \quad m_2 l_{c2} + m_3 l_2 - m_r l_{cr} = 0$$

These two balancing conditions are exactly the same conditions derived in section 4.2.1. Combining these results together, the final dynamic equations of the rigid robot are:

$$\begin{aligned} (\alpha_1 C_1^2 + \alpha_2 C_2^2)\ddot{q}_0 &- 2(\alpha_1 C_1 S_1 \dot{q}_1 + \alpha_2 C_2 S_2 \dot{q}_2)\dot{q}_0 &= \tau_0 \\ \alpha_1 \ddot{q}_1 &+ \alpha_1 C_1 S_1 \dot{q}_0^2 &= \tau_1 \\ \alpha_2 \ddot{q}_2 &+ \alpha_2 C_2 S_2 \dot{q}_0^2 &= \tau_2 \end{aligned}$$

Note that damping has been ignored but could readily be included. Also,

$$|cos(x)sin(x)| \leq \tfrac{1}{2} \quad \forall x \in \Re$$

Furthermore, in the case of a virtual wall, $\ddot{q}_i$ may be large but $\dot{q}_i$ is typically small. Indeed, in any case where a human operator is manipulating the robot end-effector, the velocities are typically small and thus, the Coriolis and centripetal terms, which are related to the square of velocity, are even smaller. Hence, the robot behaves very much like a set of single-input single-output (SISO) double-integrator systems in this case. If damping is included, then each joint of the robot behaves like a simple second-order system and only the base joint has an inertia that depends upon the robot configuration. This behaviour motivates the system identification of section 5.3.2.

4.2.4 Adjusting the Counterbalances

The purpose of the counterbalances is to adjust the positions of the centres of mass such that the balancing equations (4.2) and (4.3) and decoupling equation (4.6) are satisfied. The counterbalances are screw-mounted so that their positions may be changed.

Link 2 has no counterbalance, so l_{c2} is fixed. Thus, the only quantity which can be adjusted in (4.2) is l_{cr}. By changing the position of counterbalance m_{b4}, l_{cr}

may be tuned such that (4.2) is satisfied. In practice, this feat is accomplished by placing the robot in the home position and adjusting the location of m_{b4} until the robot is balanced in that position.

Once the position of m_{b4} has been established, the decoupling equation described in the previous section determines the value of l_{c3}, and hence the location of the link 3 counterbalance. It may be possible to determine the correct location of the counterbalance experimentally as follows. Attach a lightweight brace near joint 1 that fixes the angle between link 1 and link 2 such that the robot is in the fully extended position i.e. $(q_2 - q_1) \rightarrow \pi$. Apply a brief torque pulse to joint 1 from rest and measure the velocity at the end of the pulse. Now fasten the brace such that the robot is in the fully contracted position i.e. $(q_2 - q_1) \rightarrow 0$. Repeat the experiment using an identical torque pulse. Adjust the position of the link 3 counterbalance until the velocity is the same in both cases.

The final degree of freedom is the location of the link 1 counterbalance m_{b1}. Adjusting its position enables the second balancing condition, (4.3), to be satisfied. In practice, this balance may be accomplished by adjusting the the location of the link 1 counterbalance until the robot is balanced in any configuration.

Chapter 5

System Identification

As in any control application, control design is facilitated by having a relatively accurate model of the system dynamics. The experimental apparatus of Figure 4.2 comprises several components, including the current amplifiers, strain gauges, motors, joints and robot links. Since the dynamic equations of the rigid structure are derived in the preceding section, only those components or non-linear effects which warranted system identification will be examined here.

5.1 Power Amplifiers

Since the dynamics of mechanical systems are generally much slower than the dynamics of electrical systems, the dynamics of the power amplifiers are not considered. However, the amplifier gain and offset are critical factors because they affect the magnitude of the controller parameters. These factors are experimentally determined in [90]. The results of these experiments are enumerated in Table 5.1. The torque constants of each motor are also included in the table.

Axis	Transconductance (A/V)	Offset (A)	Torque Constant (N-m/A)
Base	3.98	-0.116	0.373
Axial	3.87	0.054	0.373
Vertical	3.96	0.001	0.373

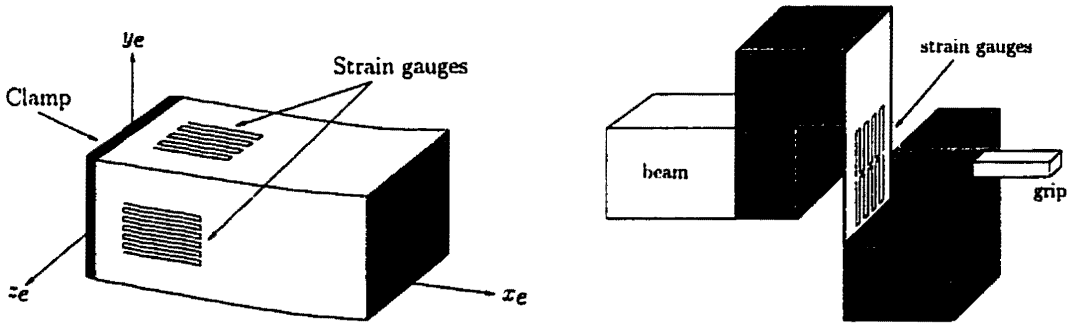
Table 5.1: Current Amplifier Measurements

5.2 Strain Gauge Calibration

Equation (3.115) in section 3.11 encapsulates the approximate relationship between the strain sensed at the clamped end of the flexible beam and the force exerted on the beam at the tip. Indeed, the force is approximately proportional to the strain, since the coefficients E, I_v, h and l in (3.115) are all constant (within the assumptions of the Euler-Bernoulli beam model). Hence, by identifying this constant of proportionality, it should be possible to estimate the torques at the joints based on the generalized coordinates and the strain readings.

5.2.1 What is Measured

Section 3.6 describes the relationship between the force, $\mathbf{F}$, at the end-effector and the joint torques, τ. However, as discussed in that section, the force $\mathbf{F}$ is expressed in *base frame coordinates*. The strain gauges are not configured to measure strain along the axes of the base frame coordinate system. Instead, they measure strain along the end-effector coordinate axes. Thus, equation (3.58) of section 3.6 may not be applied directly. Fortunately, the robot geometry is relatively simple, and the relationship between the forces deduced from the strain gauge readings and the



(a) Gauges at clamped end of beam (b) Axial measurement unit

Figure 5.1: Strain Gauge Positions

joint torques is straightforward to derive.

5.2.2 Mapping to Joint Space

The positions of the strain gauges are illustrated in Figure 5.1. Figure 5.1(a) shows the strain gauges that measure the strain of the beam in the horizontal (z_e) and vertical (y_e) directions. Strain gauges were hand-mounted in pairs, on either side of the beam. Since the strain in the beam along the neutral axis, x_e, is negligible, the unit depicted in Figure 5.1(b) is used to measure forces exerted along the neutral axis. The unit is an LCL-010 load cell from Omega Engineering Inc.

The horizontal and axial strain gauge signals are amplified using Intelligent Instrumentation PCI-5B38 strain gauge amplifiers. The vertical strain gauge signal is amplified using a higher gain DataForth SCM5B38 amplifier. The amplifiers are mounted directly on the robot base to minimize the distance between the strain gauges and the amplifiers. A separate 12V power supply, line regulated to 5V using

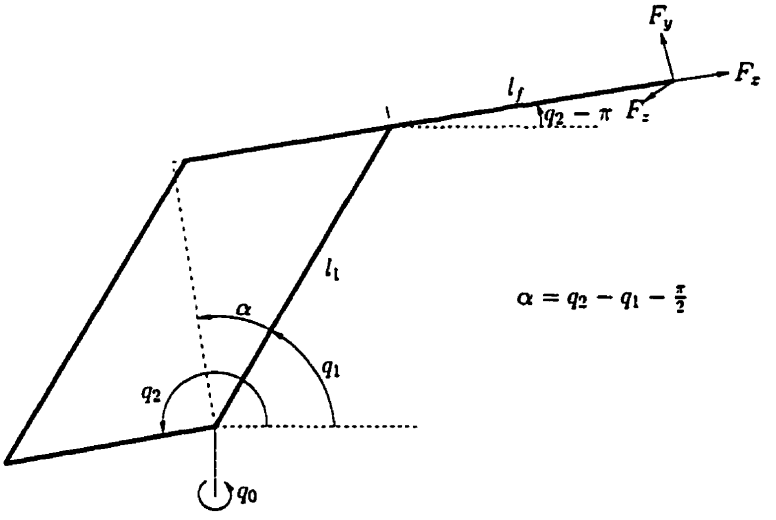


Figure 5.2: Mapping Strain to Torque

an MC7805ACT regulator on the robot, is used to power the strain gauge amplifiers. The amplifier outputs are measured using 13-bit analog-to-digital converters on a Quanser Consulting MultiQ data acquisition card.

Consider Figure 5.2, which depicts the forces acting upon the end-effector. The subscripts x, y and z correspond to the axes of the end-effector frame. The magnitudes of F_x, F_y and F_z can be deduced from the strain gauge readings via (3.115). Let τ_i denote the torque contributed to joint i by the end-effector forces. Assuming small deflections, the beam may essentially be considered rigid and thus by simple geometry:

$$\tau_0 = -F_z(l_1 \cos(q_1) + l_f \cos(q_2 - \pi)) = -F_z(l_1 \cos(q_1) - l_f \cos(q_2)) \quad (5.1)$$

$$\tau_1 = -F_x l_1 \cos(q_2 - q_1 - \tfrac{\pi}{2}) \qquad = -F_x l_1 \sin(q_2 - q_1) \quad (5.2)$$

$$\tau_2 = F_y l_f \quad (5.3)$$

where:

$$l_1 = 0.30 \text{ m} \tag{5.4}$$

$$l_f = 0.60 \text{ m} \tag{5.5}$$

Given the tiny displacement of the beam tip in comparison to the link lengths l_1 and l_f, the treatment of the beam as rigid for the purpose of mapping the end-effector forces to joint torques is reasonable. It must be remembered that the device is intended as a haptic interface and small torque errors will be indistinguishable to the human operator.

The joint angles are measured using PMI Motion Technologies type B250213 optical encoders. Table 5.2 enumerates the conversion factors between encoder counts and angular position in radians for each joint.

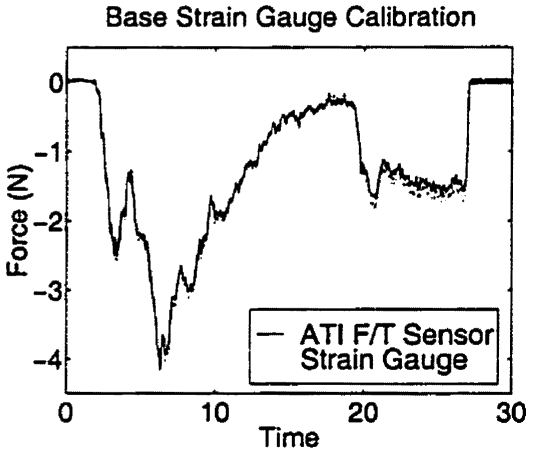
	Horizontal (q_0)	Axial (q_1)	Vertical (q_2)
$\frac{\text{rad}}{\text{count}}$	$\frac{\pi}{6000}$	$-\frac{\pi}{6000}$	$\frac{\pi}{4000}$

Table 5.2: Optical Encoder Conversion Factors

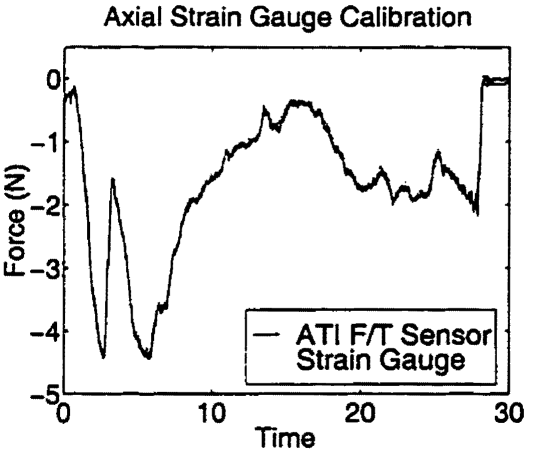
5.2.3 Calibration Experiments

Using an FT3623 nano force/torque sensor system manufactured by ATI Industrial Automation, the strain gauge readings were calibrated to end-effector forces (in Newtons).

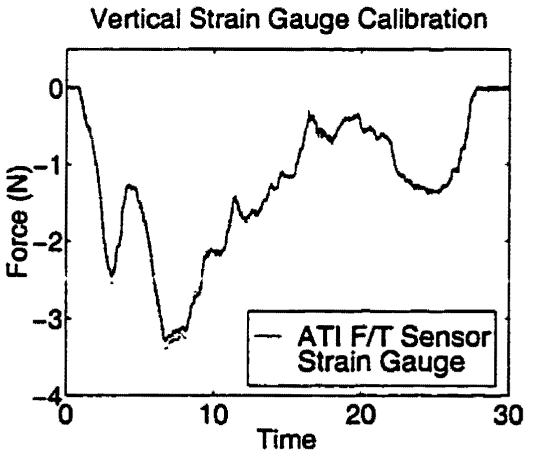
Figure 5.3 illustrates the agreement between the force measured using the commercial sensor and the force deduced from the strain gauge readings. The solid



(a) Horizontal Strain (z_e-direction)



(b) Axial Strain (x_e-direction)



(c) Vertical Strain (y_e-direction)

Figure 5.3: Strain Gauge Calibration Experiments

	Horizontal (q_0)	Axial (q_1)	Vertical (q_2)
$\frac{\text{Force (N)}}{\text{Voltage (V)}}$	8.05	13.4	5.31

Table 5.3: Strain Gauge Calibration Constants

lines correspond to the commercial sensor and the dotted lines depict the calibrated strain measurements. Note that despite the crude approximation of equation (3.115), agreement between the commercial sensor and the strain gauge readings is excellent. These measurements confirm that the robot structure itself can be used as a force sensor. It should also be observed that the "flexible" beam is actually quite rigid – tip deflections are typically less than 1 cm, even when simulating a virtual wall. The calibration constants between the forces in Newtons and the amplified strain gauge voltages are listed in Table 5.3.

5.2.4 Force Cancellation

The calibration experiments in the preceding section verify that the flexible link of the robot may be used as a reliable force sensor. However, equations (5.1)-(5.3) are not used in those experiments. To assess the accuracy of these equations and the efficacy of using strain gauge readings as force measurements, a force cancellation experiment was conducted. By applying an equal and opposite reaction force to the force exerted by the user on the end-effector, it should be possible, given reliable force measurements, to cancel the user's applied force so that the net force acting on the robot is zero. In this case, the robot will not move (assuming the initial velocity is zero).

Figure 5.4 illustrates the force cancellation control strategy. The unshaded

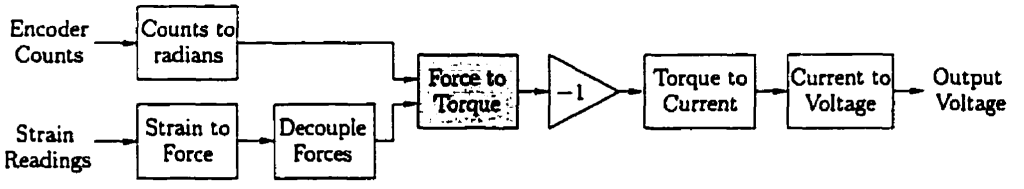


Figure 5.4: Force Cancellation

blocks denote simple linear transformations that are implemented as matrix gains. The shaded block represents the non-linear transformation defined by (5.1)-(5.3).

The strain gauges outputs are read and converted to end-effector forces using the calibration constants defined in Table 5.3. Now, some cross-coupling occurs between the strain readings. Since the gauges were hand-mounted they are not perfectly aligned with the neutral axis of the flexible beam. Hence, the axes the horizontal and vertical gauges measure are not perfectly orthogonal, resulting in some cross-coupling. Furthermore, resistive gauges are not immune to strains in non-measurement directions. For example, a gauge aligned for measuring strain in the x-direction will still produce a small reading when perturbed in the y or z directions. The ratio of the strain in the measurement direction to the strain in the orthogonal direction is known as Poisson's ratio [89].

Let F_i, $i = x, y, z$ denote the scaled strain readings, such that F_i has units of Newtons. Denote the forces after compensation as $\overline{F_i}$. Hence, the compensation may be expressed as the matrix equation:

$$\begin{bmatrix} \overline{F_x} \\ \overline{F_y} \\ \overline{F_z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -0.10 \\ 0 & 1 & 0.70 \\ 0 & -0.02 & 1 \end{bmatrix} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}$$

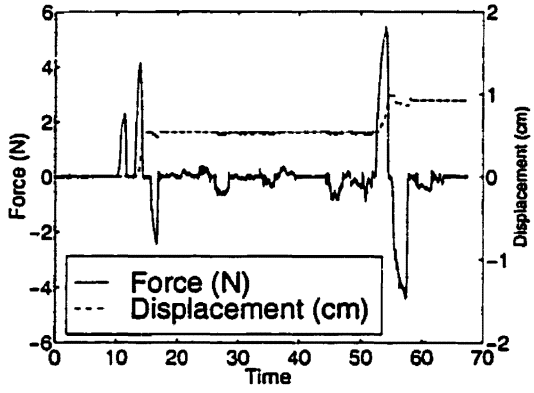
These compensation equations were not applied during the strain calibration tests

of Figures 5.3(a)-5.3(c) since the experiments were performed on each joint individually anyway. However, similar experiments were conducted in order to ascertain the degree of cross-coupling between the different strain readings.

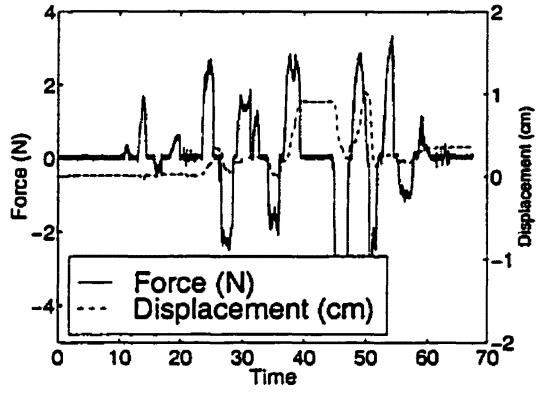
The non-linear equations (5.1)-(5.3) are applied to these compensated end-effector forces to map them to joint torques. The joint torque values are negated and then converted to output voltages for the digital-to-analog converters based on the pulse-width modulated current amplifier transconductances in Table 5.1 and each motor's torque constant. The net effect should be an equal and opposite force being exerted on the user, preventing them from moving the end-effector of the robot. The operation is sensitive to any errors in the measurements, calibration constants or parameters of the non-linear transformations from end-effector forces to joint torques.

Position information is only used to measure the generalized coordinates in the transformation equations. No position control is performed. The controller represents pure explicit force control. The results for each axis are shown in Figure 5.5. Note that the axes were tested concurrently, not individually. The force exerted by the user is represented by the solid line. The spikes indicate where the user has pushed or pulled on the end-effector.

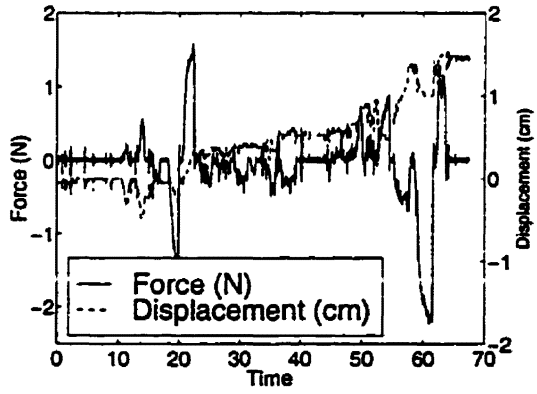
The dotted line depicts the displacement of the corresponding robot joint as a result of imperfect force cancellation. While deflections of approximately 1 cm may seem large, two observations must be made. First, since explicit force control is used, the total deflection may be expected to be large because any error in the force cancellation will cause the robot to accelerate as long as the force is applied. Yet, the deflection is still quite small, even after 70 seconds of prodding by the user. It is instructive to compare the amount the robot moves to the magnitude of the force and the duration for which it is applied. Second, the moment arm is also large so



(a) Horizontal (z_e-direction)



(b) Axial (x_e-direction)



(c) Vertical (y_e-direction)

Figure 5.5: Open-Loop Force Cancellation Experiments

the applied torque is substantial. For the horizontal and vertical axes, the moment arm is 61 cm and for the axial direction it is 30 cm.

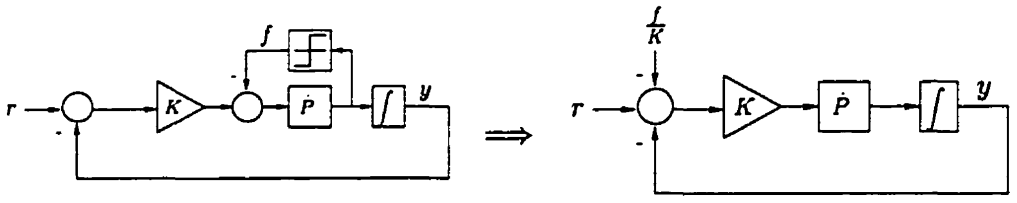
5.3 Friction Identification

5.3.1 Motivation

System identification techniques applied to the robot of Figure 4.2 produced inconsistent and typically impractical results. Many of the techniques available in standard control systems packages, such as Matlab, were attempted but the solutions either failed to predict the system response for different inputs, or clearly did not match the physical parameters of the system. However, around a given operating point, particularly with the dynamically decoupled, gravity balanced mechanical design, the system was expected to be essentially linear. Since system identification techniques were producing poor results, the non-linear effects of friction were assumed to be the confounding factor and a set of experiments were conducted to verify the presence of friction.

5.3.2 Identification Technique

Consider the proportional control system depicted in Figure 5.6(a). The plant, $\dot{P}$, represents the operator mapping torque to velocity. The signal, f, denotes the Coulomb friction, r denotes the reference input and y represents the output position. Without the presence of friction, the output y would scale linearly with the input r. Hence, for a step input, the peak magnitude of the output step response should be a linear function of the input magnitude.



(a) Proportional Control System

(b) "Equivalent" Model

Figure 5.6: Time Domain Friction Modelling

With friction, the output becomes a non-linear function of the input. According to the simple Coulomb kinetic friction model depicted in Figure 5.6(a), the friction, f, has constant magnitude with the same sign as the plant velocity.

The robot motors may be modelled approximately as second-order, minimum-phase systems. In the unit step response of second-order, minimum-phase systems under proportional control, the velocity is positive until the first peak. Addition of friction does not change this fact since Coulomb friction is a dissipative force. As long as the velocity does not change sign, the friction f will also not change sign, and will, in fact, be constant. Hence, from time zero *to the time of the first peak*, the system may be modelled as shown in Figure 5.6(b). Note that the constant friction term has been combined with the reference signal at the input.

The system shown in Figure 5.6(b) is linear. The reference signal for this "equivalent" system may be considered as the combination of r and the friction term, $\frac{f}{K}$. Let $\bar{r}$ denote the new reference signal. Then:

$$\bar{r} = r - \frac{f}{K}$$

Since the system is linear, the magnitude of the first peak will vary linearly with the magnitude of the input $\bar{r}$. This fact can be exploited to identify the Coulomb

friction, f. Let $P(s) = \frac{\dot{P}}{v}$ and let $T(s)$ denote the transfer function of the "equivalent" closed-loop system depicted in Figure 5.6(b). Let y_i be the magnitude of the first peak for a step input of magnitude r_i. Denote the inverse Laplace transform by $\mathcal{L}^{-1}$. Note that f is a constant independent of the input in this simple friction model. Then:

$$\begin{aligned} T(s) &= \frac{P(s)K(s)}{1 + P(s)K(s)} \\ y_i(s) &= \frac{T(s)}{s}\left(r_i - \frac{f}{K}\right) \\ \therefore y_i(t) &= \mathcal{L}^{-1}\left\{\frac{T(s)}{s}\left(r_i - \frac{f}{K}\right)\right\}(t) \\ &= \left(r_i - \frac{f}{K}\right)\mathcal{L}^{-1}\left\{\frac{T(s)}{s}\right\}(t) \end{aligned}$$

To identify the friction term, plot the magnitudes of the first peaks, $y_i(t_p)$ versus the magnitudes of the input steps, r_i, where t_p is the time to the first peak. The r-intercept of this plot is $r_0 = \frac{f}{K}$. Since K is known, the r-intercept may be used to determine the value of the friction via:

$$f = Kr_0 \tag{5.6}$$

Of course, once the friction is known, it is still important to identify the linear model of the plant, P. Suppose the robot actuator may be modelled as:

$$P(s) = \frac{1}{s(Js + B)}$$

where J is the combined motor and link inertia and B is the combined motor and joint viscous friction. Hence, when Coulomb friction is ignored in Figure 5.6(a), the closed-loop transfer function is the standard second-order system:

$$T(s) = \frac{K}{Js^2 + Bs + K} \equiv \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.7}$$

where:

$$\omega_n = \sqrt{\frac{K}{J}}$$
$$\zeta = \frac{B}{2\sqrt{KJ}}$$

It is well known that for the step response of an underdamped second-order system as expressed in (5.7), the time to first peak is:

$$t_p = \frac{\pi}{w_n\sqrt{1-\zeta^2}}$$

and the magnitude of the peak given an input $r - \frac{f}{K}$ is:

$$y_p = (1 + e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}})(r - \frac{f}{K})$$

Define:

$$\alpha = (1 + e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}})$$

such that $y_p = \alpha(r - \frac{f}{K})$. Hence, the slope of the $y_i(t_p)$ versus r_i plot is α. Let:

$$\beta = -\ln(\alpha - 1)$$

Thus, given α, the damping factor ζ is:

$$\zeta = \frac{\beta}{\sqrt{\pi^2 + \beta^2}} \tag{5.8}$$

Given the time to first peak, t_p, the natural frequency, ω_n, may be then be computed as:

$$\omega_n = \frac{\pi}{t_p\sqrt{1-\zeta^2}} \tag{5.9}$$

Note that since the time to first peak, t_p, for a linear second-order system is independent of the input step magnitude, r, and the equivalent friction model depicted

Dir.	Slope (rad/rad)	r-Int. (rad)	t_p (sec)	ω_n (rad/s)	ζ	J (kg-m^2)	B (N-m-s/rad)	f (N-m)
x_e	1.98	0.024	0.75	4.20	0.007	0.204	0.012	0.088
y_e	1.95	0.022	0.26	12.0	0.015	0.026	0.009	0.082
z_e	1.96	0.034	1.40	2.25	0.014	0.733	0.045	0.125

x_e = axial axis y_e = vertical axis z_e = horizontal axis

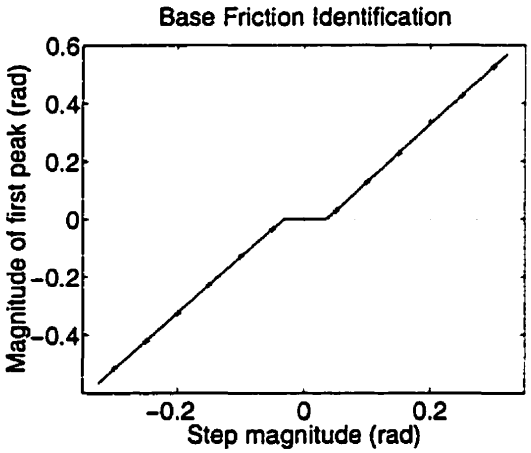
Table 5.4: Friction Identification Results

in Figure 5.6(b) is valid for this minimum-phase second-order system up to the time of the first peak, then (5.8) and (5.9) may be used to determine the parameters of the plant, P, based on the slope of the $y_i(t_p)$ versus r_i curve and the time to first peak, t_p. The friction is identified from the r-intercept of the same curve.

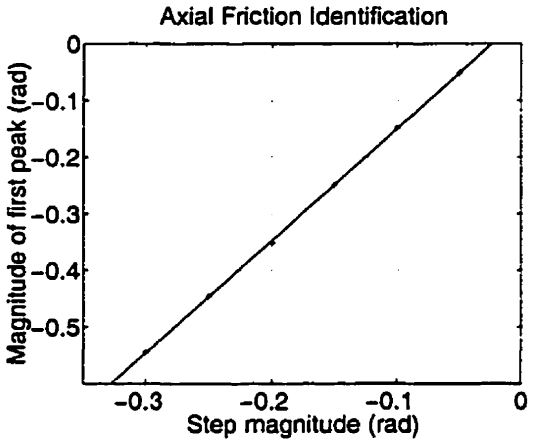
5.3.3 Results

While using an intercept to determine a quantity is typically a numerically sensitive operation and thus not ideal, it was found to be a reasonably reliable method for this experiment. Consider Figure 5.7, which illustrates the relationship between $y_i(t_p)$ and r_i for each joint. The responses are surprisingly linear with a clearly defined intercept. Note that for the base, both positive and negative step responses were conducted, in which case the friction changes sign also.

