An Analytical and Experimental Study of the Simultaneous Control of Motion and Force of a Climbing Robot

by

Dalila Isabel Argáez

B.S. Mechanical Engineering
University of Illinois, Urbana-Champaign (1991)

Submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of

Masters of Science in Mechanical Engineering

at the

Massachusetts Institute of Technology

June, 1993

© 1993 Massachusetts Institute of Technology

Signature of Author

Department of Mechanical Engineering
May 28, 1993

Certified by

Steven D. Koonsky
Thesis Supervisor

Accepted by

Ain A. Sonin
Chairman, Departmental Graduate Committee
An Analytical and Experimental Study of the Simultaneous Control of Motion and Force of a Climbing Robot

Submitted to the Department of Mechanical Engineering on May 28, 1993, in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering

by

Dalila Isabel Argáez

Abstract

To harness the potential of robotic systems for tasks considered dangerous for humans will require algorithms designed to simultaneously control both force and motion. These tasks will further require the use of systems capable of ascending and descending uncharacterized structures and environments. To address these requirements, this thesis develops a Coordinated Multi-Point Force and Motion (CMFM) control algorithm to simultaneously control force and motion at various locations and directions on multi-limbed robotic systems. These locations and directions are referred to as control points. In addition, this thesis describes the design and fabrication of the Limbed Intelligent Basic Robotic Ascender (LIBRA), a three legged planar climbing robot. The CMFM control method is applied to this system.

The CMFM method uses the kinematic relationships defined by the Jacobian matrix which maps incremental joint motions to Cartesian motions. Such Jacobian based control traditionally has been applied to the end-effector of a manipulator. Here the extension of these methods to various control points on the robot is explored.

The control gains for the control method are designed using a singular value decomposition analysis of a linear multi-input multi-output system model of the LIBRA to meet the system design specifications. A Lyapunov analysis verifies the stability of the CMFM methodology and its convergence for both motion and force control. The control analysis is verified in computer simulations of the LIBRA.

Thesis Supervisor:  Steven Dubowsky
Professor of Mechanical Engineering
Acknowledgments

My first and foremost thanks goes to my advisor, Dr. Steven Dubowsky, for his firm, steady support over the past two years. He provided inspiration, instruction, and guidance as I traded Orange and Blue for the scarlet T of the Institute. A transition that marks an evolution from the world of theory to the realm of reality. He taught me that a theory, no matter how learned and detailed, must always be accompanied by a healthy measure of practice.

I would also like to thank Professor William Durfee and Dr. Erik Vaaler for their technical contributions to this work as well as for the personal support which they afforded.

And now, in order of appearance, I most humbly and heartily thank the crew of friends and colleagues that made me laugh and let me cry, answered my questions, and pondered life with me. Joe Deck, Charles Oppenheimer, Bert Hootsmans, Miguel Torres, Peng-Yun Gu, Akhil Madhani, Krisztina Holly, Husni Idris, Chris Pohlhammer, Uwe Mueller, Jay Baker and Dennis Goodrum, I thank you for all your knowledge so generously shared and lesson so gently taught. Chantal Moore, Tom Corrigan, and Craig Sunada, I thank you for all your encouragement, your willingness to help, and your wicked wit on those less than shiny days. Andrew Kuklinski, I thank you for six years of friendship, adventure, family, and love. Special thanks go to Kate Melvin.

To my best friend, my confidant and technical advisor, Jeffrey Michael Wendlandt, thank you for everything. Check. See you in California, one. Two years is too long.

To my parents, Dr. Juvenal Argaez and Socorro Argaez, thanks for your understanding, encouragement, and unending faith in my abilities and your unconditional love. To Ernesto, thank you for the long conversations. I follow in your footsteps always. To Adolfo, my own fellow Illini, thank you for providing distraction and adding humor through the years. I admire you each individually and collectively.

I gratefully acknowledge the support of the National Science Foundation which has supported my graduate studies through an NSF fellowship. In addition, I am thankful for the support provided by NASA (Langley Research Center, Automation Branch) under Grant NAG-1-801.
# Table of Contents

Abstract.................................................................................................................................ii
Acknowledgments......................................................................................................................iv
Table of Contents.....................................................................................................................vi
List of Figures...........................................................................................................................viii
List of Tables............................................................................................................................x
Nomenclature...........................................................................................................................xi

1 Introduction..............................................................................................................................1
   1.1 Motivation ............................................................................................................................1
   1.2 Purpose and Contributions of this Thesis ........................................................................3
   1.3 Outline of Thesis................................................................................................................4

2 Background and Past Work....................................................................................................5
   2.1 Fixed base Manipulator ......................................................................................................5
   2.2 Multi-Limbed Robotic Systems ........................................................................................6
   2.3 Summary............................................................................................................................8

3 Development--Coordinated Multi-Point Force and Motion Control.................................10
   3.1 Control Problem Statement ............................................................................................11
   3.2 Development of the Coordinated Multi-Point Force and Motion Control Algorithm .........11
   3.3 Mobility Analysis..............................................................................................................17
   3.4 Lyapunov Stability Criterion ............................................................................................20
   3.5 Summary............................................................................................................................27

4 Application--Design of the Coordinated Multi-Point Force and Motion Control Algorithm