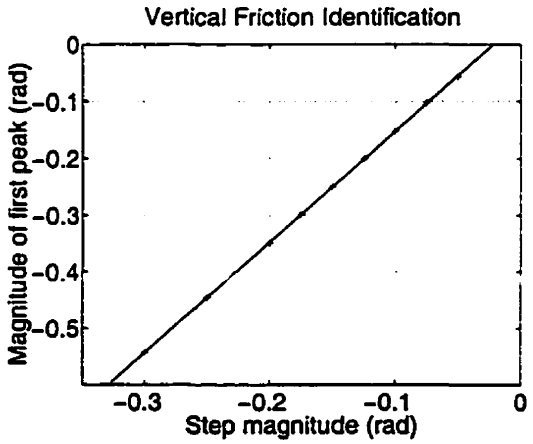
The results of the friction identification for each joint are enumerated in Table 5.4. Linear regression was used to determine the slope, α, and r-intercept of these curves. The average time to peak t_p is also shown. The natural frequency



(a) Horizontal Friction (z_e-direction)



(b) Axial Friction (x_e-direction)



(c) Vertical Friction (y_e-direction)

Figure 5.7: Friction Identification Experiments

ω_n and damping factor ζ were computed using (5.8) and (5.9). The inertia, J, and damping, B, were derived from ω_n and ζ using:

$$J = \frac{K}{\omega_n^2}$$

$$B = 2\zeta\sqrt{KJ}$$

The friction f was determined from the r-intercept using (5.6).

Direction	Value ($kg \cdot m^2$)
x_e	0.199
y_e	0.032
z_e	0.708

Table 5.5: Theoretical Inertias

Table 5.5 contains the inertias calculated for the robot based on the mass and dimensions of each component of the robot's links, motors and joints [90]. In other words, the inertias in Table 5.5 are the theoretical inertias based on knowledge of the robot structure. The value for the z_e direction is based on the home position of the robot. Note that the identified inertias of Table 5.4 agree quite well with the calculated inertias of Table 5.5. The identified inertia in the axial direction is 0.204 kg-m^2, compared to 0.199 kg-m^2 for the calculated model, or a percentage error of less than 2.7%.

The identified inertia in the vertical direction is 0.026 kg-m^2 compared to the calculated inertia of 0.032 kg-m^2, or a percentage error of 20%. This error may seem fairly large, but it is important to note that the inertia in the vertical direction is the hardest to identify because it is an order of magnitude smaller than the inertia for the other axes, and the range of motion in the vertical direction is much smaller.

The range of motion is important because it affects the magnitude of the step input that can be used in the identification process. Larger magnitude steps generally yield more reliable results since the non-linear effects of friction are less noticeable.

The identified inertia in the horizontal direction, for the robot in the home position, is 0.733 kg-m^2. The calculated inertia for the robot in the home position, is 0.708 kg-m^2, for a percentage error of less than 3.5%.

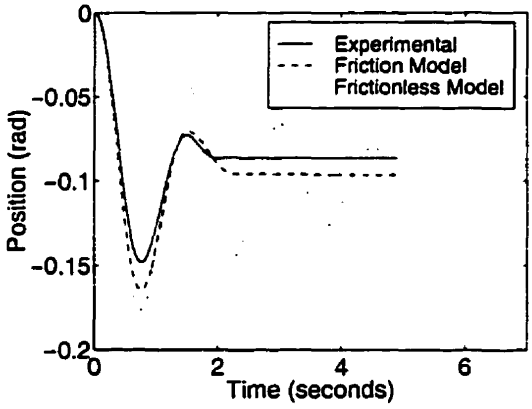
These percentage errors are well within the margin of error in both the calculated inertias and the system identification. Indeed, the correlation in the horizontal and axial directions between the calculated and identified inertias is surprisingly good.

Simulation of the friction model for the same closed-loop step responses reveals that the r-intercept calculated by linear regression does not necessarily result in the closest match between simulation and experiment. Small adjustments to the friction coefficient can improve the simulation results. For the axial joint, for example, using a friction coefficient of 0.061 N-m instead of 0.088 N-m yielded better results. While the percentage error between the hand-tuned friction coefficient of 0.061 N-m and the identified coefficient of 0.088 N-m is 30.6%, this error is not unreasonable given the intercept method of identification and the simple kinetic friction model utilized.

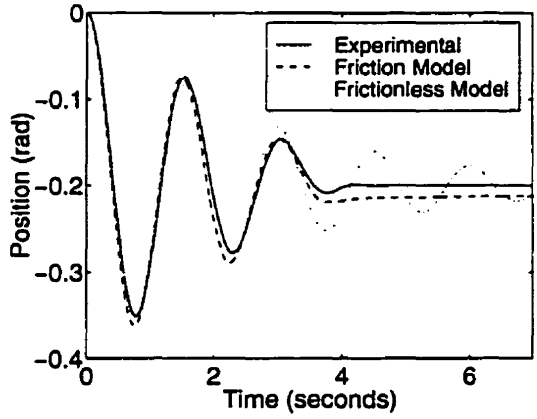
Figure 5.8 illustrates three different closed-loop step responses used to evaluate the friction in the axial joint. In each figure, the solid curve depicts the experimental data and the dashed curve represents the simulated system with friction. The dotted curve corresponds to a linear model without Coulomb friction. To compensate for the lack of Coulomb friction, the viscous friction has been increased. The linear model employed for the simulation is:

$$P(s) = \frac{1}{0.204s^2 + 0.150s}$$

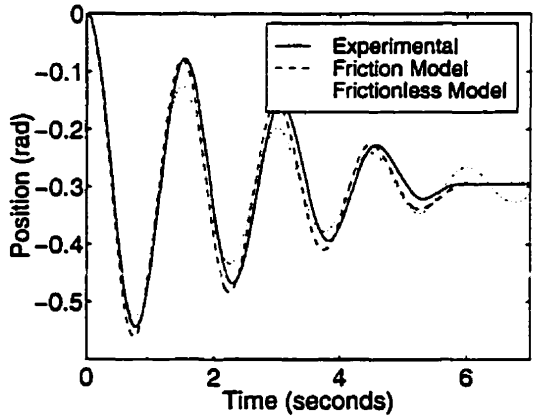
Note that the inertia is the same in all cases. The magnitude of the reference signal



(a) Reference of -0.1 rad



(b) Reference of -0.2 rad



(c) Reference of -0.3 rad

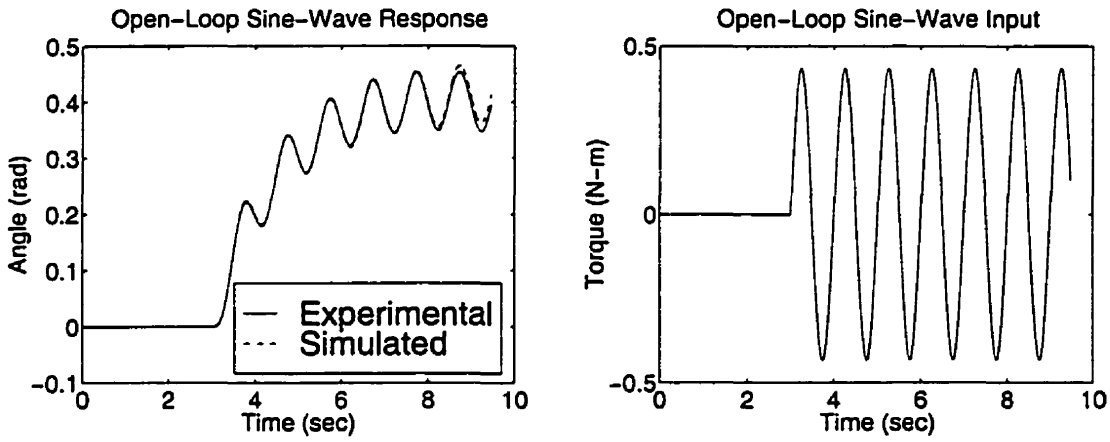
Figure 5.8: Underdamped Closed-Loop Step Responses

is different in each figure. Note that 0.1 rad corresponds to a 3 cm displacement of the end-effector.

If the plant were perfectly linear, then the plots would look identical and only the scale of the y-axis would change. However, friction clearly plays a role, since fewer oscillations occur for smaller reference inputs. Also, while the envelope of the linear model is exponential, friction causes the experimental system and friction model to decay faster than an exponential. The model that includes Coulomb friction clearly performs better than the linear model.

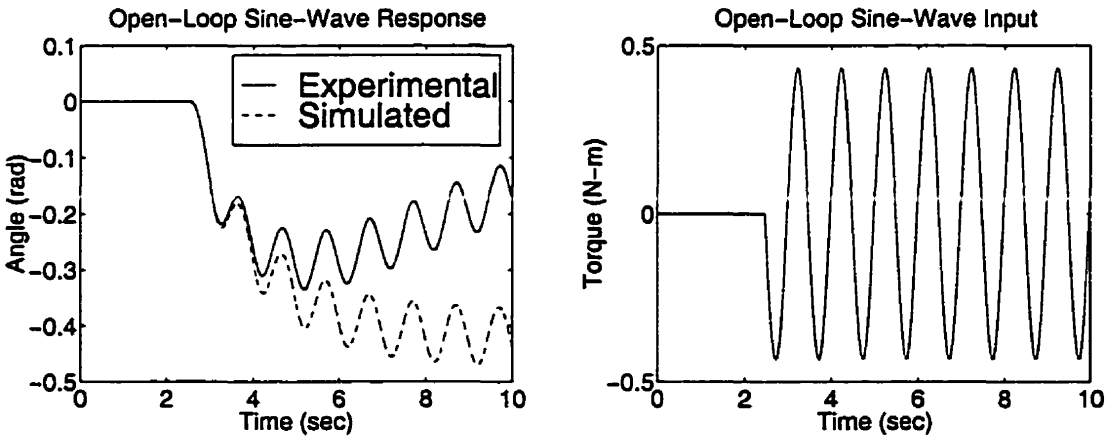
The open-loop responses of the system to two different sinusoidal inputs are illustrated in Figures 5.9 and 5.10. The sinusoid in both cases has the same magnitude – the only difference is in the initial conditions. The torque input for Figure 5.10 is 180° out of phase from the torque input in Figure 5.9. In each plot, the solid curve represents the experimental data. The dashed curve denotes the simulated response using the friction model and the same torque input.

The simulated response agrees very well with the actual experimental data in Figure 5.9. As expected, the simulated response for Figure 5.10 is just the negative of the the first response, since only the phase of the input has changed. However, the experimental data in Figure 5.10 is not symmetric. It appears that the ribbon cable transmitting the strain gauge signals to the robot base imparts a small torque on the axial joint. Since the equilibrium position of the ribbon cable "spring" does not correspond to the home position of the robot, this small torque results in a slightly asymmetric response i.e., the step response to a negative step input is not the same as the response to a positive step. Since this ribbon cable effect is unmodelled, it also contributes to the difference between the simulated and experimental results, both in the open-loop sinusoid responses and in the closed-loop step responses.



(a) Axial Joint Angle (b) Torque Input

Figure 5.9: Positive Open-Loop Sinusoid Response



(a) Axial Joint Angle (b) Torque Input

Figure 5.10: Negative Open-Loop Sinusoid Response

5.4 Vibration

A model of the experimental five-bar robot would be incomplete without the dynamics of the flexible link. The link consists of $\frac{1}{2}$" (1.27 cm) square aluminum tubing with a wall thickness of $\frac{50}{1000}$" (0.127 cm). The length of the beam, not including the area which is clamped, is 43.3 cm.

5.4.1 Natural Frequencies

The natural frequencies of vibration of the flexible link were determined by tapping the end-effector and observing the resulting vibrations in the frequency domain.

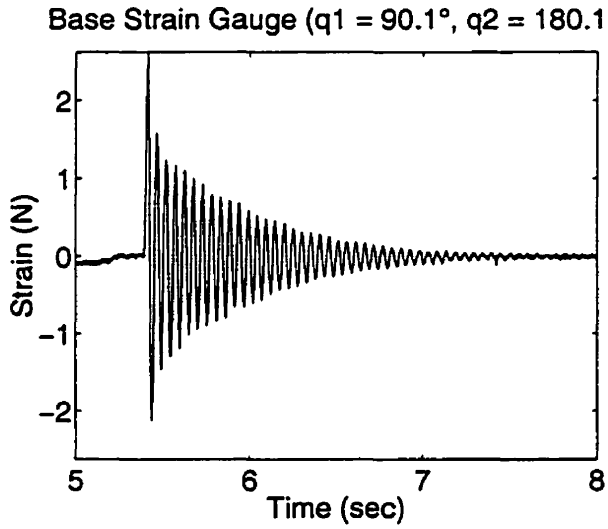
Figure 5.11(a) illustrates the response of the robot in the horizontal direction when the robot is in the home position i.e., $q_1 = 90°$ and $q_2 = 180°$. The corresponding frequency content of these vibrations is depicted in Figure 5.11(b). The frequency labelled 1.4 Hz may be ignored - it is the result of manually moving the robot into position for the test.

The frequencies of vibration for the horizontal direction were determined for a variety of robot configurations. The resulting frequencies are enumerated in Table 5.6.

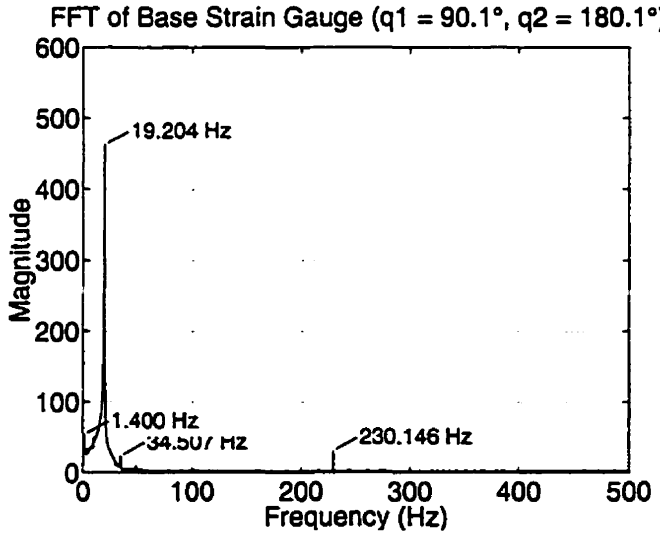
Both the axial and vertical directions exhibited only one measurable frequency of vibration. The natural frequency in the axial direction is 133 Hz and in the vertical direction is 53 Hz.

5.4.2 Analysis

Consider horizontal vibrations of the flexible link. Since the hub inertia is quite large, assume that the link may be modelled as a clamped-free cantilever beam.



(a) Time Response



(b) Frequency Content

Figure 5.11: Horizontal Vibrations

q_1 (deg)	q_2 (deg)	f_1 (Hz)	f_2 (Hz)	q_1 (deg)	q_2 (deg)	f_1 (Hz)	f_2 (Hz)
90.1	180	19.2	230	88.8	161	19.5	228
56.0	184	18.7	229	91.6	206	18.7	229
107	184	19.6	228				

Table 5.6: Dependence of Horizontal Modes on Robot Configuration

Since the modal frequencies of a pinned-free beam are higher than those of a clamped-free beam, this assumption should result in the natural frequencies being slightly underestimated. This point is important and should be remembered for subsequent analysis of the actual system.

The mode shapes for the clamped-free beam are derived in Appendix B. In particular, the characteristic equation for the modal frequencies is defined by equation (B.9). The first solution of this equation is $pl = 1.875$. The normalized frequency p is related to the natural frequency of the beam by (B.6) and (B.7). Combining these relationships:

$$\omega_n = (pl)^2 \sqrt{\frac{EI}{\rho A l^4}}$$

whence the frequency of the first mode is approximately:

$$\omega_1 = 3.516 \sqrt{\frac{EI}{\rho A l^4}} \tag{5.10}$$

Let h denote the height of the beam, l its length and d the wall thickness. Then the moment area of inertia is:

$$I = \tfrac{1}{12}h^4 - \tfrac{1}{12}(h-2d)^4 = 1.28 \times 10^{-9} \text{ m}^4$$

and the cross-sectional area is

$$A = h^2 - (h - 2d)^2 = 5.81 \times 10^{-5}\ m^2$$

For aluminum, $E = 69$ GPa and $\rho = 2800$ kg/m^3. Substituting these values into (5.10) yields $\omega_1 = 436$ rad/s or 69 Hz. The high value of the first natural frequency indicates the high stiffness of the beam. In other words, while the beam is sufficiently flexible to act as a force sensor, it is actually fairly rigid. Yet, the experiments in section 5.4.1 indicate that the actual frequencies for horizontal vibrations are lower.

Now, the mass of the flexible beam is $\rho l A = 71$ g. The mass of the axial sensor unit at the end of the beam is 43 g. Hence, the payload mass represents 60% of the beam mass, so while the sensor assembly is small, it does actually contribute significantly to a lowering of the natural frequency of the beam.

Hence, consider a more accurate model of the beam which considers the hub inertia and sensor "payload mass". Let:

J_0 = hub inertia for base joint, q_0 $(kg\ m^2)$

M_p = payload mass (kg)

L = length of beam (m)

ρ = linear mass density of beam (kg/m)

E = Young's modulus of beam (N/m^2)

I = area moment of inertia of beam (m^4)

ω = natural frequency (rad/s)

and define:

$$\beta = \frac{\rho\omega^2}{EI} \qquad \Omega_0 = \frac{J_0}{\rho} \qquad \Omega_p = \frac{M_p}{\rho}$$

From the work of Rossi [91], the characteristic equation of a flexible beam with hub inertia and point payload mass is:

$$\begin{aligned}\cos(\beta L)\sinh(\beta L) - \sin(\beta L)\cosh(\beta L) - 2\Omega_p\beta\sin(\beta L)\sinh(\beta L) \\ - \Omega_0\beta^3(1+\cos(\beta L)cosh(\beta L)) \\ - \beta^4\Omega_p\Omega_0(\cos(\beta L)\sinh(\beta L) - \sin(\beta L)\cosh(\beta L)) = 0 \quad (5.11)\end{aligned}$$

For the experimental apparatus,

$$J_0 = 0.677 \text{ kg-m}^2$$

$$M_p = 0.043 \text{ kg}$$

$$L = 0.449 \text{ m}$$

$$\rho = 0.163 \text{ kg/m}$$

$$E = 69 \times 10^9 \text{ N/m}^2$$

$$I = 1.28 \times 10^{-9} \text{ m}^4$$

Hence,

$$\Omega_0 = 4.16 \text{ m}^3$$

$$\Omega_p = 0.26 \text{ m}^3$$

Solving the characteristic equation, (5.11), with these numbers yields the first fundamental frequency, ω, of:

$$\omega = 223 \text{ rad/s} \quad \text{or} \quad 35.5 \text{ Hz}$$

This frequency is quite a bit lower than the frequency predicted by a simple clamped-free beam model. The impact of the payload mass is evident. It is also

closer to the observed frequency of approximately 19 Hz. However, it is not close enough to conclude that the beam is the sole source of vibration in the system.

Since the hub inertia is so large, errors in the hub inertia have little effect on the predicted frequency of vibration. Increasing the hub inertia by a factor of 10^6 reduces the vibrational frequency by less than 0.4 Hz. Hence, errors in hub inertia cannot explain the discrepancy between the predicted value and empirical observation.

While the predicted frequency is more sensitive to errors in the length of the beam, the length of the beam is very well known. Even adding 4 cm to the beam length only reduces the predicted frequency to 31 Hz and the length is known to far greater accuracy than to within 4 cm.

Thus, the inescapable conclusion is that the flexible link is not the only source of vibration in the system. *The robot structure itself must be flexible.* Given that link 2 and link 3 are constructed of the same aluminum tubing as the so-called "flexible" link 5, this conclusion is hardly surprising. Since link 2 is only 9 cm in length, most of the flexibility in the robot parallelogram is likely to be contributed by the 30 cm link 3, particularly given the 235 g counterbalance at one end of the link. Indeed, small vibrations of the counterbalance are visible during the horizontal vibration experiments.

Various experiments were performed to try to discern the source of vibration for modelling purposes. For example, if a single link was primarily responsible for the vibration, then mounting a mass at an appropriate location on the robot structure would change the frequency of vibration in a predictable way. However, experiments such as this failed to pinpoint a particular source of vibration. Hence, further research into the source of oscillation, perhaps through finite element modelling, is

necessary.

However, even with another source of vibration in the system, the natural frequencies of the beam should still appear in the frequency spectrum. Close examination of Figure 5.11(b) reveals a very small peak at 34.5 Hz. This measured frequency is very close to the predicted frequency of 35.5 Hz derived above. The percentage error is only 2.7%.

Unfortunately, the presence of vibration in the robot structure itself prevents the general model derived in chapter 3 from being applied. The model of chapter 3 only applies to flexibility in the last link and not the whole structure. A proper model of the structural vibrations would require finite element methods or similar techniques applied to the entire structure. However, it should be possible to make controller designs robust to vibrations in the robot structure by minimizing the closed-loop sensitivity at the natural frequencies of the robot. Note also that the flexibility in the robot structure does not invalidate the mechatronics approach of this work. Better construction and more substantial tubing for links 1 through 4 would undoubtedly eliminate the structural vibration problem. Sturdier construction does not negate the ability to gravity balance and dynamically decouple the robot.

Chapter 6

Controller Design

Haptic devices are designed to emulate the physical characters of some physical object or environment by providing force feedback to a human operator. One perspective on the problem is that the goal of the controller is to present a specific mechanical impedance to the human operator. A virtual wall designed to emulate aluminum, for example, would need to provide a stiffness[1] of approximately 70 GPa. Hence, it is not surprising that implementation of a virtual wall is one of the more challenging problems in haptic interface control. This chapter investigates a variety of control strategies for achieving a target impedance. The performance with respect to virtual walls is used as a basis of comparison.

6.1 Implicit Force Control

The simplest closed-loop control strategy for achieving a desired target impedance is the implicit force control scheme depicted in Figure 6.1. The operator P denotes the

[1]Refer to section 2.2.4 for background on virtual wall implementations

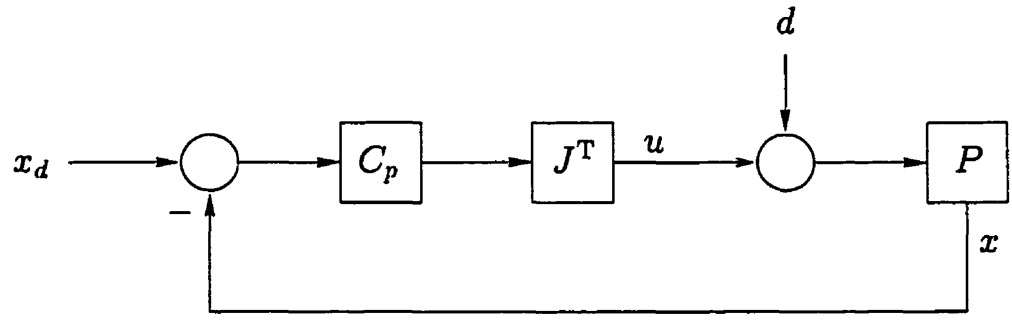


Figure 6.1: General Implicit Force Control

plant. Interaction with an environment is included only as a disturbance, d, in the diagram because the human operator acts as the "environment". It was decided to model the operator as a purely external force because the user actively manipulates the haptic end-effector. Salcudean [92] also includes a passive component for the human operator, but this approach is not investigated here. Implicit force control offers the advantage of not requiring a force sensor, as discussed in section 2.1.4. While implicit force control demands a plant that is backdriveable, position sensors such as optical encoders generally produce much cleaner signals than force sensors and so control based solely on position might be expected to be less susceptible to high-frequency vibration. Hence, implicit force control was investigated first as a benchmark control strategy to which more sophisticated controllers could be compared.

In the general strategy of Figure 6.1, the Jacobian, J, maps the force computed by the controller, C_P, to a torque suitable for input to the plant. For the sake for simplicity, the control strategies of this chapter are implemented in joint-space, so that the non-linear Jacobian is not necessary i.e., $J = I$. Since virtual walls already include the nonlinearity of a unilateral constraint, it would be harder to isolate the

source of limit cycles or other performance considerations if further non-linearities, such as the Jacobian, were also included in the control system.

From section 2.1.4, the impedance produced by the implicit force control scheme of Figure 6.1 at "equilibrium" is $Z_0(s) = s^{-1}C_P(s)$. Thus, the stiffness at "equilibrium" is $K_0(s) = C_P(s)$. Hence, a particular stiffness is achieved at "equilibrium" simply by setting the controller equal to the desired stiffness.

A wall might be expected to behave as a damped spring. Let K_p denote the proportional gain (Hooke's Law) of the spring and let K_d be the viscous damping coefficient. Then the stiffness of the damped spring may be represented by the transfer function:

$$K(s) = K_p + K_d s \tag{6.1}$$

Hence, according to section 2.1.4, the corresponding controller is simply $K(s)$.

However, $K(s)$ is improper and can not be implemented in practice. Hence, for the sake of implementation, let the derivative term be band-limited by a first-order low-pass filter with cutoff frequency, ω_0. Thus,

$$K(s) = K_p + K_d s \frac{\omega_0}{s + \omega_0} \tag{6.2}$$

The term $s\frac{\omega_0}{s+\omega_0}$ simply yields a filtered version of the angular velocity. Since only the position measurements are available, the velocity can be estimated using a first-order Euler difference equation. Due to quantization, the derivative can be noisy and the filtering is necessary. Note that at the very low angular velocities involved, tachometer readings would also be very noisy.

The response of the axial joint for $K_p = 400$ and $K_d = 0$, a light grip and a sampling period of 1 ms is depicted in Figure 6.2. The output torque is illustrated in

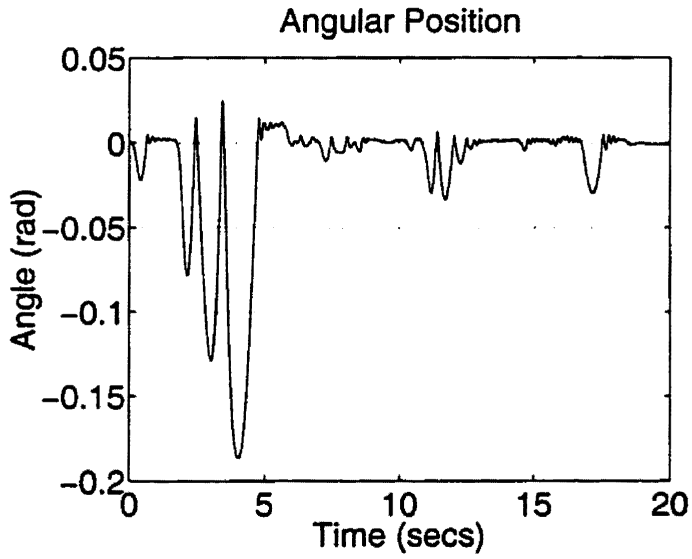


Figure 6.2: Virtual Wall by Implicit Force Control: Angular Position

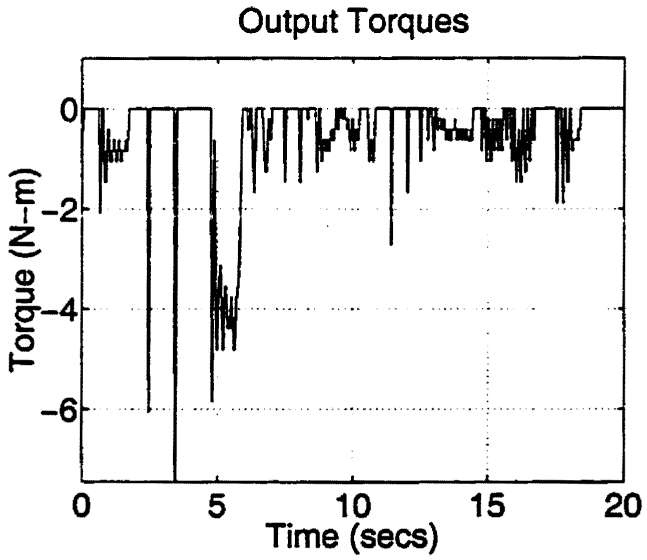


Figure 6.3: Virtual Wall by Implicit Force Control: Output Torques

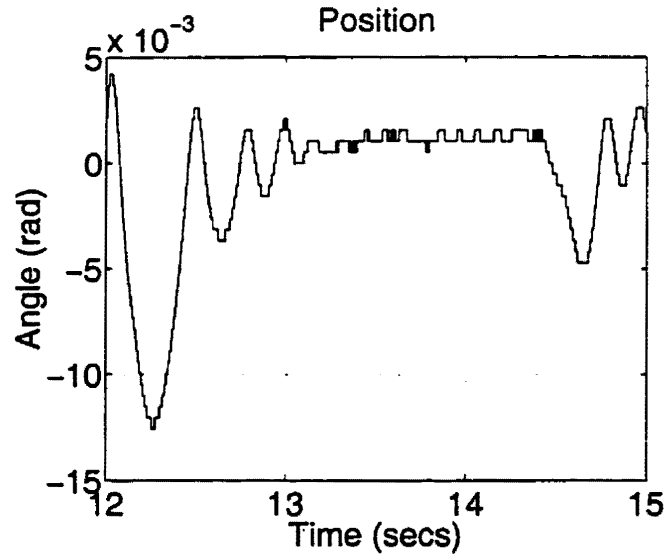


Figure 6.4: Magnified View of Angular Position

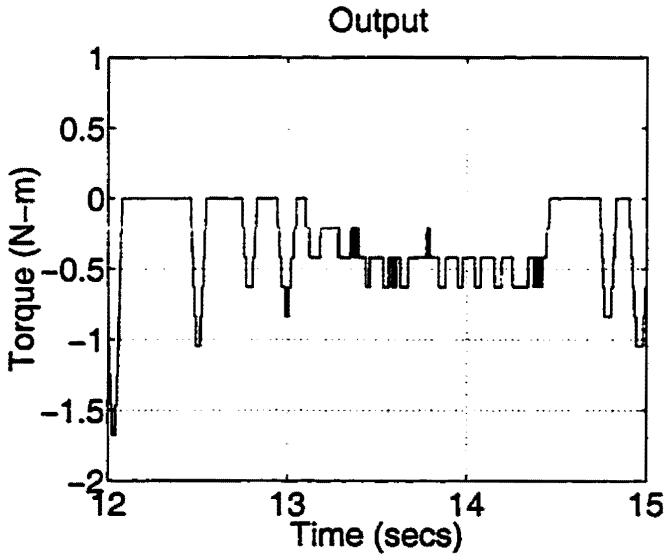


Figure 6.5: Magnified View of Output Torques

Figure 6.3. Note that the torque graph has been truncated at the negative saturation torque for the motor. For this large proportional gain, there is a small amount of vibration when the robot is in contact with the virtual wall. This vibration can be observed more readily in the magnified view of Figure 6.4. Figure 6.5 shows a magnified view of the corresponding output torques. The magnified views focus on a particular impact, where the end-effector is moved into the wall and then held gently against the wall. As presented in section 2.2.3, gentle pressure against the virtual wall with a light grip on the robot end-effector has been shown to be more destabilizing than firm pressure and a tight grip. This distinction is also observed in practice with the experimental apparatus employed for this work. Hence, only the situations where gentle pressure is applied are examined in detail.

Since no viscous damping term is present, impacts with the wall feel much like contacting a hard rubber surface using a rubber mallet. The effect is distinct, but does not have the "crisp" feeling one might expect. Neglecting the vibration, even the sound of impact resembles contact with hard rubber.

The quantization of the encoder measurements is readily apparent in the magnified view, as is the resulting quantization of the output signal. Note that the force being exerted by the user on the virtual wall is rather slight at this point. However, the effect of quantization is audibly noticeable as a buzzing sound and the oscillations can be felt on the fingertips. Low-pass filtering the encoder signals to smooth out the quantized signal results in low-frequency oscillations induced by the delay inherent in the filtered position measurement. If higher cutoff frequencies are used, the buzz is not eliminated. Hence, the proportional gain of 400 N-m/rad appears to be slightly beyond the maximum gain for acceptable performance.

Addition of a viscous damping term is ineffective. If the damping term is high enough to be perceptible, it causes the oscillations to become completely unaccept-

able. Increasing the proportional gain also results in oscillations. At a proportional gain K_p of 600 N-m/rad, the vibrations are so severe that they trigger one of the hard limit switches and disable the robot power.

6.2 Implicit Impedance Matching

In section 2.1.4, the impedance produced by the implicit force control scheme of Figure 6.1 at "equilibrium" is derived as $Z_0(s) = s^{-1}C_P(s)$. However, suppose the control is done in joint space so that the Jacobian transformation in Figure 6.1 is not necessary. In this case, the system is linear (assuming a linear plant). Hence, the closed-loop transfer function, $T(s)$, from the user input, d, to the plant output x is:

$$T(s) = \frac{x(s)}{d(s)} = (I + P(s)C(s))^{-1}P(s)$$

$T(s)$ represents the compliance of the closed-loop system, and is the inverse of the stiffness. Of interest is designing a controller $C(s)$ such that the closed-loop compliance of the system is identically equal to a desired compliance. Let $G(s)$ denote the desired compliance. For the actual compliance to equal the desired compliance:

$$\begin{aligned} G(s) &= (I + P(s)C(s))^{-1}P(s) \\ \therefore (I + P(s)C(s))G(s) &= P(s) \\ \therefore C(s) &= G(s)^{-1} - P(s)^{-1} \end{aligned} \tag{6.3}$$

But $G(s)^{-1}$ is simply the desired stiffness, while $P(s)^{-1}$ is the plant stiffness. Hence, the controller $C(s)$ is again determined by the desired stiffness, but in this case the

plant stiffness also enters into the controller equation. It is assumed that unstable pole-zero cancellations do not occur.

Of course, if the plant transfer function is strictly proper, then $P(s)^{-1}$ will be improper and the controller may not be implemented. Note that it is not possible to reduce the relative degree of the closed-loop system by the addition of linear feedback, so an inner control loop cannot be employed to make the plant non-strictly proper. However, a proper controller that approximates $G(s)^{-1} - P(s)^{-1}$ over a suitable bandwidth may be implemented instead.

Consider the one degree-of-freedom apparatus presented in section 4.1. From (4.1), the form of the plant transfer function is:

$$P(s) = \frac{1}{as^2 + bs}$$

Suppose the desired impedance is that of a mass-spring-damper i.e., for some desired mass, M, viscous damping, B, and spring constant, K, the desired compliance is:

$$G(s) = \frac{1}{Ms^2 + Bs + K}$$

The resulting controller from (6.3) is:

$$C(s) = (M - a)s^2 + (B - b)s + K \tag{6.4}$$

The inertia of a system is a difficult quantity for a human operator to judge. To implement a virtual wall, a spring-damper model is often used, as in the work of Colgate and Brown [11]. The mass of the wall is not such a concern. Thus, consider allowing the mass, M, of the desired system match the inertia of the plant, a, exactly. Then, the controller is simply the proportional-derivative controller:

$$C(s) = (B - b)s + K$$

Using this controller, the system will behave exactly like a mass-spring-damper system with corresponding coefficients of a, B and K respectively (provided friction is cancelled). Note that for free motion, $K = B = 0$, yielding the proportional-derivative control law:

$$C(s) = -bs$$

i.e., to achieve "free" motion, the controller must provide negative damping to compensate for the inherent damping of the plant (assuming M is small). To avoid internal stability problems, the controller may be implemented as:

$$C(s) = \frac{-b\omega_0(s+\epsilon)}{s+\omega_0}$$

where $\frac{\omega_0}{s+\omega_0}$ provides low-pass filtering with a cutoff frequency of ω_0 and ϵ is a small quantity used to avoid the pole-zero cancellation at the origin.

To test the validity of the simple analysis above, the proportional-derivative-derivative controller (6.4) was implemented on the experimental apparatus. The derivative and double derivative terms were calculated using standard Euler backward difference formulae. The controller was implemented digitally on an IBM PC with an Intel 486 CPU running at 66 MHz and the Windows 3.1 operating system. Sampling times of less than 400 microseconds were achieved. Different sampling rates were attempted to examine the effect of sampling period on performance. The results for the one-dimensional apparatus included here are based on a 5 ms sampling period, except where indicated.

The motor was driven by a PAO103 voltage-to-current amplifier, provided by Quanser Consulting. To compensate for friction, a preload function, F_f, was used. Figure 6.6 illustrates the preload function, F_f, as a function of velocity. The deadband region was used to filter out noise in the velocity estimates.

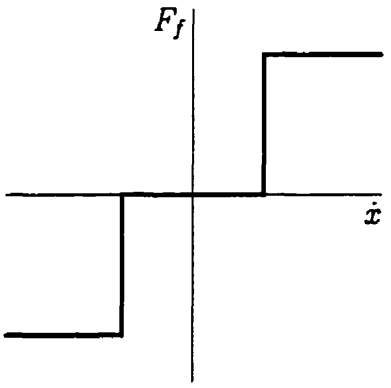
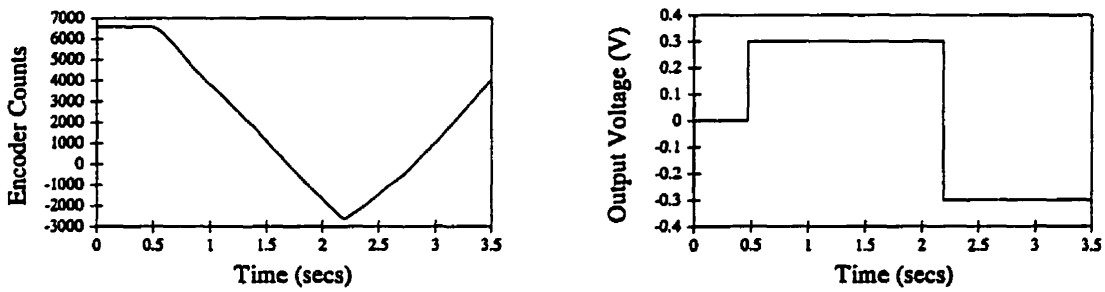


Figure 6.6: Preload Function with Deadband

Free Motion

Figure 6.7 shows the response of the system to a user-applied impulse, using friction compensation alone. Note that since no force sensors were implemented, it was not possible to measure the force exerted by the user during the impulse. Nor was it possible to measure the force exerted by the user for the purposes of control. However, similar results were obtained for various user-applied impulses.

The impulse was applied at approximately 0.5 seconds. Prior to the impulse, the carriage was at rest. The end-effector changed direction when it gently collided with the rubber stopper at the end of its travel. Ideally, the position versus time curve should be linear, indicating constant velocity, before and after the collision with the robber stopper. Observing the slope of the position versus time curve, it is evident that the velocity is not perfectly constant during the motion. The variation in velocity is caused by friction differences along the length of the travel, as well as the motor shaft being slightly bent. (The apparatus is a modification of an old, discarded printer.) There is also a small amount of viscous damping in the



(a) Position vs. Time

(b) Control Signal

Figure 6.7: Friction Compensation for Free Motion

system, possibly due to the lubricant. The deadband of 200 counts/s in the preload function is so small that it does not appear in Figure 6.7(b).

Qualitatively, the carriage was quite easy to move with the preload function enabled. The friction in the system with the power amplifier disabled is substantial - the carriage does not glide freely along the length of the apparatus without active compensation, even for very large impulses. Thus, the use of a preload function greatly improves the "feel" of the system.

However, use of a preload function is not without problems. While a continuous preload function can be implemented for systems with non-uniform friction, the greatest difficulty arises when starting to move the carriage from a standstill. Since no force sensors are used on the system, the direction the user is applying a force cannot be detected. Instead, the velocity of the system is used to determine the sign of the preload function. Unfortunately, when the end-effector is at rest, the velocity is zero. Thus, the preload function cannot compensate for static friction. It can only compensate for kinetic friction once the carriage begins moving. As a

result, it is very difficult to position the manipulator accurately. While the carriage is at rest, the control system cannot compensate. Hence, to get the carriage to move, the operator must apply a force equal in magnitude to the static friction. Once the end-effector begins to move, the active compensation virtually cancels the kinetic friction and no force is required to move the end-effector. The sudden drop in the force required to move the carriage causes it to "jump", since the operator cannot compensate quickly enough to stop applying a force.

The problem of static vs. kinetic friction is less of a problem, of course, when such implicit force control is used to control actuator stiffness for a pre-defined trajectory. In this case, the actuator is rarely at rest anyway, and even if the trajectory does bring the robot to a stop, the trajectory is primarily pre-planned so that direction of motion as the end-effector begins to move is known. Thus, it may be possible to compensate for static friction using implicit force control in these circumstances.

If fine positioning is required of the experimental haptic interface, alternate preload functions may be used. For example, fine positioning would likely be difficult on a frictionless surface when the operator has nothing upon which to brace their wrist or hand. Thus, it may not only be acceptable, but perhaps preferable, to allow some friction at low speeds while compensating fully at high speeds.

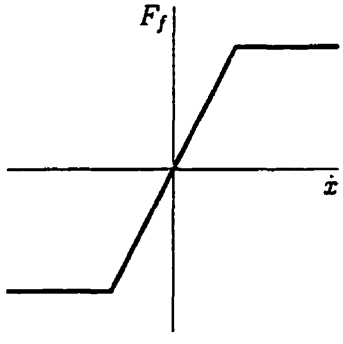
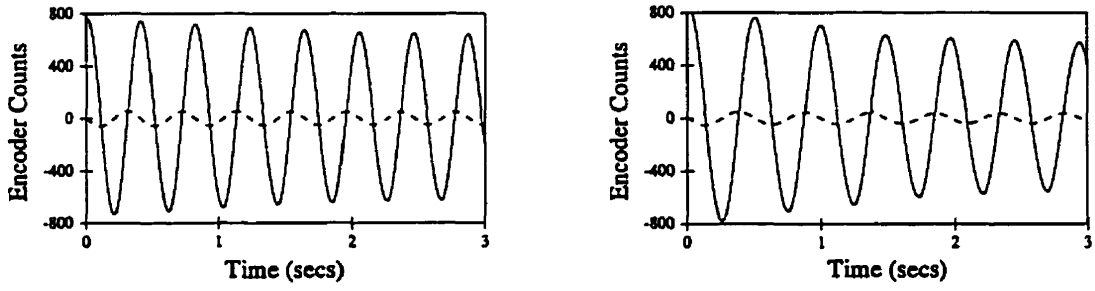


Figure 6.8: Preload

Consider the preload function illustrated in Figure 6.8. As the velocity increases from zero, the perceived friction will decrease from the static friction magnitude to zero, once a certain threshold velocity is reached. Up to the threshold, the preload function is essentially identical to nega-



(a) Spring Only

(b) Spring and Inertia

Figure 6.9: Simulation of a Spring

tive damping. While this technique met with greater success for fine positioning, the slope of the linear portion of the preload function had to be quite shallow, so that the velocity threshold was rarely reached by the operator. This observation suggests that a more non-linear preload function may better match human capabilities. Note that the use of force sensors eliminates the difficulty because then even the static friction can be measured as the operator applies a force to move the manipulator.

Simulation of a Spring

Figure 6.9(a) depicts the response of the system to an impulse when the desired transfer function, $G(s)$, is set to that of a linear spring i.e., $G(s) = K^{-1}$. The solid line represents the position in encoder counts, while the dotted line represents the velocity in encoder counts per sampling period. The sampling period was 5 ms.

For simplicity, the carriage inertia and viscous damping were left uncompensated, so that the controller was simply $C(s) = K = 0.001$ V/count. Adding a

small amount of negative damping to the controller eliminates the slight decay in the response seen in Figure 6.9(a). It must be noted, however, that too much negative damping (i.e. $B < 0$) leads to instability. With damping added to the controller, the transfer function from the operator's input to the carriage position is $G(s) = (Bs + K)^{-1}$. If $B < 0$ then $G(s)$ has a pole in the right-half plane, resulting in instability. With $B = 0$, the system behaves as an undamped spring i.e., $G(s) = K^{-1}$ and the system is marginally stable.

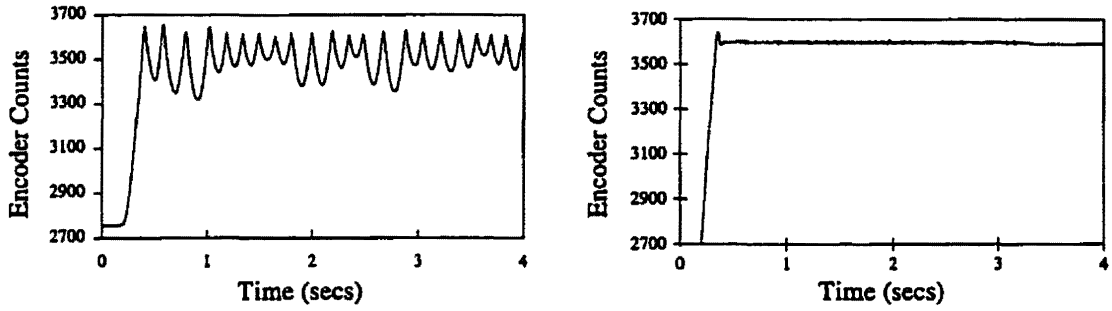
Figure 6.9(b) shows the response of the system to an impulse when the same controller is used, except an "inertial" term of $M = (a + 2 \times 10^{-6})$ V/(counts/s^2) is added, yielding:

$$C(s) = 0.001 + (2 \times 10^{-6})s^2$$

Addition of the "inertial" term to the controller results in a desired transfer function, $G(s)$, with a higher mass than the actual mass of the carriage. Thus, one would expect the frequency of oscillation to be lower than the previous case. A comparison of Figure 6.9(b) with Figure 6.9(a) confirms this hypothesis – the controller with added "inertia" results in a system which feels heavier. Although not shown here, adding a negative "inertial" term to the controller results in a higher frequency of oscillation since the effective mass of the system is actually reduced. This mass-reduction effect was observed experimentally. Once again, however, it is important not to decrease the inertia too much (i.e. $M < 0$) or instability will result.

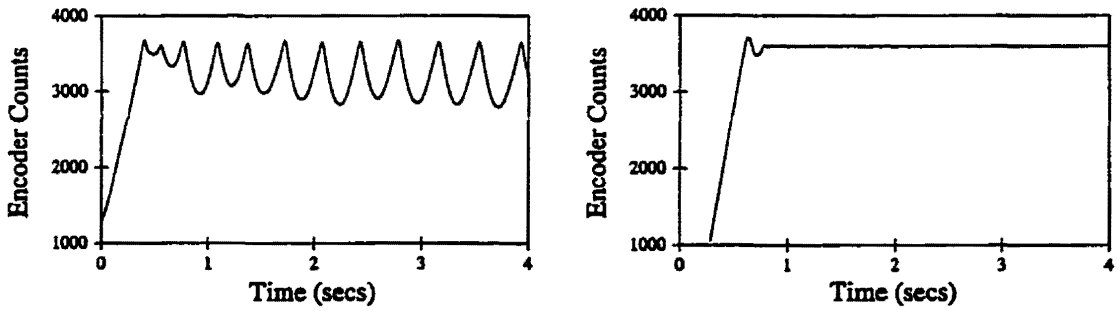
Simulation of a Wall

One of the most difficult effects to simulate is a virtual wall. Figures 6.10(a) through 6.11(b) illustrate the response of the system when the operator moved into contact



(a) Loose Grip (b) Tight Grip

Figure 6.10: Virtual Wall Using 5 ms Sampling Period



(a) Loose Grip (b) Tight Grip

Figure 6.11: Virtual Wall Using 10 ms Sampling Period

with a virtual wall. The wall was simulated by using a controller of the form:

$$C(s) = 0.2 + 0.001s$$

when the end-effector came "into contact" with the virtual wall. Outside the virtual wall, only free motion was implemented, using the simple preload function F_f with deadband already described in the previous subsection. Using this controller, the system behaves as a very strong compression spring with a small amount of viscous damping. Only a small viscous damping term was used so that the result would emphasize the difference in response caused by changing the sampling period and the operator's grip.

Figure 6.10(a) depicts the response when the operator uses a very loose one-finger grip and the controller employs a 5 ms sampling period. As reported in the experimental results of Colgate and Brown [7], oscillations occur when the end-effector comes into contact with the virtual wall. At a 5 ms sampling period, with a loose one-fingered grip, the amplitude of the oscillations is at most 350 encoder counts. At a 10 ms sampling period, the amplitude increases to at least 600 encoder counts, as shown in Figure 6.11(a). Note that no oscillations occur if the operator simply releases the end-effector. The user must push gently against the virtual wall to incite vibration. A similar increase in the amplitude of oscillations was observed for different operators when reducing the sampling period from 5 ms to 10 ms.

With a tight full-fingered grip, the amplitude of oscillation is significantly reduced, as observed by both Colgate and Brown [7], Lawrence and Chapel [6], and Hannaford and Anderson [93]. Figure 6.10(b) depicts the response of the system to a virtual wall for a tight full-fingered grip and a 5 ms sampling period. Figure 6.11(b) illustrates the response when the sampling period is increased to 10 ms. In Figure 6.10(b), the amplitude of oscillations is approximately 11 encoder counts,

while in Figure 6.11(b), the amplitude is approximately 15 encoder counts. Once again, a slower sampling period leads to larger oscillations, even with a tight grip.

Clearly, it is desirable to use the minimum sampling period possible. It is also evident, as concluded by Lawrence and Chapel [6], that the "loose-grip regime" determines the stability of the virtual wall implementation. While not included here, experiments performed at a 400 microsecond sampling period demonstrated greatly improved responses, even for the loose one-fingered grip.

The experiments of this section were performed using the one-dimensional apparatus. It is interesting to note that the limiting factor in this apparatus appears to be the sampling rate. Faster sampling rates yield less oscillatory responses. Yet for the five-bar robot, it is the quantization, not the sampling rate, that is the limiting factor. This observation is less surprising when two facts are considered. First, the one-dimensional apparatus has an encoder resolution of about 40 μm. The five-bar robot has a position resolution of only 160 μm for the axial joint. Hence, the one-dimensional apparatus has 4 times the positional accuracy and quantization might be expected to be less of an issue. Furthermore, the one-dimensional apparatus has far more friction than the five-bar robot. Like viscous damping, friction may be expected to improve the performance of virtual walls since it naturally damps out vibrations. Indeed, Colgate and Brown [7] assert that increasing the physical damping is the "least expensive and highest payoff measure available" for improving the performance of virtual walls. Friction, then, should yield similar "payoffs".

6.3 State Estimation

Unfortunately, implicit force control suffers from a number of drawbacks when position measurements alone are available, as demonstrated in the preceding sections.

If the desired stiffness includes some damping, then velocity estimates are required. Using a derivative to calculate velocity can be a noisy operation and employing filters can introduce undesirable lags into the system. For virtual walls, these phase lags may be destabilizing, particularly for very stiff walls.

Quantization is also an issue. Ironically, as the stiffness of the virtual wall is increased, the quantization problem becomes more severe. Higher stiffnesses lead to less wall penetration and fewer encoder counts being traversed. In the extreme case, for a very stiff wall, the user may only be able to penetrate the virtual wall to a depth of one encoder count. For a proportional implicit force controller, where the output torque is simply a large gain, k, times the position in encoder counts, there would be a large step in the output torque as the user entered the virtual wall. Given the sampled nature of the controller, it is not difficult to see how this would be destabilizing since the system would be non-conservative. This destabilizing effect is investigated by Gillespie and Cutkosky [24] (see section 2.2.5). The output torque is maintained by the digital-to-analog converter between sampling periods, so while the torque may force the user out of the wall, the output would not go to zero until the next sampling instant. Figure 6.5 illustrates the staircase control signal that can result from insufficient encoder resolution[2].

An obvious solution is to increase the encoder resolution. However, direct-drive robots, which are well-suited for haptic interfaces, have no gearing, and thus it is more difficult to obtain the requisite encoder resolution for very stiff walls.

An alternative is to use state estimation to predict the velocity *and position* of the plant. It may seem odd to estimate position when it is a measured quantity,

[2]The problem is not limited to encoders. Any analog signal is ultimately converted to a digital signal for computer control using a finite word-length analog-to-digital converter. Hence, all analog signals will be quantized.

but the state-estimator naturally produces a smooth, rather than quantized, output. Furthermore, the velocity estimate is improved because the plant model is incorporated into the estimator design. Also, by including the net torque acting on the plant as one of the estimator inputs, the lag problem introduced by filtering can be greatly reduced. The net torque is simply the difference between the torque exerted by the user through the robot end-effector, and the torque commanded by the controller. Hence, to compute the net torque, the torque applied by the user must be measured. In the five-bar apparatus employed for this research, the force applied to the end-effector is measured using the strain gauges mounted on the flexible link. The torque may then be deduced using the techniques of section 5.2.

This approach is similar in some respects to the work of Hacksel and Salcudean [94], who used a model-based observer to estimate the environment forces acting on the plant. The velocity is also estimated using an observer and corrections based on the estimated environment forces.

In this work, the forces are actually measured using the strain gauges mounted on the last link. The measured forces are then fed to a linear quadratic estimator to produce a velocity *and* position estimate since quantization of the position measurements proved to be a problem. Nonlinear correction terms are applied to the estimator to account for the kinetic friction observed in section 5.3.3.

Consider the axial joint for the five-bar apparatus. From Table 5.4, the identified plant model for the axial joint is:

$$P(s) = \frac{1}{Js^2 + Bs} = \frac{1}{0.204s^2 + 0.012s} \quad \text{rad/N-m} \tag{6.5}$$

Since a mechatronics approach was applied to the design of the manipulator, the plant dynamics are well-suited for control, being linear and low order. The linear dynamics allow a state estimator to be employed to estimate the position and

velocity of the plant. While some Coulomb friction is present, its magnitude is small because the robot motors are direct-drive. However, non-linear correction terms are added later to account for the friction as well. Now, let the plant model be represented in state-space form as:

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + Bu + G\mathbf{w} \\ q &= C\mathbf{x} + Du + v \end{aligned} \tag{6.6}$$

The signal $\mathbf{w} = (w_1 \quad w_2)^{\mathrm{T}}$ represents the process noise and v denotes the noise in the encoder measurements. Since the position and velocity are the states of interest, let the (A, B, C, D) state-space matrices be derived using the control canonical form, so that the states $\mathbf{x} = (x_1 \quad x_2)^{\mathrm{T}}$ are, indeed, position and velocity:

$$A \equiv \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{B}{J} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -0.0565 \end{bmatrix}$$

$$B \equiv \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} = \begin{bmatrix} 0 \\ 4.9016 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$D = 0$$

Define the noise covariances:

$$Q = E(\mathbf{w}\mathbf{w}^{\mathrm{T}}) \qquad R = E(vv^{\mathrm{T}})$$

Normally, for state-estimators, the input to the plant, u, is known, being generated by the controller. Hence, the linear quadratic estimator design is not structured specifically to account for noise in the input, u. However, in the case of the haptic interface, the input u is the net torque and is a function of both the control signal and the torque imparted by the user. This user "disturbance" is measured using

the strain gauges and thus is also susceptible to noise. To account for this input noise in the state-estimator design, let G in (6.6) be defined as follows:

$$G = \begin{bmatrix} 1 & 0 \\ 0 & B_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 4.9016 \end{bmatrix}$$

Substituting for G, (6.6) may be rewritten in expanded form as:

$$\begin{aligned} \dot{x_1} &= A_{11}x_1 + A_{12}x_2 + w_1 \\ \dot{x_2} &= A_{21}x_1 + A_{22}x_2 + B_2(u + w_2) \\ q &= C\mathbf{x} + Du + v \end{aligned}$$

Thus, w_2 may be interpreted as noise in the input torque u rather than process noise. It could also be interpreted as weighted process noise B_2w_2, so the linear quadratic estimator design procedure should minimize the error covariance:

$$J = \lim_{t\to\infty} E\left((\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^{\mathrm{T}}\right)$$

given Gaussian white noise in the torque measurements derived from the strain gauges, as well as some Gaussian process noise, and noise in the encoder readings.

Now, the covariance matrix, Q, may be expanded as follows:

$$Q = E(\mathbf{w}\mathbf{w}^{\mathrm{T}}) = E\left(\begin{bmatrix} w_1^2 & w_1w_2 \\ w_2w_1 & w_2^2 \end{bmatrix}\right)$$

Assuming the process noise and input noise are uncorrelated, let Q be diagonal. Thus, the (1,1) element of Q corresponds solely to the process noise, and the (2,2) element relates to input noise. Since the input torque is a function of the strain gauge readings, it may be expected to have a larger variance. Hence, the (2,2) element should be larger than the (1,1) element.

Disregarding the non-Gaussian nature of quantization errors, the encoder measurements have little noise and so the associated noise covariance, R, is expected

to be small. Thus, for the purposes of design, let:

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 41.6 \end{bmatrix} \quad \text{and} \quad R = 0.1$$

The resulting gain matrix, L, for the stationary Kalman filter is:

$$L = \begin{bmatrix} 14.4 \\ 99.2 \end{bmatrix}$$

The gain matrix may be calculated by solving an algebraic Riccati equation [86].

The estimator takes two inputs: the net input torque, u, and the encoder measurement, q. Hence, the state-space matrices for the estimator may be calculated as:

$$A_e = A - LC \qquad B_e = \begin{bmatrix} B - LD & L \end{bmatrix}$$

$$C_e = I \qquad D_e = 0$$

Substituting for the actual values of A, B, C, D and L, the estimator state-space representation is:

$$\left[\begin{array}{c|c} A_e & B_e \\ \hline C_e & D_e \end{array}\right] = \left[\begin{array}{cc|cc} -14.4 & 1.000 & 0 & 14.4 \\ -99.2 & -0.057 & 4.90 & 99.2 \\ \hline 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{array}\right]$$

The C_e matrix is set to the 2×2 identity matrix so that both states – the position and velocity – are available to the controller.

To verify the design, a sine-wave was fed to the axial joint motor of the actual experimental apparatus and the resulting joint position was measured and recorded over a period of 10 seconds. The sinusoidal torque input was also recorded. This torque was then input to the state estimator in simulation.

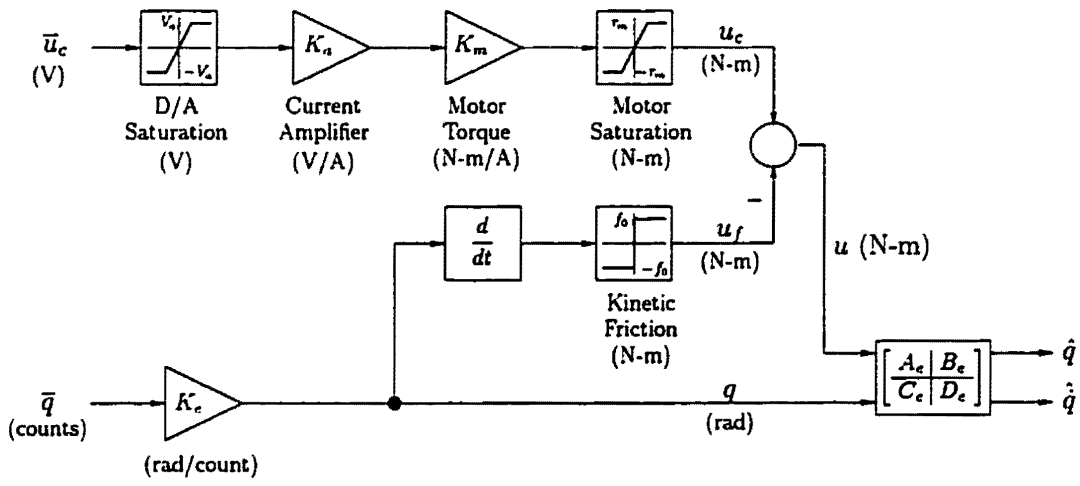


Figure 6.12: Block Diagram for State Estimator Simulation

A block diagram of the state estimator simulation is shown in Figure 6.12. The simulation is complicated by practical considerations. First, the measured position is in encoder counts and must be converted to radians for the state estimator, which was designed using SI units. The measured position in raw encoder counts is denoted $\overline{q}$ in the figure. Encoder counts are converted to radians using a factor K_e of $-\frac{\pi}{6000}$, based on Table 5.2.

Similarly, the interface to the analog domain is achieved using a digital-to-analog converter. Since these converters have a finite word length and limited output range, the output must be saturated in order not to exceed the capabilities of the digital-to-analog converter. The raw voltage output to the digital-to-analog converter is demarcated as $\overline{u}_c$ in the figure and is followed by a saturation operator to limit its range. The saturation of the digital-to-analog converter occurs at a voltage V_a of 5 V.

The output voltage is converted to a current to drive the motor by a pulse-

width modulated current amplifier with transconductance K_a, as seen in the figure. The current amplifier has a transconductance, K_a, of 3.87 A/V, as recorded in Table 5.1. The output current drives the motor and results in a motor torque of u_c. It is assumed that the electrical time constant of the motor is much faster than the mechanical time constant. Hence, the electrical dynamics of the motor are ignored and the motor is modelled by the motor torque constant and a saturation operator to represent the peak torque of the motor. The motors have a peak torque, τ_m of 36.8 N-m. The torque constant for the motor, K_m, is 0.373 N-m/A, from the motor specifications. Note that for the open-loop simulations, the saturation operators for the digital-to-analog converter and peak motor torque are not required, since the torque output is known to lie within the linear range. However, they are included here for consistency – all the simulations performed in this work include these saturation operators. The operators are important in the closed-loop simulations where saturation may, in fact, occur.

The torque produced by the motor is not the net torque operating on the plant. Friction reduces the torque output of the motor. The friction, u_f, is modelled as simple kinetic friction in the simulation. Its magnitude, f_0, is constant (except at zero velocity), but its sign depends on the sign of the robot velocity. Let $\mathrm{sgn}(\cdot)$ denote the signum function. Then the kinetic friction may be expressed as:

$$u_f = f_0 \,\mathrm{sgn}(\dot{q}) \tag{6.7}$$

The net torque, u, is computed as the control torque, u_c, minus the friction torque, u_f. Note that the direct-drive servomotor design of the mechanical system means that non-linearities caused by gearing, such as backlash, are not present. These non-linearities can be complicated to model. Here, due to the mechatronics approach to mechanical design, only the simple torque calculations outlined above

are required to compute the net torque acting on the plant. The force exerted on the end-effector by the human operator is incorporated into the net torque calculation later. It is not needed for the open-loop test described here.

The net torque u and position q are fed to the linear quadratic estimator to yield the estimated position and velocity, $\hat{q}$ and $\hat{\dot{q}}$ respectively. Since friction is not measured, but is included in the model, it is considered as part of the overall estimator. Hence, the estimator is non-linear due to the inclusion of a non-linear kinetic friction model. A friction magnitude, f_0, of 0.061 N-m is employed in the simulation, as discussed in section 5.3.3.

The position predicted by the state-estimator is compared to the measured position in Figure 6.13. The presence of the friction model noticeably enhanced the estimator performance. Since the measured and estimated curves are virtually indistinguishable, a magnified view of part of the curve is depicted in Figure 6.14. The quantization of the measured position is clearly visible in the magnified view, yet the estimated position is smooth. Furthermore, there is no appreciable phase lag in the estimate. Thus, if the assertion that quantization destabilizes virtual walls is correct, then the estimator should yield improved performance.

The other claim is that derivative calculations are noisy but the phase lag introduced by filtering the derivative also destabilizes virtual walls. Hence, it is instructive to compare the estimated velocity to the velocity calculated from the measured position. Figure 6.15 illustrates the estimated velocity superimposed on the calculated velocity. The velocity was calculated from the position measurements using the simple first-order Euler difference method:

$$\frac{dq}{dt} = \frac{q(t) - q(t-h)}{h}$$

where h is the sampling period of the controller. For these experiments, h was

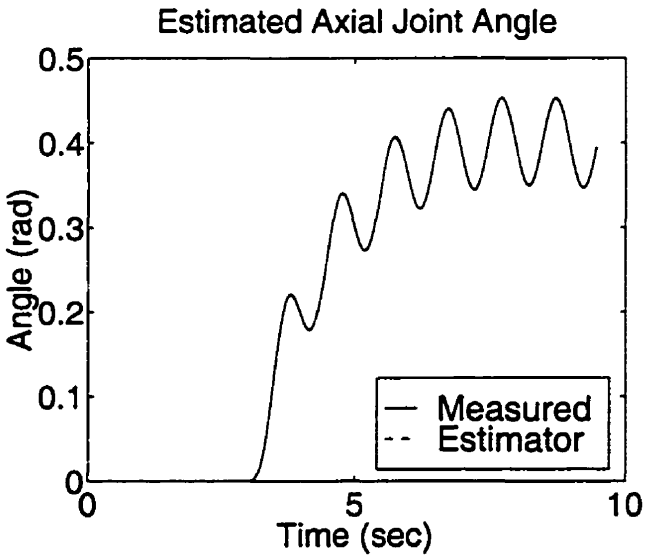


Figure 6.13: Non-linear Estimation of Axial Joint Angle

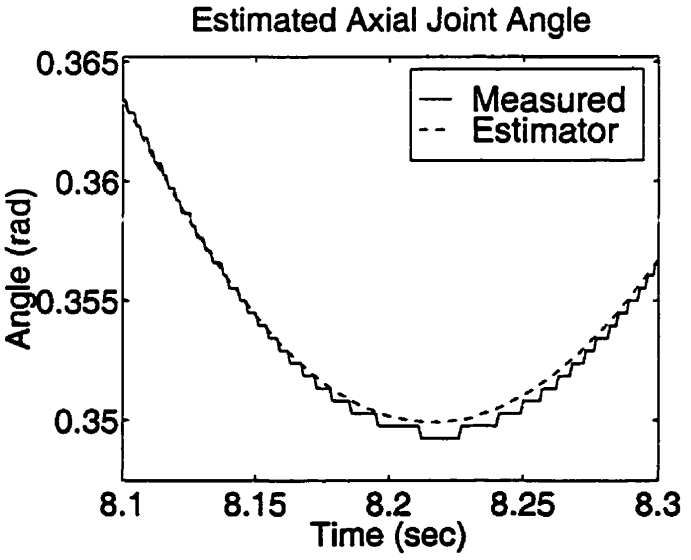


Figure 6.14: Non-linear Estimation of Axial Joint Angle - Magnified View

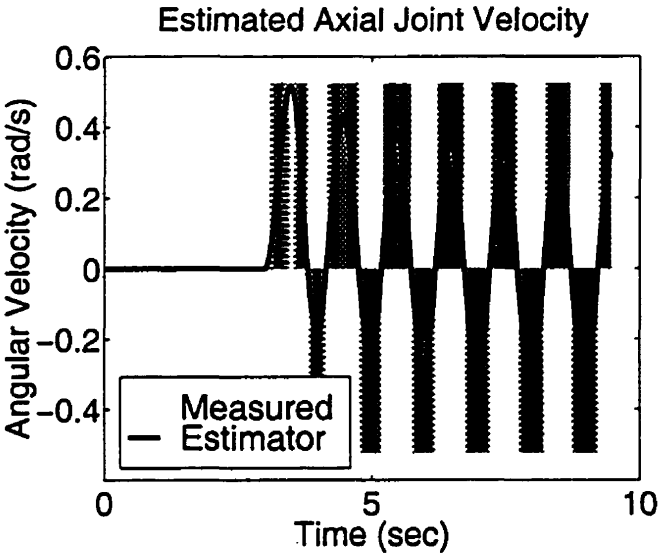


Figure 6.15: Non-linear Estimation of Axial Joint Velocity

1 millisecond. The calculated velocity is extremely noisy due to the fast sampling rate. As mentioned in section 2.2.4, there is a paradox when dealing with quantized signals: the faster the sampling rate, the noisier the calculated velocity. This observation is graphically illustrated in Figure 6.15.

Colgate and Brown [7] reported improved results when they digitally filtered the calculated velocity. This improvement is validated by Figure 6.16, which depicts the estimated velocity and the calculated velocity after filtering using a first order low-pass filter with a 20 rad cutoff frequency. A magnified view of this graph is shown in Figure 6.17. Note the phase delay in the filtered velocity. While 20 rad is a low cutoff frequency, it still does not yield a signal as smooth as the velocity produced by the state estimator. The cutoff frequency could be increased to reduce the phase lag, but the noise would increase proportionately. Although not shown here, the velocity estimate, $\hat{\dot{q}}$, was integrated to verify that the integral did indeed

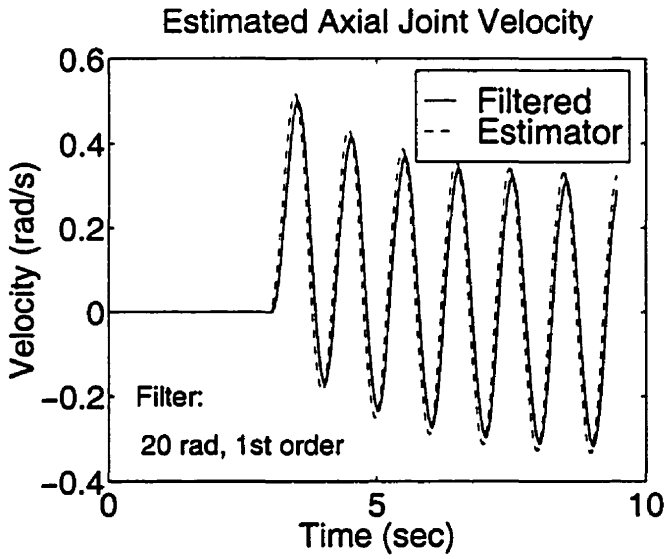


Figure 6.16: Filtered Velocity Measurement

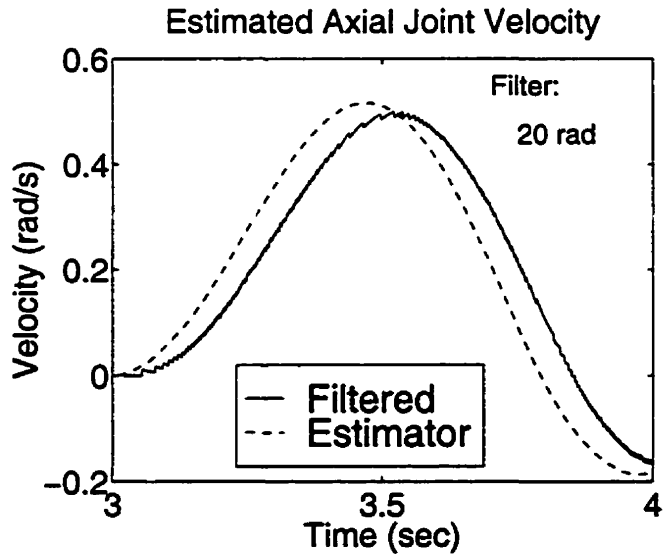


Figure 6.17: Filtered Velocity Measurement - Magnified View

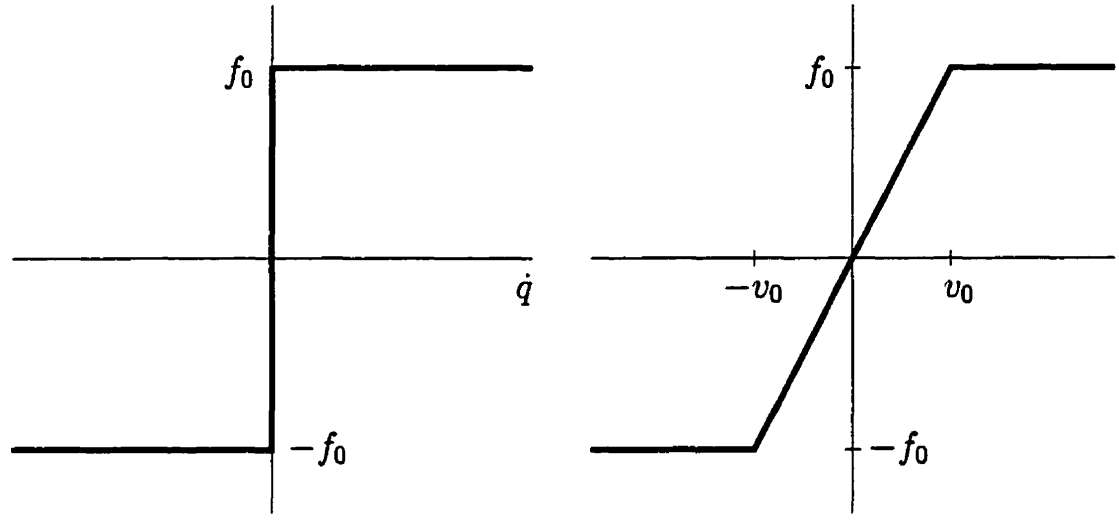
reproduce the position, q, with no apparent phase delay.

Similar experiments were performed for the other two joints. The results were comparable, so they are not included here. Note that other values of Q and R could have been chosen for the estimator design, although choosing Q too large or R too small leads to numerical problems in the estimator implementation.

While the above results demonstrate the effectiveness of the non-linear state estimation approach for generating smooth position and velocity estimates, the results would not be complete without experimental verification of the estimator employed in a closed-loop control system. However, there are difficulties in implementing the friction compensation depicted in Figure 6.12.

Consider the kinetic friction operator of Figure 6.18(a), which is given by (6.7). Due to the sampled nature of a digital controller, and the zero-order hold, this block cannot be used in the actual controller. Even for a very small velocity, (6.7) still produces an output torque of magnitude f_0. If this torque is sufficient to reverse the direction of motion of the robot before the next sampling instant, then the robot could oscillate back and forth in a sustained limit cycle. Also, the derivative operation used to calculate the velocity has already been shown to be a noisy operation, and filtering is undesirable.

To compensate for these two factors, the friction operator used in the actual experimental state estimation is the saturation-type operator depicted in Figure 6.18(b). For small velocities, the friction is modelled as viscous damping so that the limit cycle described above does not occur. The friction saturates at the value f_0, so that at velocities greater than v_0 the simple kinetic friction model is employed. The velocity input to the friction operator is derived from the state estimator to reduce noise.



(a) Coulomb Friction (b) Friction Approximation

Figure 6.18: Friction Compensation

The resulting compensator is depicted in Figure 6.19. The encoder counts, $\overline{q}_1$ and $\overline{q}_2$, are converted to radians using the conversion factors of Table 5.2:

$$K_{e1} = -\frac{\pi}{6000} \text{ rad/count} \qquad K_{e2} = \frac{\pi}{4000} \text{ rad/count}$$

The axial strain measurement, ε, is mapped to a force, F_x, on the end-effector via the constant of proportionality, $K_\varepsilon = 13.4$ N/V, from Table 5.3. The resulting force, F_x, is mapped to a torque on the axial joint, τ_u, via the nonlinear equation (5.2). τ_u represents the torque due to the user exerting a force on the robot end-effector.

The net torque exerted on the robot is a combination of the torque applied by the user, the torque commanded of the motors, and the friction. The friction is estimated using the nonlinear saturation-type operator already described. The friction saturates at an amplitude of $f_0 = 0.061$ N-m when the velocity reaches a magnitude

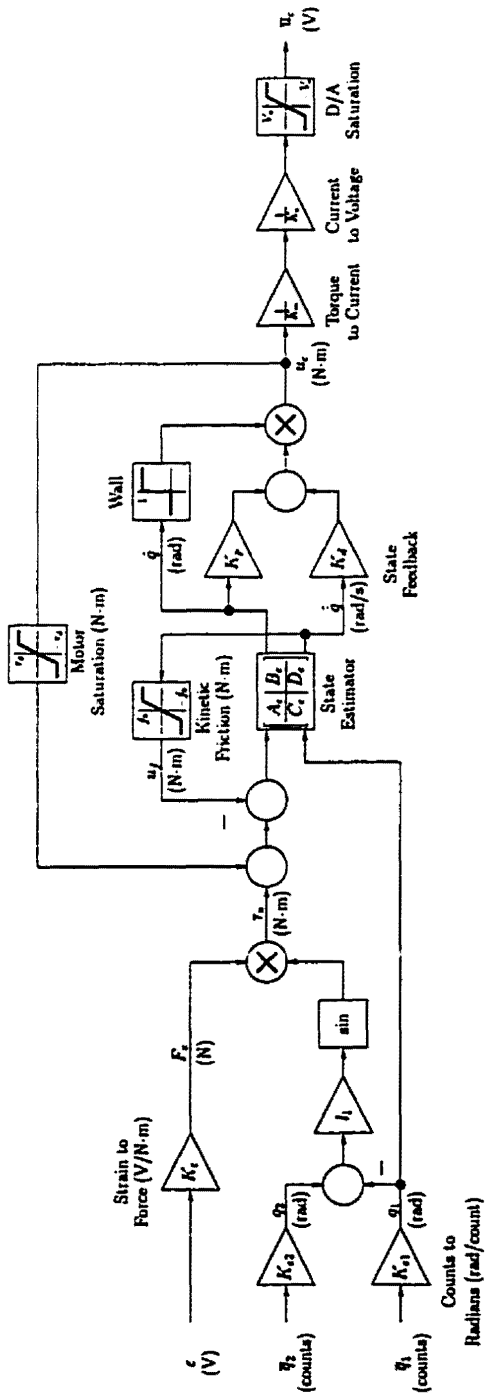


Figure 6.19: Non-linear State Estimator-Based Force Control for the Axial Joint

of 0.03 rad/s. At the 1 millisecond sampling period used for the experiments, this velocity corresponds to 3×10^{-5} rad/period, or 0.057 encoder counts/period. At first glance, a velocity resolution of less than one encoder count per sampling period may seem impossible to resolve. However, it must be remembered that both the strain measurement and encoder measurement are employed by the state estimator to arrive at the estimated velocity, $\hat{\dot{q}}$. Also, finite word length effects can introduce small, yet noticeable, errors in the vicinity of zero. Using viscous damping within this very small velocity range, $[-v_0, v_0]$, rather than just implementing the kinetic friction block as it appears in the simulation of Figure 6.12 does actually make a difference in the controller performance.

The torque commanded of the motors, u_c, is not used directly in the net torque calculation due to potential saturation of the digital-to-analog converter. Hence, when the command torque is fed back in order to compute the net torque acting on the plant, it is first saturated at a magnitude τ_d. Since the digital-to-analog converter saturates before the peak motor torque is reached, the value of τ_d may be computed as:

$$\tau_d = K_m K_a V_a = 7.23 \text{ N-m}$$

where:

$K_m = 0.373$ N-m/A	(motor torque constant)
$K_a = 3.87$ A/V	(amplifier transconductance)
$V_a = 5$ V	(D/A saturation voltage)

The values for the motor torque constant and amplifier transconductance are taken from Table 5.1.

Once the position and velocity are estimated with the state estimator, the same control strategy used for the implicit force control scheme of section 6.1 is employed.

The controller is simply set to equal the desired stiffness, in this case:

$$K(s) = K_p + K_d s$$

The unilateral constraint for the virtual wall is applied based on the estimated joint angle, $\hat{q}$. It is depicted as a step function multiplying the output torque in the diagram. The user enters the virtual wall when the joint angle becomes positive. Note that this constraint is different from the unilateral constraint model proposed by Colgate and Brown [7] in Figure 2.10. In the scheme of Colgate and Brown, the user is subjected to the zero-input response of the controller whenever the user exits the virtual wall. Hence, the user may feel a force even after they have left the wall surface. In the scheme employed here, the output torque is completely eliminated the instant the virtual wall is exited. In section 6.1, the maximum stiffness that could be achieved was less than $K_p = 400$ N-m/rad. For the state estimator results shown later in this section, the stiffness and viscous damping terms are $K_p = 800$ N-m/rad and $K_d = 5$ N-m-s/rad.

Finally, the commanded torque, u_c, is converted to a voltage, $\overline{u}_c$ for the digital-to-analog converter via the motor torque constant, K_m and amplifier transconductance, K_a.

The result of this state estimation strategy is depicted in Figure 6.20, which illustrates the joint angle when the user taps upon the virtual wall using the robot end-effector. Both the measured and estimated joint angle are shown. Note the quantization of the measured joint angle, while the estimated angle remains relatively smooth. The presence of the virtual wall is readily apparent, since the user only enters the wall by a couple of encoder counts.

The corresponding output torque and the torque applied by the user are depicted in Figure 6.21. The dotted "contact" lines below the zero line indicate those areas

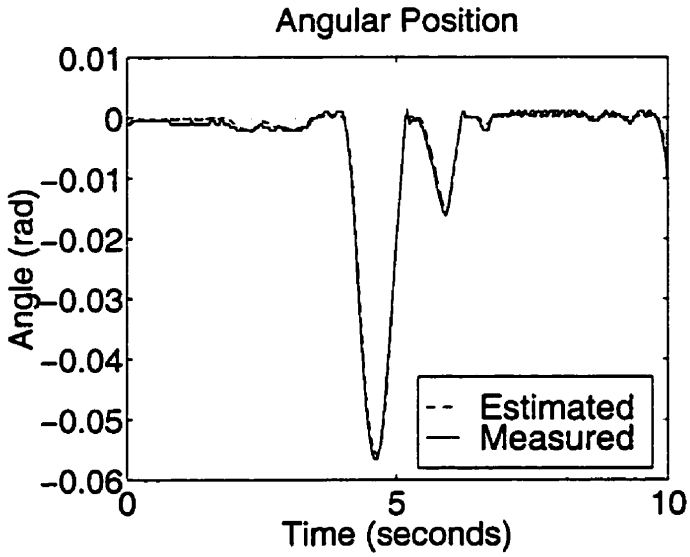


Figure 6.20: Virtual Wall by State Estimation: Angular Position

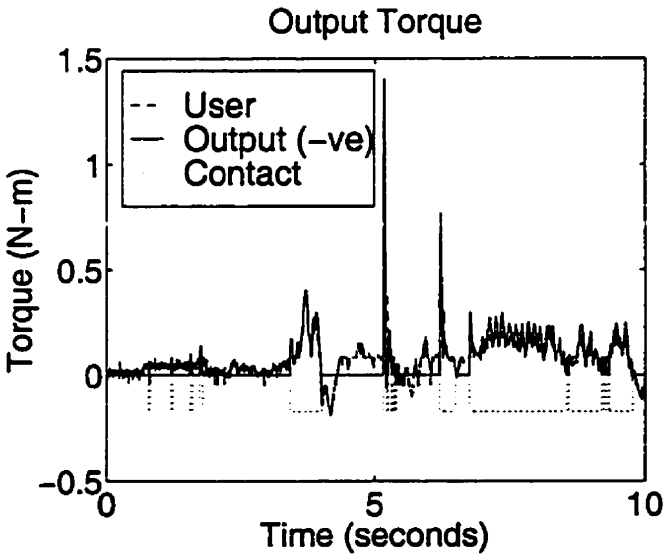


Figure 6.21: Virtual Wall by State Estimation: Torques

where the user is in contact with the wall, based on the position measurement. Note that the output torque tracks the user's input torque within the virtual wall. This force cancellation is to be expected, since the difference between the two curves is the net torque acting on the joint. A net torque of zero results in no displacement, as seen in the graph of the joint angle in Figure 6.20.

The spikes which occur on initial impact are due to the derivative term, K_d, and help to slow the robot down on impact. The result is a "crisp" feel to the virtual wall so that the wall does not simply resemble hard rubber. This "improved" feel may result from the fact that introducing damping reduces the velocity at which the end-effector leaves the virtual wall surface. This reduced velocity has two consequences. First, it decreases the effect of the robot inertia outside the virtual wall, for which there is no compensation in this experiment. The impact of the robot inertia is addressed in the work of Ching and Wang [10].

Second, in an impact between two bodies, there is energy loss when the coefficient of restitution is not unity. The velocity just after impact is related to the velocity just before impact by the coefficient of restitution [95]. While linear damping does not accurately model the coefficient of restitution, it does result in energy loss and a reduced velocity after impact, and it is these factors which may improve the qualitative feel for a human operator. Yigit et al [95] also make the interesting observation that the impact velocity required to cause yield in even hard steel is actually very small (0.14 m/s) and that, as a consequence, "most impacts between metallic bodies involve some plastic deformation" [95].

To demonstrate the success of the nonlinear state estimator at producing a smooth velocity estimate, the estimated velocity and a filtered version of the raw velocity are compared in Figure 6.22. The filtered velocity is computed by using the first-order Euler difference formula on the raw encoder measurements and then

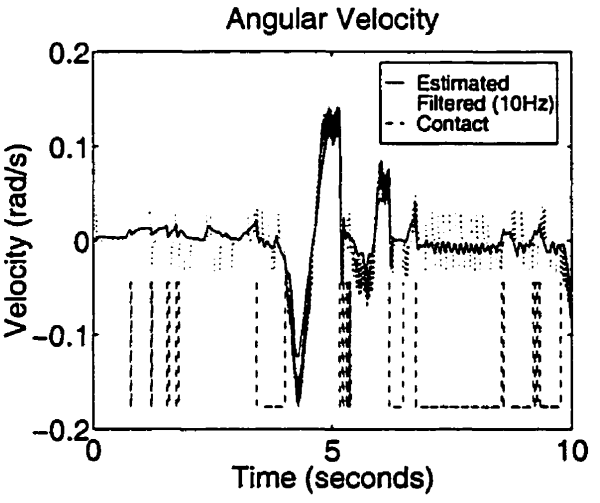


Figure 6.22: Virtual Wall by State Estimation: Velocities

passing the result through a 10 Hz first-order low-pass filter. As seen in the figure, the filtered velocity is still noisier than the estimator output, despite the 10 Hz cutoff frequency of the filter. The dashed curve indicates the regions of robot contact.

The performance of the non-linear state estimator design is superior to the standard implicit force control technique investigated in section 6.1. Much higher stiffnesses are achievable as well as higher damping terms, leading to a more realistic virtual wall. The problem of quantization is addressed by this design technique. The state estimator approach is successful because of the simpler plant dynamics resulting from the mechatronics approach to the robot design. Also note that the same sampling rate was used in both the standard implicit force control design and the state estimator design. Furthermore, it was observed that removing the friction estimation reduces the achievable stiffness. Hence, the friction modelling is an integral part of the estimator strategy.

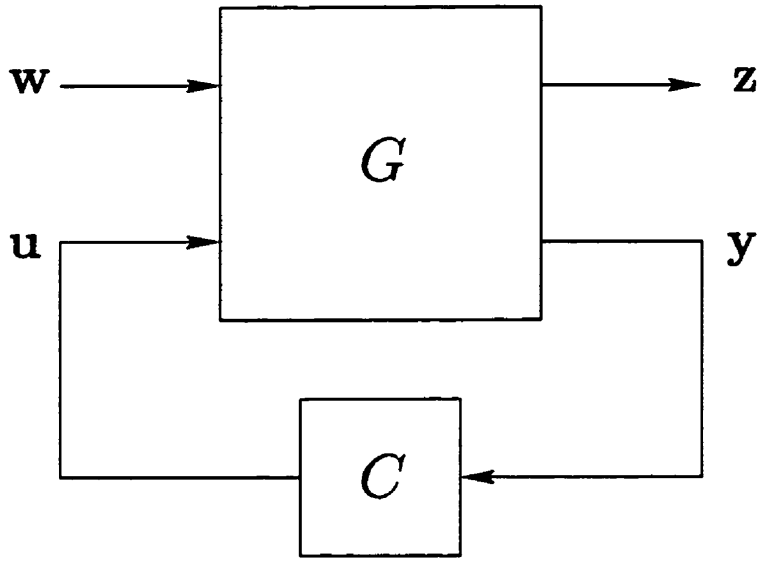


Figure 6.23: General H_∞ Framework [2]

6.4 H_∞ Control

6.4.1 General Framework [2, 3, 4]

While state estimation addresses the need for higher resolution position and velocity coordinates, the design does not necessarily optimize the controller to best match the desired mechanical impedance. Since the goal of this research is to provide the user with realistic force feedback, it seems natural to minimize the difference between the impedance of the controlled system and the target impedance.

The general framework typically used for H_∞ optimal control is illustrated in Figure 6.23. G represents the augmented plant and C depicts the controller. Let G and C be linear time-invariant so that they may be represented by real-rational, proper transfer functions.

The input vector $\mathbf{w}$ represents exogenous inputs such as the reference signal in a trajectory tracking problem. The input vector $\mathbf{u}$ denotes the control signal. The output vector $\mathbf{z}$ represents the quantities to be minimized, such as the control effort or tracking error while the output vector $\mathbf{y}$ represents the signals used by the controller, such as position or velocity. Many standard control problems can be formulated using this general framework.

The equations governing this general framework are:

$$\begin{bmatrix} \mathbf{z} \\ \mathbf{y} \end{bmatrix} = G \begin{bmatrix} \mathbf{w} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{u} \end{bmatrix} \tag{6.8}$$

Let $G_{\mathbf{w},\mathbf{z}}$ denote the transfer function from $\mathbf{w}$ to $\mathbf{z}$. Hence $G_{\mathbf{w},\mathbf{z}}$ is simply the linear fractional transformation (LFT):

$$G_{\mathbf{w},\mathbf{z}} = LFT(G, C) = G_{11} + G_{12}C(I - G_{22}C)^{-1}G_{21}$$

Let C denote the set of all admissible controllers which stabilize the closed-loop. The H_∞ optimization problem seeks to minimize the H_∞ norm of the linear fractional transformation, $LFT(G, C)$, over this set of stabilizing controllers. i.e., the goal is to find C^* such that:

$$C^* = \arg\min_{C \in \mathcal{C}} \|LFT(G, C)\|_\infty$$

The H_∞-norm of $G_{\mathbf{w},\mathbf{z}}$ represents the system gain from $\mathbf{w}$ to $\mathbf{z}$ as follows:

$$\|G_{\mathbf{w},\mathbf{z}}\|_\infty = \sup_{\|\mathbf{w}\|_2 \neq 0} \frac{\|\mathbf{z}\|_2}{\|\mathbf{w}\|_2}$$

Since it represents the system gain with respect to the L_2-norms of the input and output signals, $\mathbf{w}$ and $\mathbf{z}$, the H_∞ optimization problem minimizes based on the *energies* of the signals rather than their amplitude. However, for linear time invariant

systems the distinction is partly academic because L_2 stability implies bounded-input bounded-output stability [96]. For further discussion on norms and stability, refer to [96]. An algorithm to solve the H_∞ optimization problem was developed by Doyle, Glover, Khargonekar and Francis [2]. Their result will be referred to as the "DGKF algorithm". Only the assumptions behind the algorithm are included here, since the algorithm is implemented in commercial software packages such as Matlab™.

Let the augmented plant, G, be represented in state-space form as:

$$G = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right]$$

where G_{ij}, $i,j = 1,2$ from (6.8) is represented by the quadruple (A, B_j, C_i, D_{ij}). Define $m_1 = \dim \mathbf{w}$, $m_2 = \dim \mathbf{u}$, $p_1 = \dim \mathbf{z}$ and $p_2 = \dim \mathbf{y}$. The following assumptions on the general framework of Figure 6.23 are made for the DGKF algorithm [97]:

1. (A, B_1) is stabilizable and (C_1, A) is detectable
2. (A, B_2) is stabilizable and (C_2, A) is detectable
3. D_{12} is tall $(m_2 \leq p_1)$ and full column rank
4. D_{21} is wide $(m_1 \geq p_2)$ and full row rank

These assumptions are necessary for the algorithm but are not all required for a solution to the H_∞ problem to exist. The necessary and sufficient conditions for a solution to exist may be found in [2, 3, 4] and others. The assumptions are listed here because they affect the formulation of the H_∞ implicit force control and H_∞ impedance control problems discussed in subsequent sections.

6.4.2 H_∞ Implicit Force Control

As outlined in section 2.1.4, implicit force control employs measurements of the generalized coordinates alone to produce the desired closed-loop mechanical impedance. Force sensors are not used. While disposing of the need for costly force/torque sensors has financial merit, it should be noted that implicit force control only has utility for a haptic interface that is backdriveable. The user must be capable of moving the device, since the user's input can only be detected through the position sensors of the haptic device.

Furthermore, static friction has a detrimental impact on the quality of the haptic simulation in implicit force control and should be minimized. The problem is simply that the device does not move until static friction is overcome. Once moving, a "free-space" controller can counteract kinetic friction, but the net result is that the haptic device feels "sticky" to the operator, because every time the device is brought to rest, it gets "stuck" by the static friction.

These two factors generally reduce the applicability of implicit force control to direct-drive haptic devices, or devices with a low gearing ratio. However, since haptic devices are often designed as direct-drive devices in order to be backdriveable and to avoid gear backlash, and since force/torque sensors are typically costly units, optimal model matching in the context of implicit force control will be considered.

Consider Figure 6.24. The plant P represents the haptic device, and C denotes the controller. The input u designates the force or torque applied by the user and n signifies the sensor noise. Since the user applies forces directly to the haptic device, the user input appears at the plant input, where, "traditionally", there is a disturbance. In essence, the user *is* a disturbance on the haptic device, but *the transfer function of interest is the mapping from the user "disturbance" to the plant*

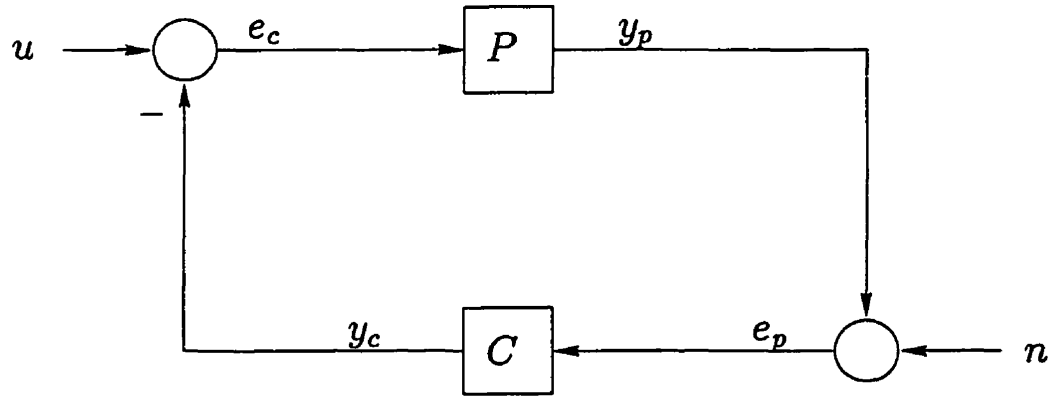


Figure 6.24: Implicit Force Control for Haptic Devices

output, y_p. Again, the user is treated as a purely external force – a model of the human operator is not included.

Note that due to the direct interaction between the haptic device and the user, the controller *cannot* appear between the user and the plant input, so it is necessarily in the feedback loop. Without a force/torque sensor to measure the user input, feedforward terms are not feasible.

Define the sensitivity S and complementary sensitivity, T:

$$S \triangleq (1 + PC)^{-1}$$
$$T \triangleq (1 + PC)^{-1}PC$$

Then the mappings from u and n to the signals, y_c and y_p are:

$$\begin{bmatrix} y_p \\ y_c \end{bmatrix} = \begin{bmatrix} (I + PC)^{-1}P & -(I + PC)^{-1}PC \\ (I + CP)^{-1}CP & (I + CP)^{-1}C \end{bmatrix} \begin{bmatrix} u \\ n \end{bmatrix}$$
$$= \begin{bmatrix} SP & -T \\ T & CS \end{bmatrix} \begin{bmatrix} u \\ n \end{bmatrix} \tag{6.9}$$

The compliance of this controlled system, as perceived by the user, is the transfer function from u to y_p, or SP. A quantity of interest then is the difference between this compliance and the desired system compliance, P_d. Hence, to ensure nominal performance, the weighted difference should be minimized. i.e., find C such that:

$$\|W_1 (SP - P_d)\|_\infty < 1 \tag{6.10}$$

The weight W_1 is chosen to limit the frequency range over which "tracking" of the compliance is deemed important. Hence, it is chosen to have a low-pass characteristic since a human operator will typically probe the wall very slowly. From the work of Lawrence and Chapel [6], the user's probing motions should have a bandwidth of less than 20 rad/s.

In practical control systems, the torque output of the actuators is limited. Hence, it is desirable to bound the control effort in order to avoid the non-linear effects of motor saturation. The transfer function from the user input, u, to the control output, y_c, is:

$$\frac{Y_c}{U} = T$$

The control effort can be restricted according to frequency by defining a frequency-dependent weighting function, W_2, and requiring:

$$\|W_2 T\|_\infty < 1 \tag{6.11}$$

The weighting function, W_2, is typically chosen to be high-pass since the control effort needs to be large at low frequencies (particularly within the bandwidth of human sensation) to provide good compliance tracking, but should be small at high frequencies to avoid oscillations due to high frequency noise or unmodelled dynamics.

However, it is instructive to consider the nature of the input signals. The user input, u, has a fairly limited bandwidth e.g. 20 rad/s [6], since it is produced by a human operator. Most plants also have low-pass characteristics, being mechanical systems. Thus, the input u is not expected to be a source of high-frequency noise. The restriction $\|W_2T\|_\infty < 1$ is based on the transfer function from the input, u, to the control signal, y_c, so due to the bandwidth limitation of u and P, it may not be necessary to make W_2 high-pass.

Indeed, the expected source of high-frequency noise in the system is the sensor noise, n. Thus, the transfer function of interest as far as limiting the control effort, is actually the transfer function from n to y_c:

$$\frac{Y_c}{N} = CS$$

Hence, using a frequency weighting W_3, a further control effort constraint is:

$$\|W_3CS\|_\infty < 1 \tag{6.12}$$

Like W_2, the weight W_3 should be high-pass, based on the prior discussion. For most systems, $\lim_{|s|\to\infty} S(s) = 1$, since both the plant, and the controller, tend to roll off at high frequencies. Hence, typically, $\lim_{|s|\to\infty} C(s)S(s) = 0$, as desired.

At first glance, there would appear to be no tradeoffs. The key transfer functions in (6.10), (6.11) and (6.12) are SP, T and CS respectively. All three typically tend to zero at high frequencies, as desired. (SP represents the compliance of the closed-loop. The smaller the closed-loop compliance, the stiffer the robot appears to the user.)

However, the transfer function $SP - P_d$ deserves closer examination. As demonstrated in section 6.2, perfect compliance matching is achieved for:

$$C^* = P_d^{-1} - P^{-1}$$

However, for the experimental apparatus, P is strictly proper with relative degree two and a pole at the origin. For a virtual wall consisting of a very stiff spring, K_p, and damper, K_d, the desired compliance is:

$$P_d = (K_d s + K_p)^{-1}$$

which is also strictly proper. Hence, the controller which achieves perfect model matching is high gain due to the P_d^{-1} term, and actually *increases* in magnitude with frequency due to P^{-1}. Let the controller be C^*. Then:

$$SP - P_d = 0$$

$$T = 1 - P_d P^{-1}$$

$$C^* S = P^{-1}(1 - P_d P^{-1}) = P^{-1} T$$

The nominal performance is ideal, but T and C^*S have very poor characteristics. Since P has relative degree two and a pole at the origin, and P_d only has relative degree one, the magnitude of T has a peculiar frequency response. At low frequencies, where the gain of P is high and P_d is small, T is close to one. At higher frequencies, the magnitude of $P^{-1}P_d$ will increase continuously since P has a higher relative degree than P_d. Indeed, it will increase at 20 dB/decade. Hence, T will also increase in magnitude at high frequencies. Thus, the control effort required to achieve perfect tracking becomes exorbitantly large at high frequencies, and may never be small.

The behaviour of C^*S is similar, since $C^*S = P^{-1}T$. The difference is that C^*S is very small at low frequencies due to the pole at the origin and relative degree of P. At high frequencies, it will increase even faster than T, again due to the relative degree of P. Indeed, C^*S will increase at 60 dB/decade! Hence, while the effect of sensor noise on control effort is small at low frequencies, it gets quickly worse at higher frequencies and the controller must roll off quickly to compensate.

Thus, the weights must be chosen to balance performance against control effort and sensor noise. The effect of sensor noise is expected to be particularly difficult to manage while maintaining good performance. Since the requisite control effort and the impact of sensor noise increase dramatically with frequency, the practical operating bandwidth is expected to be limited. Indeed, the tradeoff is likely to be most apparent as the desired operating bandwidth is progressively increased during the design process.

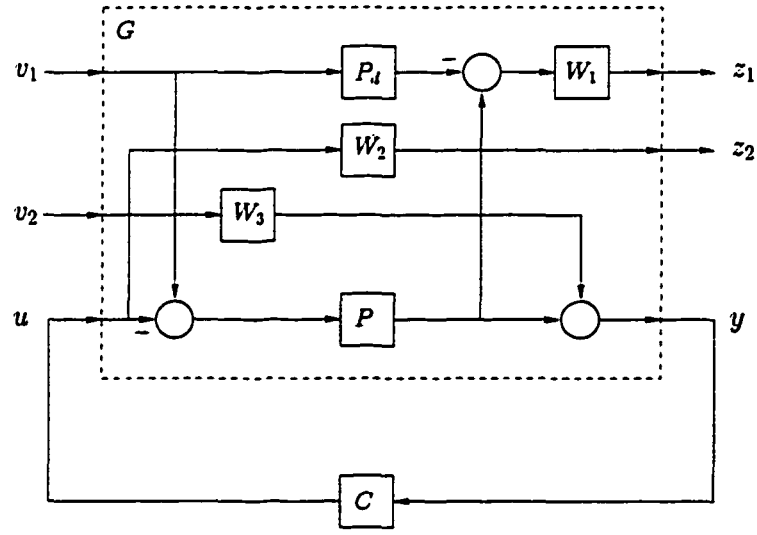


Figure 6.25: Implicit Model Matching Framework

The framework for the implicit force control model-matching problem is shown in Figure 6.25. The inputs to the system are:

v_1 = external torque applied by the operator

v_2 = additive output sensor noise

u = control signal

while the outputs of the system are:

z_1 = weighted error between controlled plant and target

z_2 = weighted control effort

y = measured plant output

Typically, the measured plant output is position rather than velocity. Hence, the plant, P, maps torque to position and thus represents the plant *compliance* rather than its impedance (see section 2.1.3). Similarly, P_d denotes the target compliance. Thus, the quantity z_1 is the weighted measure of the error in plant compliance described by (6.10). The operator C represents the controller.

The weighting function W_1 allows the compliance comparison to be restricted to the bandwidth of interest. The weight W_2 limits the control effort while W_3 weights the sensor-noise.

The output equations corresponding to this framework are:

$$z_1 = W_1(P(v_1 - u) - P_d v_1)$$
$$z_2 = W_2 u$$
$$y = W_3 v_2 + P(v_1 - u)$$
$$u = Cy$$

or in matrix form:

$$\left[\begin{array}{c} z_1 \\ z_2 \\ \hline y \end{array}\right] = \left[\begin{array}{cc|c} W_1(P - P_d) & 0 & -W_1 P \\ 0 & 0 & W_2 \\ \hline P & W_3 & -P \end{array}\right] \left[\begin{array}{c} v_1 \\ v_2 \\ \hline u \end{array}\right] \equiv \left[\begin{array}{c|c} G_{11} & G_{12} \\ \hline G_{21} & G_{22} \end{array}\right] \left[\begin{array}{c} v_1 \\ v_2 \\ \hline u \end{array}\right] \quad (6.13)$$

Let the state-space representation of G_{ij} be denoted by the matrix quadruple $(A_{ij}, B_{ij}, C_{ij}, D_{ij})$. The addition of a non-strictly proper weight, W_2, on the control effort ensures that D_{12} is full column rank, as per assumption 3. Without W_2, the only term in D_{12} would be due to W_1P. Since P is strictly proper, and W_1 is preferably low-pass (hence proper), W_1P is strictly proper. Strictly proper transfer functions have a zero D matrix in their state-space representation, and hence D_{12} would not be full column rank in this case.

Similarly, the addition of a non-strictly proper weight W_3 on the sensor noise ensures that D_{21} is full row rank, as required by assumption 4. Let:

$$\mathbf{z} \triangleq [z_1 \quad z_2]^{\mathrm{T}} \qquad \text{— signals to be constrained}$$

$$\mathbf{v} \triangleq [v_1 \quad v_2]^{\mathrm{T}} \qquad \text{— exogenous inputs}$$

The transfer function minimized in the H_∞ problem is the mapping from $\mathbf{v}$ to $\mathbf{z}$, which is the linear fractional transformation (LFT) of $G(s)$ and $C(s)$:

$$LFT(G, C) = G_{11} + G_{12}C(I - G_{22}C)^{-1}G_{21}$$

Substituting for the components of G into the LFT and using the relationships:

$$P(I - CP)^{-1} = (I - PC)^{-1}P$$

$$\text{and} \qquad (I - PC)^{-1} = I + C(I - PC)^{-1}P,$$

it is readily shown that:

$$LFT(G, C) = \begin{bmatrix} W_1\left((I - PC)^{-1}P - P_d\right) & -W_1(I - PC)^{-1}PCW_2 \\ W_3(I + C(I - PC)^{-1}P) & W_3C(I - PC)^{-1}W_2 \end{bmatrix}$$

$$= \begin{bmatrix} W_1(SP - P_d) & -W_1TW_3 \\ W_2T & W_2CSW_3 \end{bmatrix}$$

Notice that the three transfer functions of interest: $SP - P_d$, T and CS are included in the linear fractional transformation, with appropriate weightings. However, the fact that there is a combined weight of W_1W_3 on T and W_2W_3 on CS can make it more difficult to choose the weighting functions.

The objective of the standard H_∞ control synthesis problem is to arrive at a stabilizing controller, C, which minimizes the H_∞ norm of this linear fractional transformation.

Now, let $\mathbf{e}_i$ denote the i-th column of the identity matrix and consider a matrix function $G \in H_\infty^{n \times p}$. Let g_{ij} denote the (i,j)-th element of G. Clearly,

$$g_{ij} = \mathbf{e}_i^T G \mathbf{e}_j$$

Hence, using the submultiplicative property of induced norms:

$$||g_{ij}||_\infty = ||\mathbf{e}_i^T G \mathbf{e}_j||_\infty \leq ||\mathbf{e}_i||_\infty ||G||_\infty ||\mathbf{e}_j||_\infty = ||G||_\infty$$

Thus,

$$\|LFT(G,C)\|_\infty \geq \|LFT_{ij}(G,C)\|_\infty,\ i = 1,2, j = 1,2$$

The DGKF algorithm solves for C such that $\|LFT(G,C)\| < 1$. Hence, it also ensures that the ∞ norm of each element of $LFT(G,C)$ is less than one. Thus, the above design framework results in the satisfaction of the constraints defined by (6.10), (6.11) and (6.12) [bearing in mind that the weight on CS is now the combined weight W_2W_3], although the results may be conservative.

Now, consider the axial joint of the five-bar robot, as before. The plant model is defined in (6.5). Let the desired plant compliance be:

$$P_d(s) = \frac{1}{K_d s + K_p}$$

to match the wall characteristics used in section 6.1 and defined by (6.1). In particular, to compare the results to section 6.1, let $K_p = 400$ N-m/rad and let $K_d = 0$ N-m-s/rad.

To get perfect compliance matching, the controller C^* in this case is:

$$C^*(s) = -Js^2 + (K_d - B)s + K_p = -0.204s^2 - 0.012s + 400$$

The corresponding transfer functions, T and CS, are:

$$T(s) = \frac{-Js^2 + (K_d - B)s + K_p}{K_d s + K_p} = \frac{-0.204s^2 - 0.012s + 400}{400}$$

and

$$\begin{aligned} C(s)S(s) &= \frac{s(-J^2s^3 + J(K_d - 2B)s^2 + (K_dB + K_pJ - B^2)s + K_pB)}{K_d s + K_p} \\ &= \frac{s(-0.042s^3 - 0.0047s^2 + 82s + 4.6)}{400} \end{aligned}$$

The magnitudes of T and C^*S are depicted in the frequency response of Figure 6.26. The control effort $T(s)$ has two zeroes at approximately 44 rad/s. Hence, it is virtually constant at its DC gain of one, and then increases at 40 dB/decade for frequencies higher than 44 rad/s, or 7 Hz.

The effect of noise on the control effort, $CS(s)$, has the same zeroes as $T(s)$, as well as zeroes at the poles of the plant: 0 and -0.0564. It crosses the 0dB point at approximately 2.2 rad/s. It increases at 40 dB/decade until it reaches 44 rad/s, at which point it begins to increase at 80 dB/decade.

Note that the zero crossing point is actually due more to the plant than the particular form of the controller. Omitting the Laplace variable for clarity, notice

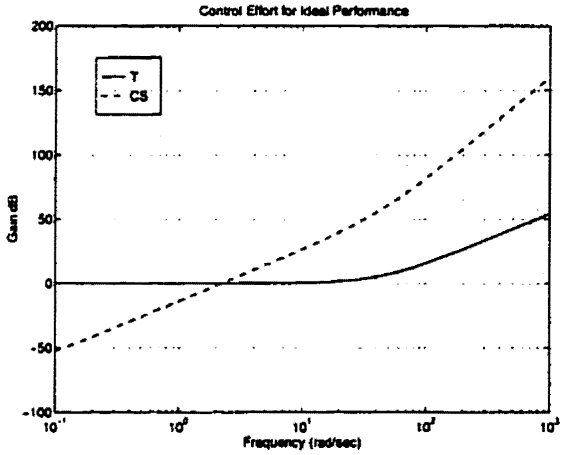


Figure 6.26: Control Effort for "Ideal" Controller C^*

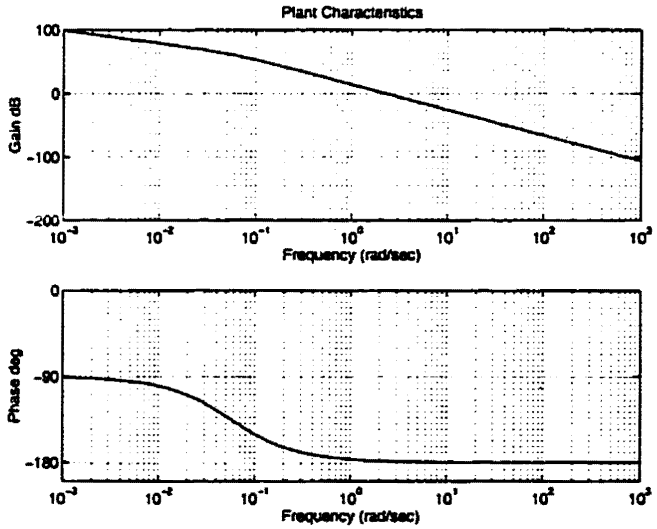


Figure 6.27: Frequency Response for the Axial Joint

that:

$$CS = \frac{C}{1+PC} = \frac{1}{\frac{1}{C}+P} \approx P^{-1} \text{ for large } C$$

But the 0dB frequency of P^{-1} is 2.2 rad/s. Even for a stiffness, K_p, as low as 10 N-m/rad, the 0dB frequency of CS is still 2.1 rad/s.

Clearly, it is likely to be very difficult to achieve good tracking over a bandwidth larger than 44 rad/s, and problems may begin to appear between 2.2 rad/s and 44 rad/s.

Consider as well the Bode plot of the plant, as shown in Figure 6.27. The cross-over frequency is approximately 2.2 rad/s. Note that the phase at the cross-over frequency is almost -180° because the plant has relative degree two. For good performance over a bandwidth larger than 2.2 rad/s, the controller must have high gain over this bandwidth, resulting in a loop gain, PC, with a higher cutoff frequency.

At higher frequencies, the phase of the plant is even closer to -180° , so for a minimum-phase controller to stabilize the system, it must have positive phase at the cross-over frequency of PC. But, for a stable, minimum-phase controller, C, normalized such that $C(0) > 0$, the magnitude uniquely determines the phase [3]. In particular, consider the following lemma:

Lemma 6.4.1 *[3] Given a proper transfer function C such that $C(0) > 0$, and C and C^{-1} are analytic in $\mathfrak{Re}\{s\} \geq 0$, for each frequency, ω_0:*

$$\angle C(j\omega_0) = \frac{2\omega_0}{\pi} \int_0^\infty \frac{\ln|C(j\omega)| - \ln|C(j\omega_0)|}{\omega^2 - \omega_0^2} d\omega$$

Therefore, as a general rule of thumb, the faster the magnitude Bode plot changes close to a particular frequency, ω_0, the greater the magnitude of the phase at that point. For example, if the magnitude of C is decreasing at ω_0, then the phase of C will generally be negative at ω_0. If the magnitude of C is increasing, then the phase will generally be positive.

Thus, for the phase of C to be positive at the cross-over frequency of L, then the magnitude of C must be *increasing* at that frequency. This fact results in somewhat of a paradox. The loop gain, L, determines the bandwidth of the system. And generally, it is desirable, due to noise considerations, to have the controller roll off outside this bandwidth. Yet, due to the relative degree of the plant being two, the controller gain must be *increasing* at the edge of the operating bandwidth. Hence, it is expected that the optimization will produce a small peak in the controller transfer function just past the loop gain cutoff frequency.

Consider the following H_∞ design. Let the weighting functions be:

$$\begin{aligned} W_1(s) &= 100\left(\frac{10}{s+10}\right) \\ W_2(s) &= 0.1 \\ W_3(s) &= 0.001; \end{aligned} \tag{6.14}$$

Hence, the nominal performance has a weight, W_1, of 100 over a 10 rad/s bandwidth. A weight of 100 may seem very large, but consider the units of the transfer function $SP - P_d$. The units are rad/N-m. If the robot is being operated about the home position, then the axial force exerted by the user is amplified by a moment arm of $l_1 = 30$ cm. Hence, 1 N-m of torque is produced by a force at the end-effector of 3.3 N or 0.34 kgf. Now, for a SISO system, the H_∞ norm may be written:

$$\|G\|_\infty = \sup_\omega |G(j\omega)|$$

Suppose $\|G\|_\infty \leq \alpha$ for some constant α. Then $|G(j\omega)| \leq \alpha$ for any choice of frequency ω. Hence, if the input u is a sinusoid of frequency ω_0 i.e., $u(t) = \sin(\omega_0 t)$, then the output will be sinusoidal, with amplitude less than α.

Returning to the choice of weight, W_1, consider a sinusoidal torque input of 1 N-m magnitude at a frequency ω_0. A constant weight, W_1, of 100 ensures that $\|SP - P_d\|_\infty < 0.01$. Hence, the difference in the actual position of the robot and the desired position will be a sinusoid of magnitude less than 0.01 rad when this 1 N-m sinusoidal torque is applied. Applying the moment arm, this error corresponds to a distance of 3 mm at the end-effector, which is reasonable.

Similarly, W_2 weights the transfer function from the user input torque to the control torque. Setting W_2 to 0.1 allows the control torque to exceed the torque applied by the user. The control torque must exceed the torque applied by the user if the user is to be pushed out of the virtual wall. Setting W_2 to a constant puts a limit on this control torque.

Finally, W_3 weights the sensor noise. Since the encoders generate very clean signals, the sensor noise is expected to be small. For example, each encoder count corresponds to $\frac{\pi}{6000}$ rad, or 0.00052 rad. Hence, for a sinusoidal input v_2 of magnitude one, the sensor noise injected as an output disturbance will have magnitude 0.001 rad, or approximately 2 encoder counts.

In order to perform the optimization using the DGKF algorithm, a state-space representation of the augmented plant, G, of (6.13), must be formed. The state-space representation should be minimal to ensure that the requisite stabilizability and detectability requirements are satisfied[3]. A minimal representation (assuming

[3]Forming the augmented plant in transfer function form, and then converting to a minimal state-space realisation, led to occasional numerical problems that produced spurious results. Other "reduction" techniques also failed when certain weights were chosen.

no pole-zero cancellations due to the choice of weights) was derived manually. For the augmented plant of (6.13), the state-space representation is:

$$A = \begin{bmatrix} A_p & 0 & 0 & 0 & 0 \\ 0 & A_d & 0 & 0 & 0 \\ B_{W1}C_p & -B_{W1}C_d & A_{W1} & 0 & 0 \\ 0 & 0 & 0 & A_{W2} & 0 \\ 0 & 0 & 0 & 0 & A_{W3} \end{bmatrix} \tag{6.15}$$

$$B = \left[\begin{array}{cc|c} B_p & 0 & -B_p \\ B_d & 0 & 0 \\ B_{W1}(D_p - D_d) & 0 & -B_{W1}D_p \\ 0 & 0 & B_{W2} \\ 0 & B_{W3} & 0 \end{array}\right] \tag{6.16}$$

$$C = \left[\begin{array}{ccccc} D_{W1}C_p & -D_{W1}C_d & C_{W1} & 0 & 0 \\ 0 & 0 & 0 & C_{W2} & 0 \\ \hline C_p & 0 & 0 & 0 & C_{W3} \end{array}\right] \tag{6.17}$$

$$D = \left[\begin{array}{cc|c} D_{W1}(D_p - D_d) & 0 & -D_{W1}D_p \\ 0 & 0 & D_{W2} \\ \hline D_p & D_{W3} & -D_p \end{array}\right] \tag{6.18}$$

The controller is calculated using the DGKF algorithm. Substituting for the plant and weights from (6.5) and (6.14), in (6.15)-(6.18), the resulting controller is:

$$A_c = \begin{bmatrix} -54.5 & 44.8 & -45.1 \\ -124.8 & -92.1 & 73.9 \\ 31.4 & 13.4 & -27.9 \end{bmatrix} \quad B_c = \begin{bmatrix} -721 \\ -3228 \\ 907 \end{bmatrix}$$
$$C_c = \begin{bmatrix} 4.19 & -22.0 & 37.3 \end{bmatrix} \quad D_c = \begin{bmatrix} 0 \end{bmatrix} \tag{6.19}$$

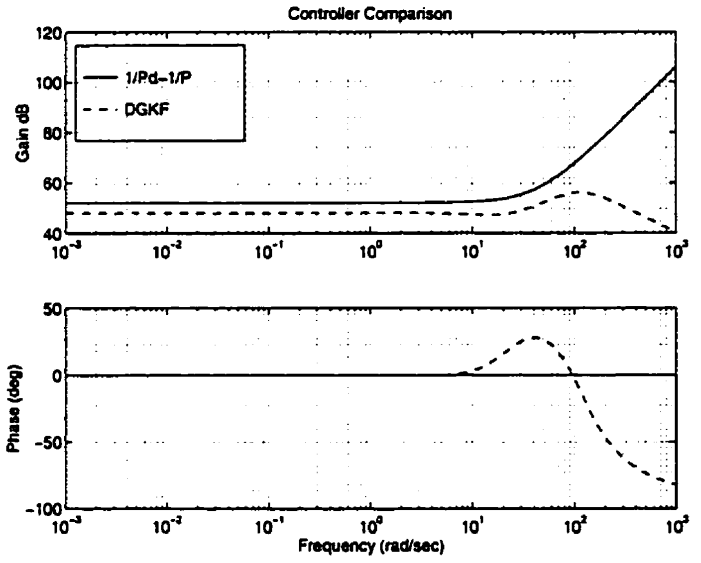


Figure 6.28: Bode Plot of Controller

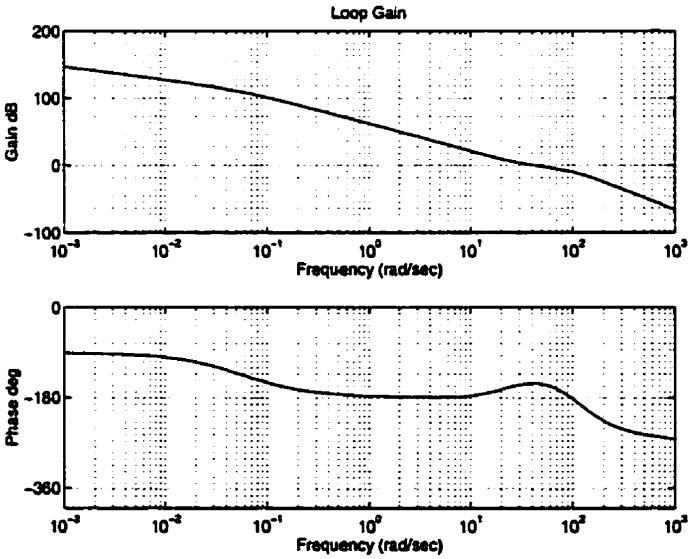


Figure 6.29: Bode Plot of Loop Gain

The Bode plot of the controller is depicted in Figure 6.28. The solid curve represents the frequency response of the controller, C^*, that results in perfect compliance matching. The dotted curve illustrates the frequency characteristics of the optimal controller of (6.19). Notice that the optimal controller is close to the ideal controller over the bandwidth of interest (10 Hz or 62 rad/s). Also observe that the phase of C^* only appears flat because it is non-minimum phase (the zeros are -44.30 and 44.25). There is a dip in the phase around 44 rad/s but its magnitude is too small to be visible at the scale of Figure 6.28.

Note the small peak in the controller magnitude around 100 rad/s. The reason for the peak becomes clear by examination of the loop gain. The loop gain is depicted in Figure 6.29. The loop gain cross-over frequency is approximately 41.7 rad/s. The phase ventures very close to -180° around 2 rad/s but the phase peaks at the cross-over frequency. Evidently, the optimization attempts to provide some measure of robustness to the system by adding phase near the cross-over frequency. This additional phase is provided by the controller.

The extra phase can also be seen in the phase plot of the controller in Figure 6.28. As discussed previously, the plant is relative degree two and thus has a phase close to -180° at higher frequencies. Hence, the loop gain, PC, also has a phase close to -180°, except where the controller provides additional phase. The controller in this case is stable and minimum-phase, so by Lemma 6.4.1, its magnitude must be increasing where additional phase is required. This phenomenon is readily visible in Figure 6.28. The phase peaks where the magnitude is increasing. As the magnitude levels off, the phase returns to zero. As the magnitude decreases, the phase goes negative. The result of this peak in the controller magnitude after the loop gain cross-over frequency of 41.7 rad/s is the needed positive phase contribution. Note that even when a small cutoff frequency is chosen for the performance weighting,

W_1, the controller still exhibits a peak just above the loop-gain cross-over frequency.

It should also be noted that this design is not standard in that the transfer function of interest is SP rather than S. Typically, the sensitivity S measures performance because it is the transfer function from the input to the tracking error. Multiplicative uncertainty in the plant is then managed by weighting the complementary sensitivity, T. Since $S + T = I$, there is a clearly defined tradeoff. In the case of the H_∞ implicit force control framework developed in this section, the tradeoff is not as well-defined, and thus may be more difficult to manage. However, it should be noted that the problem itself is essentially equivalent to requiring that the robot be of such high stiffness that it is perceived by the human operator as coming to an immediate stop upon contact with the virtual wall – not an easy task for any control system.

The frequency responses of the performance measure, z_1, and control effort measure, z_2, to the user torque input are shown in Figure 6.30. The responses for the sensor noise input are illustrated in Figure 6.31. In all cases, the responses fall well below the 0 dB line. Hence, the specifications are easily met.

Of course, it is important to verify that the controller actually works on the experimental apparatus. Figure 6.32 illustrates the output torque of the controller as the user probes the virtual wall. The solid line denotes the motor torque, and the dotted line depicts the torque applied by the user. The corresponding joint angles are illustrated in Figure 6.33.

The controller only produces a torque when the user enters the wall. While damping was not specified as part of the desired compliance, the spikes in the output torque as the user enters the wall seem to indicate that there is some damping in the controller. Notice that when the user holds the end-effector against the wall

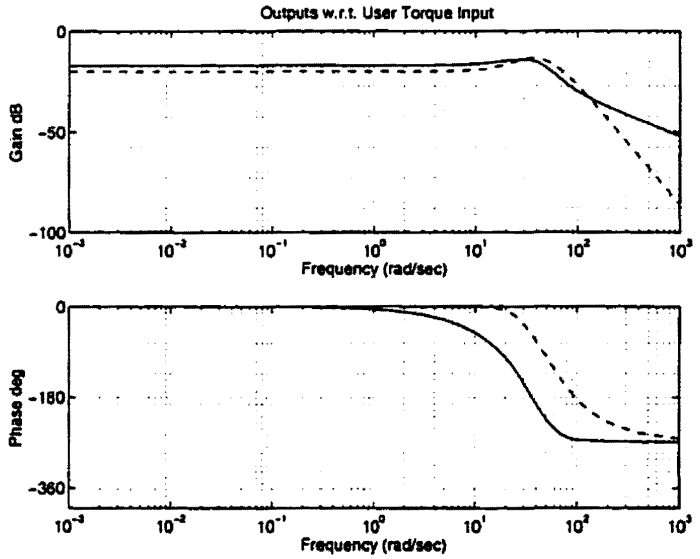


Figure 6.30: Performance and Control Effort w.r.t. User Input

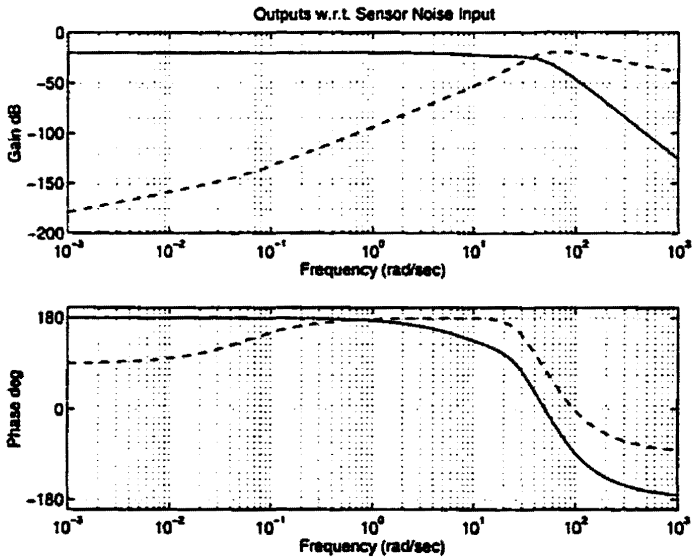


Figure 6.31: Performance and Control Effort w.r.t. Output Disturbances

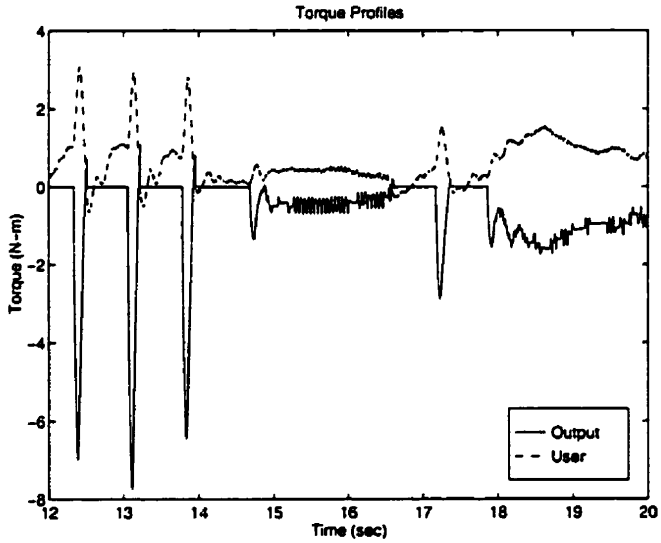


Figure 6.32: Torque Applied By Operator

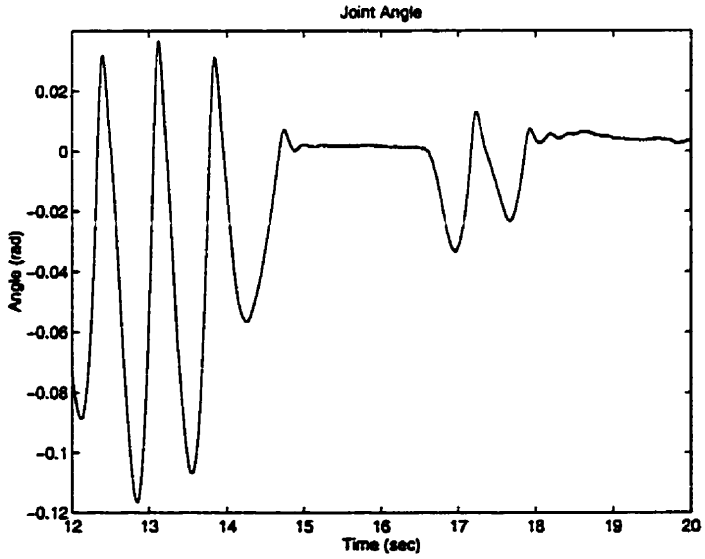


Figure 6.33: Axial Joint Angle as User Probes Wall

the output torque approximately cancels the user torque, as expected. However, the motor torque signal is noisy, and this noise is perceived as a small vibration by the user. The frequency of the noise is 19.6 Hz, or 123 rad/s, according to a Fast Fourier Transform analysis of the recorded data. Adding damping to the desired compliance model, P_d, and redoing the design failed to eliminate the small vibrations.

Note that the frequency of the vibrations changed depending on the controller and appeared to have some correlation with the location of the peak in the controller transfer function, although a definitive relationship has yet to be ascertained. For this particular controller, it is also very close to the natural frequency of the robot structure in the *horizontal* direction of 19.2 Hz (see section 5.4.1), even though the axial joint is being controlled. Hence, presumably there is some interplay between the natural frequency of the robot structure and the controller. The horizontal direction is uncontrolled in these experiments.

The displacement when the user holds the end-effector against the wall is quite small, as depicted in Figure 6.33. The magnified view of Figure 6.34 shows that the oscillations apparent in the torque translate into displacements of only one encoder count. As in the other experiments, the sampling rate is 1 kHz, much faster than the bandwidth of the closed-loop system and the frequency of the vibrations. Hence, it is the encoder resolution that appears to be the cause of the vibrations.

6.4.3 H_∞ Impedance Control

The state estimator approach yields a better virtual wall than both the implicit force control schemes addressed up to this point. The addition of the torque measurement allows the state estimator to smooth out the velocity measurement and

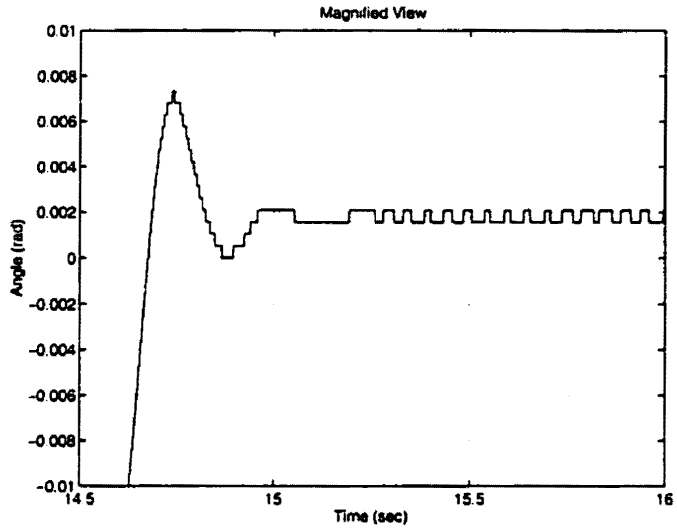


Figure 6.34: Magnified View of Joint Angles

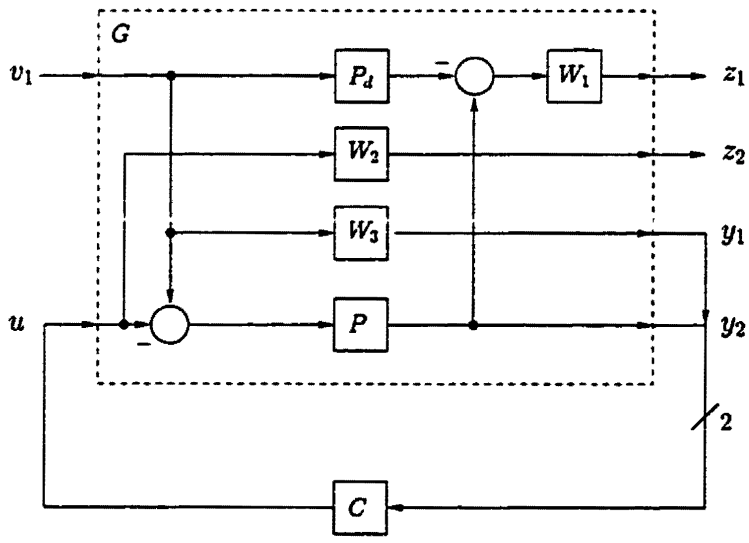


Figure 6.35: Preliminary H_∞ Impedance Control Framework

quantized encoder signals. Hence, a natural extension of the H_∞ implicit force control scheme of the preceding section is to incorporate the measured torque signal into the controller design, yielding an H_∞ *impedance* controller.

A preliminary framework for optimal H_∞ impedance control is illustrated in Figure 6.35. The output z_1 is the same nominal performance measure employed for the H_∞ implicit force control of the preceding section. Similarly, z_2 is the weighted control effort measure.

The key difference is in the controller. The controller takes two inputs instead of one. The first input, y_1, is the measured user torque. However, the torque is weighted to allow sensor noise considerations to be accommodated in the design. The second input, y_2, represents the position as measured by the encoder. Hence, like the state estimator design, the controller has both the torque and position at its disposal. Presumably, then, this framework should yield results comparable with the state estimator design.

The augmented plant, G, may be represented in transfer function form as:

$$G(s) = \left[\begin{array}{c|c} W_1(P-P_d) & -W_1P \\ 0 & W_2 \\ \hline W_3 & 0 \\ P & -P \end{array}\right] \equiv \left[\begin{array}{c|c} G_{11} & G_{12} \\ \hline G_{21} & G_{22} \end{array}\right]$$

As before, let the state-space representation of the transfer function matrices, G_{ij}, be denoted by the quadruple $(A_{ij}, B_{ij}, C_{ij}, D_{ij})$ of constant matrices. Provided the weight on the control effort, W_2, is non-strictly proper, D_{12} will be full row rank, satisfying assumption 3 of the DGKF algorithm. Similarly, if W_3 is non-strictly proper, D_{21} will be full column rank and assumption 4 will be satisfied.

While all the conditions for the DGKF algorithm are satisfied, the optimization

produces an interesting, but impractical result. Let the controller be denoted as $C = [K_1 \quad K_2]$ such that:

$$u = K_1 y_1 + K_2 y_2 = K_1 W_3 v_1 + K_2 y_2$$

The optimization produces a controller in which, regardless of the choice of W_3, $K_1 = W_3^{-1}$ and $K_2 = 0$. The logic is simple. To prevent the robot from moving, it is sufficient, by Newton's Law, to apply an equal and opposite torque to the torque being applied by the user, v_1. Since W_3 is necessarily non-strictly proper in this design framework, in order to satisfy assumption 4 of the DGKF algorithm, it is always invertible. Hence, the optimization produces:

$$u = W_3^{-1} W_3 v_1 + 0 y_2 = v_1$$

Due to noise in the strain gauge measurement and potential calibration errors, it is not practical to implement a virtual wall using pure force control. Hence, the framework must be adjusted to prevent the optimization algorithm from producing $u = v_1$. The new framework is depicted in Figure 6.36. The inputs to the new augmented plant are:

v_1 = user torque input

v_2 = noise in the strain gauge measurement

v_3 = noise in the encoder measurement

and similarly the outputs are:

z_1 = weighted nominal performance

z_2 = weighted control effort

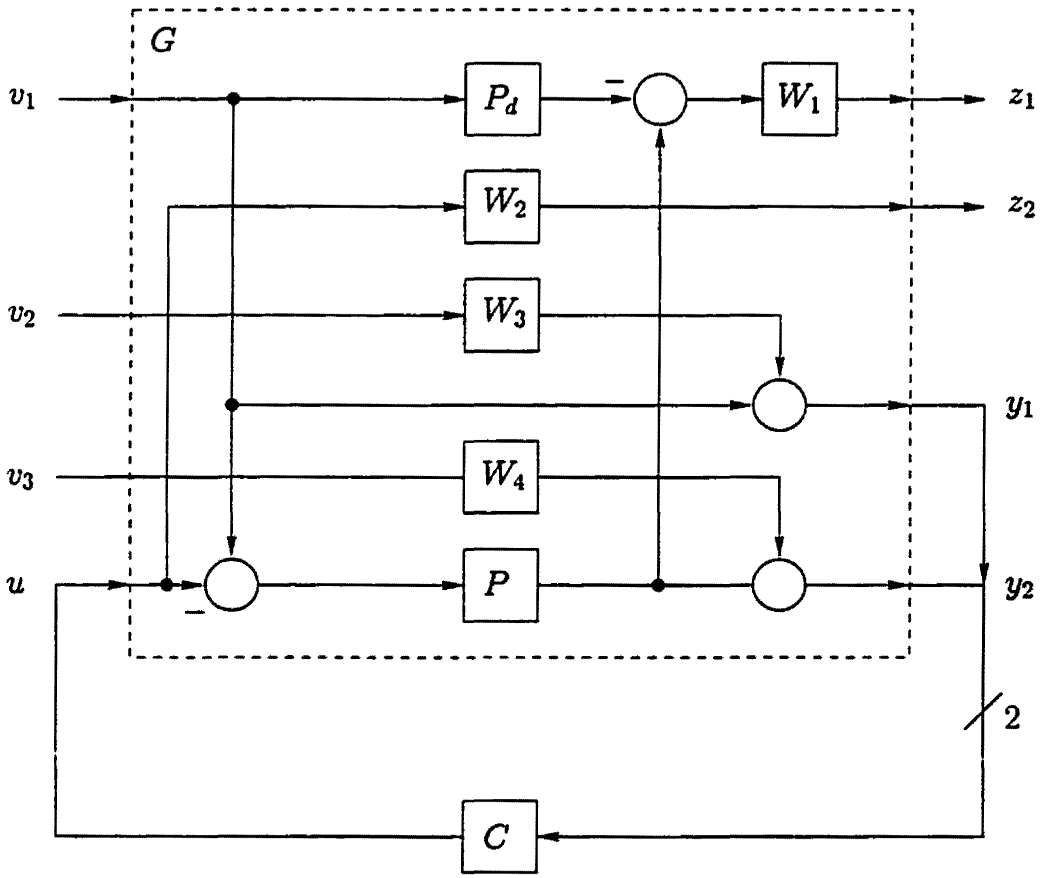


Figure 6.36: H_∞ Impedance Control Framework

y_1 = measured user torque, including noise

y_2 = measured position, including noise

The transfer function matrix for the augmented plant, G, is:

$$G(s) = \left[\begin{array}{ccc|c} W_1(P-P_d) & 0 & 0 & -W_1P \\ 0 & 0 & 0 & W_2 \\ \hline I & W_3 & 0 & 0 \\ P & 0 & W_4 & -P \end{array}\right]$$

By adding a separate input for the strain gauge noise, the optimization algorithm can no longer eliminate the weighting function, W_3. In fact, W_3 may now be strictly proper without affecting the row rank of D_{21}.

However, due to the change in structure, the weight W_3 could not be used to force D_{21} to be full rank. Hence, another input, the noise of the encoder signal, v_3, had to be added. Thus, the noise for both the strain gauge measurements and encoder measurements is modelled, and can be shaped using the weighting functions W_3 and W_4 respectively.

Typically, W_3 and W_4 are chosen to be constants or high-pass filters to reduce the effect of high-frequency noise on the controller performance. Let α be a constant parameter. Suppose $W_3 = \alpha W_4$. By varying α, the relative contribution of the strain gauge measurements versus the encoder readings to the controller output may be adjusted.

The state-space representation of the augmented plant, G, was manually derived. Provided there are no pole-zero cancellations between the weighting functions, the plant, P, and the desired plant, P_d, this state-space representation will be minimal. Let the state-space representation of the plant be (A_p, B_p, C_p, D_p)

as before. Similarly, let the desired plant be represented by the matrix quadruple (A_d, B_d, C_d, D_d) and weight i by $(A_{Wi}, B_{Wi}, C_{Wi}, D_{Wi})$. Then the state-space representation of the augmented plant, G, is:

$$A_G = \begin{bmatrix} A_p & 0 & 0 & 0 & 0 & 0 \\ B_{W1}C_p & -B_{W1}C_d & A_{W1} & 0 & 0 & 0 \\ 0 & 0 & 0 & A_{W2} & 0 & 0 \\ 0 & 0 & 0 & 0 & A_{W3} & 0 \\ 0 & 0 & 0 & 0 & 0 & A_{W4} \end{bmatrix}$$

$$B_G = \left[\begin{array}{ccc|c} B_p & 0 & 0 & -B_p \\ B_d & 0 & 0 & 0 \\ B_{W1}(D_p - D_d) & 0 & 0 & -B_{W1}D_p \\ 0 & 0 & 0 & B_{W2} \\ 0 & B_{W3} & 0 & 0 \\ 0 & 0 & B_{W4} & 0 \end{array}\right]$$

$$C_G = \left[\begin{array}{cccccc} D_{W1}C_p & -D_{W1}C_d & C_{W1} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{W2} & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & C_{W3} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{W4} \end{array}\right]$$

$$D_G = \left[\begin{array}{ccc|c} D_{W1}(D_p - D_d) & 0 & 0 & -D_{W1}D_p \\ 0 & 0 & 0 & D_{W2} \\ \hline I & D_{W3} & 0 & 0 \\ D_p & 0 & D_{W4} & -D_p \end{array}\right]$$

Notice the similarity between the state-space representation of the augmented plant for the H_∞ impedance control framework above, to the state-space representation of the H_∞ implicit force control augmented matrix of (6.15)-(6.18).

Let $S \triangleq (1 + K_2P)^{-1}$. The linear fractional transformation for the augmented plant is:

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_1\left(P(I - S(K_1 + K_2P)) - P_d\right) & -W_1W_3PSK_1 & -W_1W_4PSK_2 \\ W_2S(K_1 + K_2P) & W_2W_3SK_1 & W_2W_4SK_2 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$$

Note that if $K_1 = 0$ then the LFT reduces to the implicit force control transfer function. Unfortunately, with the number of transfer functions involved, the results are likely to be conservative.

The above H_∞ impedance control framework was applied to the axial joint of the experimental apparatus. Numerous designs were attempted and many controllers were implemented successfully on the experimental apparatus, but only the following design will be presented here.

Let the weighting functions be chosen to be:

$$W_1 = \frac{10000}{s+1}$$
$$W_2 = 0.1\frac{(s/20+1)^2}{(s/200+1)^2}$$
$$W_3 = \frac{8 \times 10^{-4}}{(s/60+1)^2}$$
$$W_4 = 10^{-6}$$

The nominal performance weighting W_1 is chosen as a high gain low-pass filter so that performance is emphasized at low frequencies, but becomes less important at high frequencies. A large DC gain is necessary to get reasonable performance. Recall that W_1 weights $SP - P_d$, which has units of rad/N-m, or compliance. For very stiff virtual walls, the compliance is very small. Hence, W_1 may be large.

The weighting on the control effort, W_2, is chosen to be small at low frequencies

so that the control effort may be large, achieving a stiff virtual wall. However, at higher frequencies it increases so that the control effort is lower at high frequencies.

The weight on the strain gauge noise is chosen as a low-pass filter. It was observed that the noise in the encoder measurements was leading to vibration in the response, so by lowering the weight on the strain gauge measurements at high frequencies, the strain gauge readings are used more by the controller at high frequencies. One might expect the strain gauge readings to be noisier than the encoder measurements, but due to the high controller gain on position and the quantization of the encoders, the strain gauge readings actually yield smoother results.

Finally, the encoder noise is weighted by a constant. Note that W_4 must be non-strictly proper to satisfy assumption 4 for the optimization.

The H_∞ controller was implemented at a 1 kHz sampling rate. The results of one test are depicted in Figures 6.37 and 6.38. Figure 6.37 illustrates the axial joint angle as the operator moved the end-effector in and out of the virtual wall. Positive angles lie within the virtual wall. Negative angles are outside the wall. The user probed the wall a dozen times within the 20 second interval, pushing with varying intensity. The resulting torque is depicted as a dotted line in Figure 6.38. The output torque, represented by the solid line, spikes as the user enters the wall, but tracks the applied torque fairly well when the users presses up against the wall i.e. the output torque cancels the applied torque, and the displacement into the wall is small.

However, when the user presses lightly against the wall, there is some vibration. Figure 6.39 shows a magnified view of a time interval over which the user pressed lightly against the wall. The deflection is only a couple of encoder counts and the vibration is not obvious, but it can be felt by the operator and the performance is

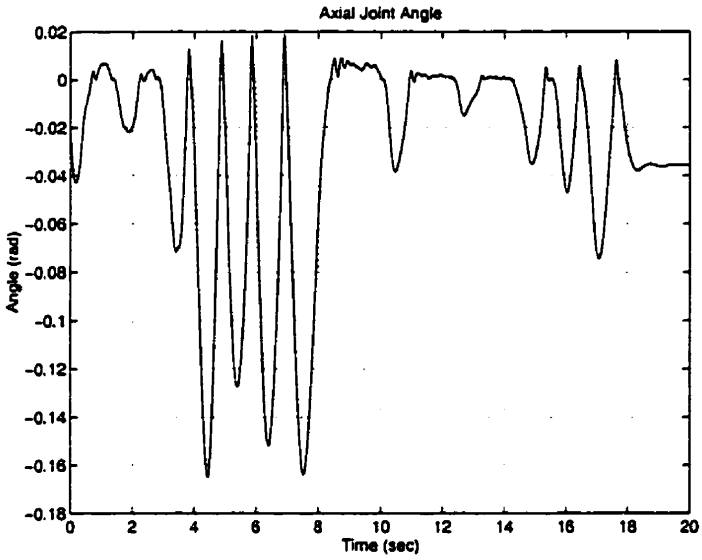


Figure 6.37: Axial Joint Angles

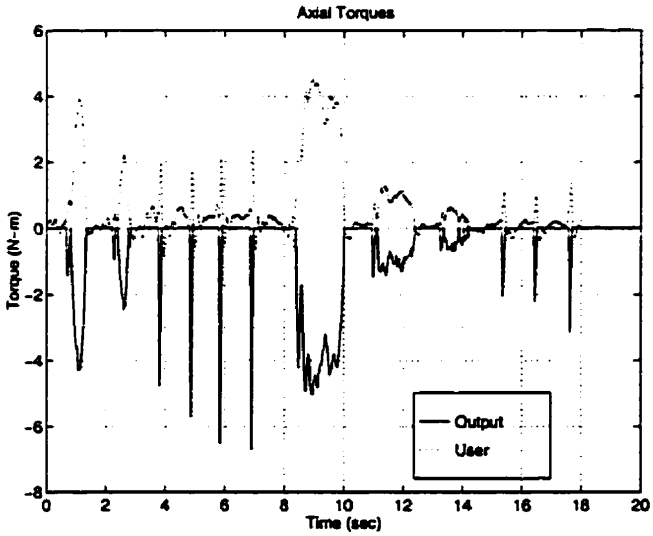


Figure 6.38: Axial Joint Torques

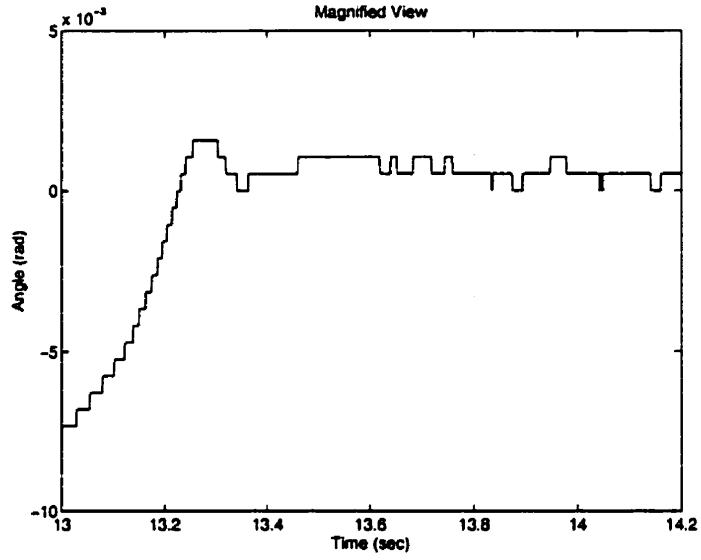


Figure 6.39: Magnified View of Joint Angle

definitely not as good as the state estimator design.

Successive design iterations were unable to completely eliminate the vibration without compromising too much on performance. It is interesting that the vibration can be eliminated by reducing the magnitude of W_3, the weight on the strain gauge noise, so that the controller uses more of the strain gauge measurement in producing its output. However, this increased emphasis on the measured torque eventually introduces a "sticky" feel to the wall. The virtual wall is very stiff but appears to be highly damped. Indeed, once the user enters the wall, even just a little, the force cancellation that makes the wall stiff also prevents the user from exiting the wall. Experiments with walls that are less stiff but have very high damping resulted in a similar problem. Making the output torque unidirectional eliminates the stickiness, but negates the benefits of damping for reducing oscillations. The ideal solution may be to have different gains for motion entering the wall than for motion leaving

the wall. For example, Saludean [98] addresses the issue by generating a brief torque pulse the instant the user enters the virtual wall. Ching and Wang [10] focus on the moment the user exits the virtual wall. A combination of these techniques, along with the non-linear state estimation discussed in section 6.3, may yield excellent results. This combined approach is not investigated here but is left for future research.

An alternative approach to reducing the vibration problem is to increase the weights W_2 and/or W_3 at the frequency of vibration. The increase forces the control effort to be smaller at that frequency. Hence, the controller is less likely to excite the natural frequency of the robot structure. Likewise, the weight W_1 can be notched at the frequency of vibration, leading to reduced performance at that frequency but less vibration. The drawback of modifying the weighting functions to account for the vibration is an increase in the controller order. These approaches were attempted with some success, but the vibration could not be eliminated completely without degrading the performance.

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

A general model has been presented for a flexible beam mounted on an arbitrary robot. Vibrations in two directions are modelled. The model describes the dynamics in a concise form that separates the base robot dynamics from the flexible dynamics in a hierarchical structure. The equations are maintained in a block matrix form that allows various elements, such as the inertia of the flexible beam, to be readily identified. Since the model may also be applied to the transportation of a flexible beam payload by a base robot, the block matrix form also allows the contributions of the payload flexibility to the overall dynamics to be ascertained.

The Jacobian relating end-effector forces to joint torques is also expressed in a concise matrix form. The quadratic nature of the centripetal and Coriolis terms is more obvious than in the traditional $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ representation of these terms. The

matrix form allows an arbitrary number of flexible modes to be represented without resorting to cumbersome summations.

A state-space representation is derived from this general model and linearized about a common operating point. The impact of certain design decisions can be observed directly in the resulting equations. For example, the advantages of gravity balancing become immediately apparent since it grossly simplifies these linearized equations.

Experiments are performed using two different apparatus: a one-dimensional linear actuator with gearing and significant friction, and a three degree-of-freedom five-bar robot. The five-bar robot was designed using a mechatronics approach - its mechanical structure is specifically tailored to yield simpler dynamics through gravity balancing and dynamic decoupling, and to incorporate force sensing directly into the robot structure. The two experimental setups highlight two distinct issues related to haptic interface control – sampling rates and quantization.

Excellent position resolution and high friction combine to yield good virtual walls in the one-dimensional device, without the problems inherent to quantized measurements. Hence, in this device, the sampling period is the key factor in determining stability. The experiments on this device verify the results of Lawrence and Chapel [6]. Limit cycles are observed that disappear at faster sampling rates. The differences in response between a loose grip and tight grip are also verified.

The use of a preload function to address kinetic friction is also investigated and its shortcomings are discussed, particularly with regard to haptic devices. It is concluded that while friction can enhance the stability of a virtual wall implementation, it is a difficult not to compromise the performance of "free motion", even with compensation, unless a force/torque sensor is utilized.

The five-bar robot also has good position resolution, but its low viscous and Coulomb friction, and large torque capability, result in quantization problems when the standard proportional-derivative implicit force control strategy is applied. The large torque capacity is an issue because the stiffer the virtual wall, the more quantization may be a destabilizing factor. Hence there is a tradeoff between position resolution and torque output in any haptic device.

It is demonstrated that this tradeoff can be alleviated somewhat by employing a non-linear state estimator strategy that compensates for Coulomb friction. While the strategy introduced here is shown to outperform the conventional techniques, it must be noted that the strategy is only possible because of the mechatronics approach taken to the robot design. Hence, the merits of a mechatronics design philosophy are verified by the experimental results.

The passivity result of Theorem 2.2.1 and the limit cycle theorem introduced as Theorem 2.2.2 are not applied to the resulting control system because they do not address the problem of quantisation present in the five-bar robot experiments.

The three degrees-of-freedom of the five-bar robot have not been exploited. However, due to the dynamic decoupling inherent in the mechanical system, the axial and vertical joints can be considered independently. Again, this decoupling is a result of the mechatronics approach to the robot design. The dynamics are simple enough that it may be possible to implement the inverse dynamics techniques outlined in section 2.1.5.

Friction in the five-bar robot is identified using a technique that, as far as the author is aware, is new, albeit specific to the problem at hand. However, the technique is very simple and proved to be effective. More general friction identification techniques were attempted but with inconsistent results.

General frameworks for optimal implicit force control and impedance control (or more properly, compliance control) based on H_∞ optimization are also introduced. Controllers are designed and implemented successfully using these frameworks, but prove to be less effective than the state-estimator approach. Both approaches are based on a model-matching paradigm, where the controller is designed to shape the closed-loop transfer function to match a desired compliance.

7.2 Recommendations

There are many potential areas of future research, both experimental and theoretical. These two areas are examined separately.

7.2.1 Experimental Work

The five-bar robot used as the primary research platform for this research suffers from unanticipated structural vibrations. By increasing the rigidity of the robot links it should be possible to eliminate these structural vibrations. Hence, the main source of vibration would be the last link, which is "flexible". However, the modelling results of this work could then be applied and the flexibility of the last link incorporated into the plant model. Alternatively, the flexibility of the link could be increased so that its flexibility dominates the system response. Hence, the model could again be applied. Note however that increasing the link flexibility is likely to degrade performance because the robot's structural vibrational modes would still fall within the bandwidth required for stiff virtual walls. Furthermore, the maximum achievable wall stiffness is limited by the stiffness of the robot.

Visual observation of the robot vibration suggests that the problem may be

linked to torsional vibration in the five-bar parallelogram structure. Hence, the oscillations cause unmodelled coupling between the axial joint being controlled in chapter 6 and the base joint. By implementing a multi-input multi-output compensation scheme to account for this cross-coupling, the vibration may be eliminated more effectively.

Gravity-balancing of the five-bar robot is accomplished by designing the centre of mass to be at the origin of the robot base coordinate frame. The centre of mass location is fixed and does not change with configuration. Hence, by designing a three degree-of-freedom wrist with the same characteristics it should be possible to extend the number of degrees of freedom of the robot to six without jeopardizing the gravity balancing, nor the dynamic decoupling of the axial and vertical joints. (Indeed, by "chaining" versions of the current robot together, it is theoretically possible to build an n-degree-of-freedom gravity-balanced, dynamically decoupled manipulator.)

7.2.2 Theoretical Work

The model of chapter 3 allows for vibration of the flexible link in two directions. However, extension and compression of the beam is not considered. It should be relatively straightforward to add this component to the model.

In the H_∞ designs of chapter 6, the target compliance is not actually included in the final controller – it is only present in the framework employed for the design. As a result of the difficulties encountered while implementing these approaches, however, it is felt that better results may be obtained using an alternative approach. In particular, Yan and Salcudean [92] present a general framework for teleoperation and H_∞ optimization. By replacing the slave robot and its environment with a

model of the desired compliance, it should be possible to apply their results to the problem of simulating a particular compliance, such as a virtual wall. The target compliance would then be incorporated into the controller structure itself. The time delays in the Yan-Salcudean framework may be used to partially account for computational delays in a discrete-time implementation.

Of course, in most, if not all, virtual wall implementations in the literature, the unilateral constraint inherent to virtual walls is not included in the controller design procedure. While the theoretical results of Colgate and Brown [7], for example, may be applied to the final system to verify stability where the same unilateral constraint is applied, it would be useful to include the unilateral constraint in the design itself. For example, Yan and Salcudean [92] suggest the inclusion of passivity constraints into the design procedure through the use of "wave" variables. By using a similar concept to incorporate the passivity results of Colgate and Brown [7] (see Theorem 2.2.1) into the H_∞ design it may be possible to guarantee stability even with the unilateral constraint. The division of the human operator model into an active and a passive component in the work of Yan and Salcudean [92] is also very interesting.

The conservative nature of H_∞ design approaches in general can be addressed using μ-synthesis and it may be necessary to use such an approach to obtain high quality virtual walls.

Also, heuristic approaches could be combined to improve the performance of virtual walls. For example, it would be interesting to combine the non-linear state estimation examined in this work with the braking pulse suggested by Salcudean and Vlaar [98] and the inertial compensation of Ching and Wang [10].

Appendix A

Strain Energy of a Flexible Beam

The following analysis is derived from the investigation of Ghanekar [88]. Consider a beam element as shown in Figure A.1. The radius of curvature of the neutral axis is r. The angle subtended by the beam element is $\delta\theta$. The neutral axis is delimited by the points a and b. The arc length of the neutral axis is δs. Note that since the neutral axis is free of stress, δs is also the length of the undeformed beam element. It is also the original length of the other beam "fibres", such as the arc c-d. The strain in each beam fibre is $\varepsilon(y)$, where y is the radial distance of the fibre from the neutral axis. The strain is defined as the change in length over the original length. Recalling that δs is the original length of the fibre, the arc length of each beam fibre is $(1+\varepsilon(y))\delta s$. But the arc length can also be found

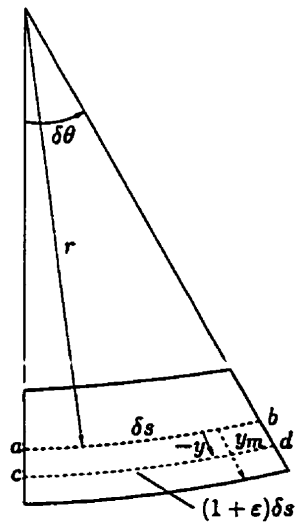


Figure A.1: Strain in Beam

from the subtended angle. Hence:

$$(r - y)\delta\theta = (1 + \varepsilon(y))\delta s$$

Similarly, the arc length of the neutral axis is $\delta s = r\delta\theta$. Consequently:

$$-y\delta\theta = \varepsilon(y)\delta s$$

or equivalently, in the limit:

$$\varepsilon(y) = -y\frac{\partial\theta}{\partial s} \tag{A.1}$$

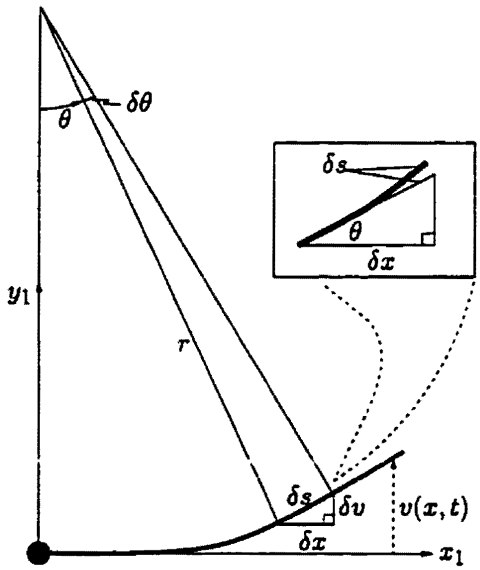


Figure A.2: Element Geometry

Now, consider Figure A.2. The inset represents the beam segment of Figure A.1, where the line of arc length δs is the neutral axis of the segment. The rest of the beam is also shown, in the x_1-y_1 frame. Let δv and δx be defined such that:

$$(\delta s)^2 = (\delta v)^2 + (\delta x)^2$$

and

$$\frac{\delta v}{\delta x} = \tan\theta$$

In the limit, the above expression differentiated with respect to s yields:

$$\sec^2(\theta)\frac{\partial\theta}{\partial s} = \frac{\partial}{\partial x}\left(\frac{\partial v}{\partial x}\right)\frac{\partial x}{\partial s}$$

$$\therefore \quad \frac{\partial\theta}{\partial s} = \cos^2(\theta)\frac{\partial^2 v}{\partial x^2}\frac{\partial x}{\partial s} \tag{A.2}$$

From the inset of Figure A.2 it is evident that in the limit:

$$\frac{\partial x}{\partial s} = \cos(\theta)$$

Thus, from (A.2):

$$\frac{\partial \theta}{\partial s} = \cos^3(\theta)\frac{\partial^2 v}{\partial x^2}$$

For small beam deflections, the angle of curvature, θ, is very small so $\cos(\theta) \approx 1$. Hence:

$$\frac{\partial \theta}{\partial s} \approx \frac{\partial^2 v}{\partial x^2}$$

Substituting this expression into (A.1), the strain $\varepsilon(y)$ becomes:

$$\varepsilon(y) = -y\frac{\partial^2 v}{\partial x^2} \tag{A.3}$$

Let E be the modulus of elasticity of the beam. Denote the length of the beam by l and an element of cross-sectional area as dA at location y. Figure A.3 depicts this elemental area and its position. The potential energy of a beam fibre of unit volume within the segment is $\frac{1}{2}E\varepsilon(y)^2$ and the total potential energy of the flexible link is:

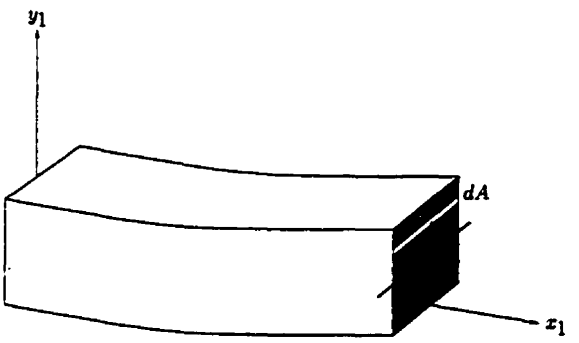


Figure A.3: Beam Cross-Section

$$\begin{aligned} V &= \int_0^l \int \tfrac{1}{2}E\varepsilon(y)^2 dA dx \\ &= \tfrac{1}{2}\int_0^l \int Ey^2 \left(\frac{\partial^2 v}{\partial x^2}\right)^2 dA dx \\ &= \tfrac{1}{2}\int_0^l EI \left(\frac{\partial^2 v}{\partial x^2}\right)^2 dx, \qquad \text{where } I = \int y^2 dA \end{aligned}$$

The term, I, is called the cross-sectional area moment of inertia. Using $(\cdot)'$ to denote differentiation with respect to x, the potential strain energy may be written in concise form as:

$$V = \frac{1}{2}\int_0^l EI(v'')^2 dx \tag{A.4}$$

Appendix B

Clamped-Free Beam Eigenfunctions

This appendix describes the Euler-Bernoulli beam model and applies it to the solution of the mode shapes for a clamped-free beam. Orthogonality and normalization of the mode shapes is discussed.

B.1 Euler-Bernoulli Beam

To derive a reasonable modal shape function, $\Phi(x)$, consider the free-body diagram of an unforced beam element, as depicted in Figure B.1. The width of the beam element is dx and the beam element is located at a position, x, along the axis x_1 and a displacement v (refer back to Figure 3.1). The quantity, q_0, of Figure 3.1 is assumed to be identically equal to zero. The mass of the beam element is dm. Only transverse vibrations are considered. The expression $\dot{()}$ refers to differentiation with respect to time. Let $()'$ denote partial differentiation with respect to x.

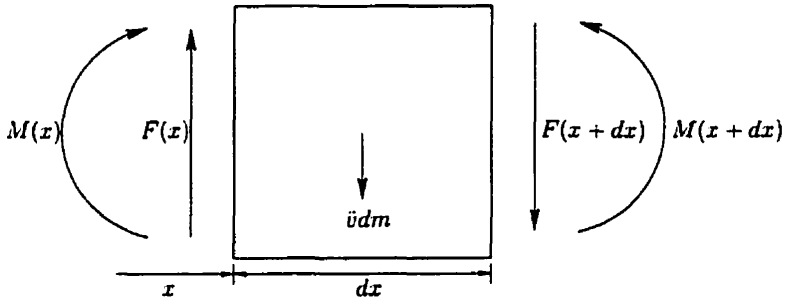


Figure B.1: Bernoulli Beam Element

The inertial forces due to acceleration of the beam element in the displacement, v, direction are $\ddot{v}dm$. The moments experienced by the beam element are $M(x)$ and $M(x+dx)$. Also, the beam element undergoes shear forces of $F(x)$ and $F(x+dx)$. Note that other factors, such as rotary inertia and shearing deformations are ignored; plane faces of the beam element are assumed to remain plane [82]. These assumptions are standard and are reasonable for a beam whose cross-section is small with respect to its length. Such a beam is known as an Euler-Bernoulli beam.

Using a Taylor's series expansion for the moment and shear force at $x+dx$ and applying Newton's law in the transverse direction produces the following two relationships:

$$\ddot{v}dm = F(x) - F(x+dx) \approx F(x) - (F(x) + F'(x)dx) = -F'(x)dx \qquad \text{(B.1)}$$

$$F(x) \approx -M'(x) \qquad \text{(B.2)}$$

It will be presumed that the cross-sectional area, A, of the beam is constant. Substituting equation (B.2) into equation (B.1), and replacing dm with ρdx, where ρ is the linear mass density of the beam element:

$$\ddot{v}\rho = -M''(x) \qquad \text{(B.3)}$$

Now, the moment $M(x)$ is generated by internal stresses in the beam. Referring to Figures 3.2 and A.3, the moment acting on an elemental area dA, a distance y from the neutral axis, under a stress $\sigma(y)$ is $y\sigma(y)dA$. (The quantity $\sigma(y)dA$ represents the force due to the internal stress). Thus, the moment acting on the beam element is:

$$M(x) = \int y\sigma(y)dA$$

But $\sigma(y) = E\varepsilon(y)$, where E is the elastic modulus of the beam element. Hence from this relationship and equation (A.3), the magnitude of the moment is:

$$\begin{aligned} M(x) &= \int Ey^2v''dA \\ &= EIv'' \qquad \text{where } I = \int y^2dA \text{ as before} \end{aligned} \tag{B.4}$$

Substituting this expression for the moment into equation (B.3):

$$\ddot{v}\rho = -EIv''''(x)$$

Using the separation of variables assumption, $v(x,t) = \Phi(x)Q(t)$, the above equation becomes:

$$EI\Phi''''(x)Q(t) = -\rho\Phi(x)\ddot{Q}(t)$$

$$\therefore \left(\frac{EI}{\rho}\right)\frac{\Phi''''(x)}{\Phi(x)} = -\frac{\ddot{Q}(t)}{Q(t)}$$

The two sides can only be the same if they equal a constant. Denote this constant by ω^2. Then:

$$\left(\frac{EI}{\rho}\right)\frac{\Phi''''(x)}{\Phi(x)} = -\frac{\ddot{Q}(t)}{Q(t)} = \omega^2$$

or as two homogeneous equations:

$$\ddot{Q}(t) + \omega^2Q(t) = 0 \tag{B.5}$$

$$\Phi''''(x) - \frac{\omega^2}{a^2}\Phi(x) = 0, \qquad \text{where } a^2 = \frac{EI}{\rho} \tag{B.6}$$

Since the time and spatial components have been separated, these two equations may be treated as ordinary differential equations. A particular solution to (B.5) is $Q(t) = e^{j\omega t}$. Define:

$$p^2 = \frac{\omega}{a} \qquad \text{or} \qquad p^4 = \frac{\omega^2 \rho}{EI} \tag{B.7}$$

The solution to equation (B.6) takes the form[1]:

$$\Phi(x) = C_1 \cos(px) + C_2 \sin(px) + C_3 \cosh(px) + C_4 \sinh(px) \tag{B.8}$$

The constants C_1, C_2, C_3 and C_4 may be found by applying the boundary conditions for the overall beam.

Boundary Conditions for the Clamped-Free Beam

Consider the flexible link depicted in Figure 3.1. The beam is essentially "clamped" to the hub at one end, and free to move at the other end. Hence, this configuration is known as a clamped-free beam. The boundary conditions at the hub are trivial. Clearly, the displacement of the beam from the x_1-axis is always zero. Similarly, as the x_1-axis is always tangential to the link, the slope of the beam at the hub is also zero. These two conditions may be expressed mathematically as:

$$v(0,t) = 0 \qquad \text{(displacement is zero)}$$
$$v'(0,t) = 0 \qquad \text{(slope is zero)}$$

Applying the separation of variables assumption, these conditions may also be expressed in terms of the modal shape function, $\Phi(x)$, as:

$$\Phi(0) = 0 \qquad \text{(displacement is zero)}$$
$$\Phi'(0) = 0 \qquad \text{(slope is zero)}$$

[1]This solution is easily derived using Laplace transforms

The boundary conditions at the free end ($x \approx l$) arise from the lack of both a moment and a shear force at the end of the beam. From equation (B.4) and the separation of variables assumption, the moment is:

$$M(x) = EI\Phi''(x)Q(t)$$

and, similarly, from equation (B.2), the shear force is:

$$F(x) = -EI\Phi'''(x)Q(t)$$

For these two quantities to be zero at the end of the beam, for all time,

$$\Phi''(l) = 0 \qquad \text{(moment is zero)}$$

$$\Phi'''(l) = 0 \qquad \text{(shear force is zero)}$$

Note that the boundary conditions for other beam configurations, such as pinned-free or simply-supported, are similar; in each case, the modal shape function and/or one or more of its derivatives will be zero at each end of the beam.

Mode Shapes for the Clamped-Free Beam

Evaluating these clamped-free boundary conditions in equation (B.8) yields the four equations:

$$\Phi(0) = 0 \implies C_1 + C_3 = 0$$

$$\Phi'(0) = 0 \implies C_2 + C_4 = 0$$

$$\Phi''(l) = 0 \implies -C_1\cos(pl) - C_2\sin(pl) + C_3\cosh(pl) + C_4\sinh(pl)$$

$$\Phi'''(l) = 0 \implies C_1\sin(pl) - C_2\cos(pl) + C_3\sinh(pl) + C_4\cosh(pl)$$

The only solution to this set of linear homogeneous equations in C_1, C_2, C_3 and C_4 is either the trivial solution $C_1 = C_2 = C_3 = C_4 = 0$ or the restriction:

$$\cos(pl)\cosh(pl) = -1 \qquad (\text{and } C_3 = -C_1, C_4 = -C_2) \tag{B.9}$$

on the mode shape "frequency", p. Note also that given equation (B.9), the constants C_1, C_2, C_3 and C_4 can all be scaled by an arbitrary factor, α, and equations (B.6) and (B.9) are still satisfied. Thus, the modal shape function $\Phi(x)$ can be normalised by an arbitrary constant. Normalization will be discussed in more detail in a later section.

Total Solution for the Clamped-Free Beam

The particular solution for the mode shape of equation (B.8) was derived based on the assumption that $Q(t) = e^{j\omega t}$ for some angular frequency ω. Recalling equation (B.7), the derivation of $\Phi(x)$ above illustrates that there are some restrictions, (B.9), on this angular frequency based on the boundary conditions. However, equation (B.9) has an infinite number of solutions. For each solution, p_i, of equation (B.9), there is a corresponding mode shape function, $\phi_i(x)$. Thus, the total solution for the displacement $v(x,t)$ may be written as the sum of all the particular solutions. Let $q_i(t) = e^{j\omega_i t}$, where $p_i^2 = \frac{\omega_i}{a}$ satisfies equation (B.9). Then the total response is:

$$v(x,t) = \sum_{i=1}^{\infty} \phi_i(x) q_i(t) \tag{B.10}$$

Orthogonality Conditions for the Modal Shape Functions

Since the product of the mode shapes often arises during the process of developing the dynamic equations of a flexible system, it is revealing to examine the relationship between two mode shapes, $\phi_i(x)$ and $\phi_j(x)$. Equation (B.6) must be satisfied by both modal shape functions. Write:

$$\lambda_i = \frac{\omega^2}{a^2} \tag{B.11}$$

Then $\phi_i(x)$ and $\phi_j(x)$ satisfy:

$$\phi_i''''(x) = \lambda_i \phi_i(x) \tag{B.12}$$

$$\phi_j''''(x) = \lambda_j \phi_j(x) \tag{B.13}$$

Multiplying equation (B.12) by $\phi_j(x)$, and multiplying equation (B.13) by $\phi_i(x)$ and then integrating across the length of the beam:

$$\int_0^l \phi_i'''' \phi_j dx = \lambda_i \int_0^l \phi_i \phi_j dx$$
$$\int_0^l \phi_j'''' \phi_i dx = \lambda_j \int_0^l \phi_j \phi_i dx$$

The dependence on x has been omitted for clarity. Integrating the left-hand sides by parts twice:

$$\int_0^l \phi_i'' \phi_j'' dx + \phi_i''' \phi_j \Big|_0^l - \phi_i'' \phi_j' \Big|_0^l = \lambda_i \int_0^l \phi_i \phi_j dx \tag{B.14}$$

$$\int_0^l \phi_j'' \phi_i'' dx + \phi_j''' \phi_i \Big|_0^l - \phi_j'' \phi_i' \Big|_0^l = \lambda_j \int_0^l \phi_j \phi_i dx \tag{B.15}$$

Substituting the boundary conditions for the clamped-free beam:

$$\int_0^l \phi_i'' \phi_j'' dx = \lambda_i \int_0^l \phi_i \phi_j dx \tag{B.16}$$

$$\int_0^l \phi_j'' \phi_i'' dx = \lambda_j \int_0^l \phi_j \phi_i dx \tag{B.17}$$

Subtracting equation (B.17) from (B.16):

$$(\lambda_i - \lambda_j) \int_0^l \phi_i \phi_j dx = 0 \tag{B.18}$$

Thus, for $i \neq j$, equations (B.18) and (B.16) yield the orthogonality relationships:

$$\begin{aligned} \int_0^l \phi_i \phi_j dx &= 0 \\ \int_0^l \phi_i'' \phi_j'' dx &= 0 \end{aligned} \tag{B.19}$$

Normalization of the Modal Shape Functions

In the prior section, Mode Shapes for the Clamped-Free Beam, the observation is made that the mode shape can be scaled by an arbitrary constant without violating the clamped-free boundary conditions. The orthogonality conditions of equation (B.19) are also independent of the scaling of the modal shape functions, ϕ_i and ϕ_j. Thus, for convenience, the mode shapes will be normalised by adding the following constraint, for each i:

$$\int_0^l \phi_i^2 dx = 1 \tag{B.20}$$

This normalization constraint is useful in analysis for simplifying the Euler-Lagrange dynamic equations.

Appendix C

Strain Measurements

Consider the cross-section of the flexible beam depicted in Figure C.1.

Define the following quantities:

h = height of beam cross-section

b = breadth of beam cross-section

$\epsilon_v(x, y, t)$ = strain at displacement $(x, y, 0)$

$\epsilon_w(x, z, t)$ = strain at displacement $(x, 0, z)$

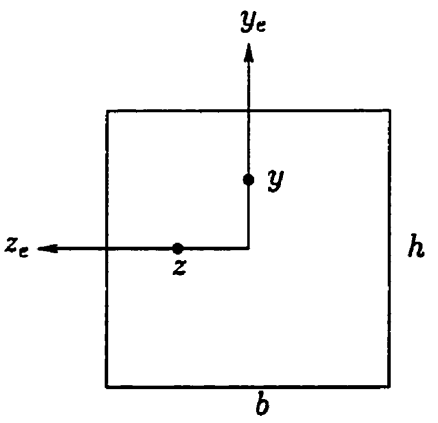


Figure C.1: Cross-Section

The relationship between the strains and the beam deflections are:

$$\epsilon_v(x, y, t) = -yv(x, t)''$$

$$\epsilon_w(x, z, t) = -zw(x, t)''$$

Suppose strain gauges are mounted on each side of the beam. Taking the difference between the measurements on opposing sides of the beam results in the amplified

strain measurements:

$$\epsilon_v(x,t) = \varepsilon_v(x,-h/2,t) - \varepsilon_v(x,h/2,t) = hv(x,t)'' \tag{C.1}$$

$$\epsilon_w(x,t) = \varepsilon_w(x,-b/2,t) - \varepsilon_w(x,b/2,t) = bw(x,t)'' \tag{C.2}$$

Hence, from (3.40) and (3.41), the amplified strains may be written:

$$\epsilon_v(x,t) \approx h\varphi''(x)^{\mathrm{T}}\nu(t) \tag{C.3}$$

$$\epsilon_w(x,t) \approx b\psi''(x)^{\mathrm{T}}\eta(t) \tag{C.4}$$

The quantities $h\varphi''(x)^{\mathrm{T}}$ and $b\psi''(x)^{\mathrm{T}}$ are constant with respect to time, and thus may be precomputed based on the positions of the strain gauges along the flexible beam. Also note that $\nu(t)$ and $\eta(t)$ are independent of the position of each strain gauge.

Suppose $4n$ strain gauges are mounted at locations $x_i, i = 1, \ldots, n$ along the beam. Using equation (C.3), the amplified strains in the v-direction at each location, $\mathbf{E}_v$, are:

$$\mathbf{E}_v(t) \approx \begin{bmatrix} h\varphi''(x_1)^{\mathrm{T}} \\ \vdots \\ h\varphi''(x_n)^{\mathrm{T}} \end{bmatrix} \nu(t)$$

Define the constant $n \times n$ matrix:

$$\mathcal{K}_v \approx h \begin{bmatrix} \varphi''(x_1)^{\mathrm{T}} \\ \vdots \\ \varphi''(x_n)^{\mathrm{T}} \end{bmatrix}$$

Then

$$\mathbf{E}_v(t) = \mathcal{K}_v\nu(t) \tag{C.5}$$

The matrix $\mathcal{K}_v$ is a constant dependent only upon the locations of the strain gauges and the beam cross-section. Equation (C.5) indicates that the generalized coordinates of the flexible beam may be computed readily from the strain measurements *provided* $\mathcal{K}_v$ *is non-singular.* Thus, the strain gauges must be placed such that $\det(\mathcal{K}_v) \neq 0$. This restriction ensures, for example, that the strain gauges are not all placed at the nodes of the same vibrational mode. Furthermore, the location of the strain gauges may be optimized by solving the non-linear optimization problem:

$$\mathbf{x}^* = \arg\max_{|\mathcal{K}_v| \neq 0} \|\mathcal{K}_v\|$$

Note that if $\mathcal{K}_v$ is non-singular then the strain measurements may be used as generalized coordinates. Assuming small deflections, the tip deflection, $v(l,t)$, may be derived in this case as:

$$v(l,t) = \left(\varphi(l)^{\mathrm{T}} \mathcal{K}_v^{-1}\right) \mathbf{E}_v(t)$$

The quantity $\varphi(l)^{\mathrm{T}} \mathcal{K}_v^{-1}$ is constant and may be precomputed off-line. In a similar fashion, a matrix $\mathcal{K}_w$ may be defined whereby $\mathbf{E}_w = \mathcal{K}_w \eta$.

Appendix D

Common Matrix Operations

This appendix merely enumerates the well-known properties of rotation matrices and skew-symmetric matrices that are used in Chapter 3. For a detailed discussion of these properties and definitions, refer to [12].

D.1 Rotation Matrices

Definition D.1.1 *Rotation Matrix*

A matrix R is a proper rotation matrix *if and only if it satisfies:*

i. $R^T R = I$

ii. $\det(R) = +1$

Let $SO(3)$, referring to Special Orthogonal Group, *denote the set of all rotation matrices in $\Re^{3\times 3}$.*

D.2 Skew-Symmetric Matrices

Definition D.2.1 *Skew-Symmetric Matrix*

A matrix A is skew-symmetric *if and only if it satisfies:*

$$A + A^T = 0.$$

Let $SS(3)$ denote the set of all skew-symmetric matrices in $\Re^{3\times 3}$.

Definition D.2.2 *Skew-Symmetric Operator,* $\mathrm{S}(\cdot)$

By virtue of the definition of $SS(3)$, all matrices in $SS(3)$ have only three independent elements. Hence, every matrix $A \in SS(3)$ may be written in the form:

$$A = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

Alternatively, let $\mathbf{a} = (a_1 \quad a_2 \quad a_3)^T$. *Define the skew-symmetric operator* $\mathrm{S}(\cdot)$ *by:*

$$\mathrm{S}(\mathbf{a}) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{D.1}$$

Clearly every $A \in SS(3)$ may be written as $\mathrm{S}(\mathbf{a})$ *for some* $\mathbf{a} \in \Re^3$.

D.2.1 Properties

The skew-symmetric operator, $\mathrm{S}(\cdot)$, has a number of useful properties. The properties enumerated below are particularly relevant for the ensuing derivation of the

general dynamic equations. Let $\mathbf{a}, \mathbf{b}, \boldsymbol{\omega}(t) \in \Re^3$ and $\alpha, \beta \in \Re$. Let $R \in SO(3)$. Let $\dot{()}$ denote differentiation with respect to time.

Linearity:	$\mathrm{S}(\alpha\mathbf{a} + \beta\mathbf{b}) = \alpha\,\mathrm{S}(\mathbf{a}) + \beta\,\mathrm{S}(\mathbf{b})$		(D.2)
Vector product:	$\mathrm{S}(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b}$		(D.3)
Anti-commutativity:	$\mathrm{S}(\mathbf{a})\mathbf{b} = -\mathrm{S}(\mathbf{b})\mathbf{a}$	(as $\mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a}$)	(D.4)
Distributivity:	$R(\mathbf{a} \times \mathbf{b}) = R\mathbf{a} \times R\mathbf{b}$		(D.5)
Transformation:	$RS(\mathbf{a})R^{\mathrm{T}} = \mathrm{S}(R\mathbf{a})$	(from (D.5) and (D.3))	(D.6)
Angular velocity:	$\dot{R} = \mathrm{S}(\boldsymbol{\omega})R$		(D.7)

Bibliography

[1] Roger A. Horn and Charles R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, New York, 1991. QA188.H664.

[2] John C. Doyle, Keith Glover, Pramod P. Khargonekar, and Bruce A. Francis. State-space solutions to standard H_2 and H_∞ control problems. *IEEE Transactions on Automatic Control*, 34(8):831–847, August 1989.

[3] John C. Doyle, Bruce A. Francis, and Allen R. Tannenbaum. *Feedback Control Theory.* Maxwell Macmillan Canada, Toronto, 1992. ISBN 0-02-330011-6, TJ216.D69.

[4] J. M. Maciejowski. *Multivariable Feedback Design.* Addison-Wesley Publishing Company, Don Mills, Ontario, 1989.

[5] Daniel E. Whitney. Historical perspective and state of the art in robot force control. *The International Journal of Robotics Research*, 6(1):3–14, April 1987. PER TJ211.I5x.

[6] Dale A. Lawrence and Jim D. Chapel. Performance trade-offs for hand controller design. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 3211–3216, 1994. TJ211.I49x.

[7] J. Edward Colgate and J. Michael Brown. Factors affecting the Z-width of a haptic display. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 3205–3210, 1994. TJ211.I49x.

[8] Gregory L. Long and Curtis L. Collins. A pantograph linkage parallel platform master hand controller for force-reflection. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, volume 1, pages 390–395, Nice, France, May 1992. ISBN 0-8186-2720-4 (TJ211.I49x).

[9] Yukihiro Hirata and Makoto Sato. 3-dimensional interface device for virtual work space. In *Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 889–896, Raleigh, N.C., July 1992. TJ210.3.I44.

[10] Mennas Ching and David W. L. Wang. A five-bar-linkage force reflecting interface for a virtual reality system. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3012–3017, Albuquerque, NM, April 1997. TJ211.I49x.

[11] Haruhiko Asada and Jean-Jacques E. Slotine. *Robot Analysis and Control.* John Wiley & Sons, Inc., Toronto, 1986. ISBN 0-471-83029-1, TJ211.A79.

[12] Mark W. Spong and M[athukumalli] Vidyasagar. *Robot Dynamics & Control.* John Wiley & Sons, Inc., Toronto, 1989. ISBN 0-471-61243-X, TJ211.4.S66.

[13] Joseph Borowiec and Anthony Tzes. Adaptive sub-optimal explicit force control of a one link flexible manipulator. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 57–1, pages 159–166, 1995. TJ212.2.P76x.

[14] L. D. Joly, C. Andriot, and V. Hayward. Mechanical analogies in hybrid position/force control. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 1, pages 835–840, Albuquerque, NM, April 1997. TJ211.I49x.

[15] Dragoljub T. Šurdilović. Contact transition stability in the impedance control. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 1, pages 847–852, Albuquerque, NM, April 1997. TJ211.I49x.

[16] Seul Jung, T. C. Hsia, and R. G. Bonitz. On robust impedance force control of robot manipulators. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 3, pages 2057–2062, Albuquerque, NM, April 1997. TJ211.I49x.

[17] Massimo Bergamasco. Design of hand force feedback systems for glove-like advanced interfaces. In *IEEE International Workshop on Robot and Human Communication*, pages 286–293, 1992. TJ210.3.I4424.

[18] Sukhan Lee and Hahk Sung Lee. An advanced teleoperator control system: Design and evaluation. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, pages 859–864, Nice, France, May 1992. ISBN 0-8186-2720-4, TJ211.I49x.

[19] J. Edward Colgate and Gerd Schenkel. Passivity of a class of sampled-data systems: Application to haptic interfaces. In *Proceedings of the American Control Conference*, volume 3, pages 3236–3240, June 1994. TJ212.A63x.

[20] J. E[dward] Colgate. Coordinate transformations and logical operations for minimizing conservativeness in coupled stability criteria. *Journal of Dy-*

namic Systems, Measurement, and Control, 116:643–649, December 1994. TJ212.J62x.

[21] J. Edward Colgate, M. C. Stanley, and G. G. Schenkel. Dynamic range of achievable impedances in force reflecting interfaces. *Telemanipulator Technology and Space Robotics*, 1993. Boston, SPIE.

[22] Lonnie Love and Wayne Book. Contact stability analysis of virtual walls. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 57-2, pages 689–694, San Francisco, CA, November 1995. TJ212.2.P76x.

[23] J. C. Tsai and J. Edward Colgate. Stability of discrete-time systems with unilateral nonlinearities. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 57-2, pages 695–702, San Francisco, CA, November 1995. TJ212.2.P76x.

[24] R. Brent Gillespie and Mark R. Cutkosky. Stable user-specific haptic rendering of the virtual wall. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 58, pages 397–406, 1996. TJ212.2.P76x.

[25] P. J. Berkelman, R. L. Hollis, and S. E. Salcudean. Interacting with virtual environments using a magnetic levitation haptic interface. In *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 117–122, Pittsburgh, Pennsylvania, August 1995.

[26] R. E. Ellis, N. Arkar, and M. A. Jenkins. Numerical methods for the haptic presentation of contact: Theory, simulations, and experiments. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 58, pages 413–420, 1996. TJ212.2.P76x.

[27] D. Stewart. A platform with six degrees of freedom. In *Proceedings of the Institute of Mechanical Engineering*, volume 180, pages 371–386, 1965–1966.

[28] Haruo Noma and Hiroo Iwata. Presentation of multiple dimensional data by 6 D.O.F. force display. In *Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1495–1500, Yokohama, Japan, July 1993. TJ211.3.I443.

[29] Mark D. Bryfogle, Charles C. Nguyen, Sami S. Antrazi, and Peter C. Chiou. Kinematics and control of a fully parallel force-reflecting hand controller for manipulator teleoperation. *Journal of Robotic Systems*, 10(5):745–766, July 1993. TJ211.J58x.

[30] Tetsuya Morizono, Kazuhiro Kurahashi, and Sadao Kawamura. Realization of a virtual sports training system with parallel wire mechanism. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3025–3030, Albuquerque, NM, April 1997. TJ211.I49x.

[31] J. F. Jansen, R. L. Kress, and S. M. Babcock. Controller design for a force-reflecting teleoperator system with kinematically dissimilar master and slave. *Journal of Dynamic Systems, Measurement and Control*, 114:641–649, December 1992.

[32] John W. Hill, Philip S. Green, Joel F. Jansen, Yonael Gorfu, and Ajit S. Shah. Telepresence surgery demonstration system. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2302–2307, 1994. TJ211.I49x.

[33] S. E. Salcudean and J. Yan. Towards a force-reflecting motion-scaling system for microsurgery. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2296–2301, 1994. TJ211.I49x.

[34] T. Sato, J. Ichikawa, M. Mitsuishi, and Y. Hatamura. A new micro-teleoperation system employing a hand-held force-feedback pencil. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 1728–1733, 1994. TJ211.I49x.

[35] Greg R. Luecke and Young-Ho Chai. Haptic interaction using a PUMA560 and the ISU force reflecting exoskeleton system. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 1, pages 106–111, Albuquerque, NM, April 1997. TJ211.I49x.

[36] Vincent Hayward, Jehangir Choksi, Gonzalo Lanvin, and Christophe Ramstein. Design and multi-objective optimization of a linkage for a haptic interface. In Jadran Lenarčič and Bahram Ravani, editors, *Advances in Robot Kinematics and Computational Geometry*, chapter 9, pages 359–368. Kluwer Academic Publishers, Boston, 1994.

[37] A. J. Kelley and S. E. Salcudean. On the development of a force-feedback mouse and its integration into a graphical interface. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 55-1, pages 287–294, 1994. TJ212.2.P76x.

[38] Thomas H. Massie and J. Kenneth Salisbury. The PHANToM haptic interface: A device for probing virtual objects. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 55–1, pages 295–301, 1994. TJ212.2.D94.

[39] V[incent] Hayward, P. Gregorio, O. Astley, S. Greenish, M. Doyon, L. Lessard, J. McDougall, L. Sinclair, S. Boelen, X. Chen, J. P. Demers, and J. Poulin. Freedom-7: A high fidelity seven axis haptic device with application to surgical training. In *Proceedings of the 5th International Symposium on Experimental Robotics, ISER'97*, Barcelona, Spain, June 1997.

[40] Daniel Gomez, Grigore Burdea, and Noshir Langrana. Modeling of the "Rutgers Master II" haptic display. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 57–2, pages 727–733, 1995. TJ212.2.P76x.

[41] Darwin G. Caldwell, N. Tsagarakis, and A. Wardle. Mechano thermo and proprioceptor feedback for integrated haptic feedback. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 3, pages 2491–2496, Albuquerque, NM, April 1997. TJ211.I49x.

[42] Costas Tzafestas and Philippe Coiffet. Computing optimal forces for generalized kinesthetic feedback on the human hand during virtual grasping and manipulation. In *Proceedings of the American Control Conference*, volume 1, pages 118–123, Albuquerque, NM, June 1997. TJ212.2.A63x.

[43] Parris Wellman and Robert D. Howe. Towards realistic vibrotactile display in virtual environments. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 57–2, pages 713–718, 1995. TJ212.2.P76x.

[44] Gerald P. Roston and Thomas Peurach. A whole body kinesthetic display device for virtual reality applications. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3006–3011, Albuquerque, NM, April 1997. TJ211.I49x.

[45] Tsuneo Yoshikawa and Akihiro Nagura. A touch and force display system for haptic interface. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3018–3024, Albuquerque, NM, April 1997. TJ211.I49x.

[46] Witaya Wannusuphoprasit, R. Brent Gillespie, J. Edward Colgate, and Michael A. Peshkin. Cobot control. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3571–3576, Albuquerque, NM, April 1997. TJ211.I49x.

[47] David [W. L.] Wang. Comparison of control strategies for the single flexible link. *Fields Institute Communications*, 2, 1993.

[48] Jr. Robert H. Cannon and Eric Schmitz. Initial experiments on the end-point control of a flexible one-link robot. *International Journal of Robotics Research*, 3(3), 1994. Massachusetts Institute of Technology.

[49] A. Truckenbrodt. Truncation problems in the dynamics and control of flexible mechanical systems. In *Proceedings of IFAC Control Science and Technology Conference*, volume 4, pages 1909–1914, Kyoto, Japan, August 1981. TJ212.C68 1982.

[50] Jeff K. Pieper. Optimal control of a flexible manipulator and controller order reduction. In *Proceedings of the American Control Conference*, volume 3, pages 1422–1423, Albuquerque, NM, June 1997. TJ212.2.A63x.

[51] Mauro Rossi, Kai Zuo, and David [W. L.] Wang. Issues in the design of passive controllers for flexible link robots. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, pages 321–326, Atlanta, Georgia, 1993. TJ211.I49x.

[52] N. C. Singer and W. P. Seering. Preshaping command inputs to reduce system vibration. *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, 112(1):76–82, March 1990. ISSN: 0022-0434.

[53] K[ai] Zuo and David [W. L.] Wang. Closed-loop shared-input control of a class of manipulators with a single flexible link. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, volume 1, Nice, France, May 1992. ISBN 0-8186-2720-4 (TJ211.I49x).

[54] Marc Bodson. An adaptive algorithm for the tuning of two input shaping methods. In *Proceedings of the American Control Conference*, volume 3, pages 1340–1344, Albuquerque, NM, June 1997. TJ212.2.A63x.

[55] David P. Magee, David W. Cannon, and Wayne J. Book. Combined command shaping and inertial damping for flexure control. In *Proceedings of the American Control Conference*, volume 3, pages 1330–1334, Albuquerque, NM, June 1997. TJ212.2.A63x.

[56] Kuldip S. Rattan and Vinvente Feliu. Feedforward control of flexible manipulators. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, volume 1, pages 788–793, Nice, France, May 1992. ISBN 0-8186-2720-4 (TJ211.I49x).

[57] A. S. Morris and A. Madani. Quadratic optimal control of a two-flexible-link robot manipulator. *Robotica*, 16:97–108, February 1998. part 1, TJ210.2.R6.

[58] Marco A. Artega. On the properties of a dynamic model of flexible robot manipulators. *Journal of Dynamic Systems, Measurement and Control*, 120(1):8–14, March 1998. TJ212.J62x.

[59] Andrei Tchernychev, Athanasios Sideris, and Jie Yu. Constrained H_∞ control of an experimental flexible link. In *Proceedings of the ASME Dynamic Systems and Control Division*, volume 57-1, pages 185–192, 1995. TJ212.2.P76x.

[60] Eric Kubica and David [W. L.] Wang. A fuzzy control strategy for a single flexible link robot. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, pages 236–241, Atlanta, Georgia, 1993. TJ211.I49x.

[61] E. Bayo, R. Moraghar, and M. Medies. Inverse dynamics of a single flexible link robot, analytical and experimental results. *International Journal of Robotics and Automation*, 3(3), 1988.

[62] H. A. Talebi, K. Khorasani, and R. V. Patel. Experimental evaluation of neural network based controllers for tracking the tip position of a flexible-link manipulator. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3300–3305, Albuquerque, NM, April 1997. TJ211.I49x.

[63] M. R. Rokui and K. Khorasani. Adaptive tracking control of a flexible link manipulator using a discrete-time nonlinear model. In *Proceedings of the American Control Conference*, volume 3, pages 1848–1853, Albuquerque, NM, June 1997. TJ212.2.A63x.

[64] M. Moallem, R. V. Patel, and K. Khorasani. Experimental result for nonlinear decoupling control of flexible multi-link manipulators. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3142–3147, Albuquerque, NM, April 1997. TJ211.I49x.

[65] M. Moallem, K. Khorasani, and R. V. Patel. An inverse dynamics sliding control technique for flexible multi-link manipulators. In *Proceedings of the*

American Control Conference, volume 3, pages 1407–1416, Albuquerque, NM, June 1997. TJ212.2.A63x.

[66] G. Zhu and S. S. Ge. A model-free approach for regulation of multi-link flexible robots. In *Proceedings of the American Control Conference*, volume 3, pages 1417–1421, Albuquerque, NM, June 1997. TJ212.2.A63x.

[67] M. D. Majors and R. J. Richards. Time-optimal transportation of flexible payloads. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3455–3460, Albuquerque, NM, April 1997. TJ211.I49x.

[68] A. G. Barto, S. J. Bradtke, and S. P. Singh. Learning to act using real-time dynamic programming. Technical Report UM-CS-1993-002, University of Massachusetts, Amherst, 1994.

[69] S. W. Tilley, Jr. R. H. Cannon, and Ray Kraft. End point force control of a very flexible manipulator with a fast end effector. *Robotics: Theory and Applications*, 3:1–9, December 1986. ASME, New York.

[70] Fumitoshi Matsuno, Toshio Asano, and Yoshiyuki Sakawa. Modeling and quasi-static hybrid position/force control of constrained planar two-link flexible manipulators. *IEEE Transactions on Robotics and Automation*, 10(3):287–297, June 1994.

[71] F[umitoshi] Matsuno, T[oshio] Asano, and Y[oshiyuki] Sakawa. Erratum: Corrections to 'modeling and quasi-static hybrid position/force control of constrained planar two-link flexible manipulators'. *IEEE Transactions on Robotics and Automation*, 11(1):167, February 1995. ISSN 1042-296X.

[72] Fumitoshi Matsuno, Satoshi Umeyama, and Shozaburo Kasai. Experimental study on robust force control of a flexible arm with a symmetric rigid tip body. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 3136–3141, Albuquerque, NM, April 1997. TJ211.I49x.

[73] Tsuneo Yoshikawa, Koh Hosoda, Kensuke Harada, Atsushi Matsumoto, and Hiroki Murakami. Hybrid position/force control of flexible manipulators by a macro-micro manipulator system. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2125–2130, San Diego, CA., 1994. TJ211.I49x.

[74] Jae Y. Lew. Contact control of flexible micro/macro-manipulators. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 2850–2855, Albuquerque, NM, April 1997. TJ211.I49x.

[75] B. C. Chiou and M. Shahinpoor. Dynamic stability analysis of a two-link force-controlled flexible manipulator. *Journal of Dynamic Systems, Measurement and Control*, 112:661–666, December 1990. PER TJ212.J62x.

[76] F. Pfeiffer, K. Richter, and H. Wapenhans. Elastic robot trajectory planning with force control. *Modelling the Innovation: Communications, Automation and Information Systems*, 1990. Elsevier Science Publishers B. V., ©IFIP 1990.

[77] Jin-Soo Kim, Kuniaki Suzuki, Mitsuhiro Yamano, and Masaru Uchiyama. Vibration suppression control of constrained spatial flexible manipulators. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 2831–2837, Albuquerque, NM, April 1997. TJ211.I49x.

[78] Ling-Hui Chang and Li-Chen Fu. Nonlinear adaptive control of a flexible manipulator for automated deburring. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 2844–2849, Albuquerque, NM, April 1997. TJ211.I49x.

[79] Jeang-Lin Chang and Yon-Ping Chen. Force control of a single-link flexible arm using sliding-mode theory. *Journal of Vibration and Control*, 4(2):187–200, March 1998. TJ212.J68x.

[80] Stewart J. Moorehead and David [W. L.] Wang. An experimental study of contact transition control of a single flexible link using positive acceleration feedback. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, volume 4, pages 2838–2843, Albuquerque, NM, April 1997. TJ211.I49x.

[81] L. Meirovitch. *Analytical Methods in Vibrations.* MacMillan Co., New York, 1967.

[82] S. Timoshenko, D. H. Young, and W. Weaver, Jr. *Vibration Problems in Engineering.* John Wiley & Sons, Inc., Toronto, fourth edition, 1974. ISBN 0-471-87315-2, TA355.T55.

[83] Sandeep Jain and Farshad Khorrami. Vibration suppression of unknown flexible payloads using a wrist mounted force/torque sensor for remote manipulator systems. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, volume 3, pages 2119–2124, San Diego, CA, May 1994. TJ211.I49x.

[84] Richard Bellman. *Introduction to Matrix Analysis.* McGraw-Hill Book Co., Toronto, 1970.

[85] Daniel R. Madill. Tensor products and matrix calculus. Technical Report 98-10, Department of Electrical and Computer Engineering University of Waterloo, 200 University Ave. W., Waterloo, Ontario, Canada N2L 2G1, September 1998.

[86] William L. Brogan. *Modern Control Theory.* Prentice Hall, Englewood Cliffs, New Jersey, 1991.

[87] F. Belleza, L. Lanari, and G. Ulivi. Exact modelling of the flexible slewing link. In *Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, pages 734–739, 1990. TJ211.I49x.

[88] Milind Ghanekar. Scaling laws for linear controllers of dynamically equivalent single flexible link manipulators. Master's thesis, University of Waterloo, 1994.

[89] Erik Oberg, Franklin D. Jones, and Holbrook L. Horton. *Machinery's Handbook.* Industrial Press Inc., 200 Madison Ave., New York, N. Y. 10016, twenty-first edition, 1980.

[90] Daniel R. Madill. Parameters of the gravity balanced, dynamically decoupled five-bar robot. Technical Report 98-12, Department of Electrical and Computer Engineering University of Waterloo, 200 University Ave. W., Waterloo, Ontario, Canada N2L 2G1, September 1998.

[91] Mauro Rossi. Passive and adaptive controller design strategies for a class of manipulators with a flexible link. Master's thesis, University of Waterloo, 1994.

[92] Joseph Yan and S. E. Salcudean. Teleoperation controller design using H_∞-optimization. *IEEE Transactions on Control Systems Technology*, 4(3):244–258, May 1996. TJ212.I33x.

[93] Blake Hannaford and Robert Anderson. Experimental and simulation studies of hard contact in force reflecting teleoperation. In *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, volume 1, pages 584–589, 1988. TJ211.I49x.

[94] P. J. Hacksel and S. E. Salcudean. Estimation of environment forces and rigid-body velocities using observers. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 931–936, San Diego, USA, May 1994. TJ211.I49x.

[95] A. S. Yigit, A. G. Ulsoy, and R. A. Scott. Dynamics of a radially rotating beam with impact, part 1: Theoretical and computational model. *Journal of Vibration and Acoustics*, 112:65–70, January 1990.

[96] M[athukumalli] Vidyasagar. *Nonlinear Systems Analysis.* Prentice Hall, Englewood Cliffs, New Jersey 07632, second edition, 1993. QA402.V53.

[97] Kirsten M. Morris. AM955 lecture notes. University of Waterloo, May 1994.

[98] S. E. Salcudean and T. D. Vlaar. On the emulation of stiff walls and static friction with a magnetically levitated input/output device. In *ASME Journal of Dynamics, Measurement and Control*, April 1996. in press.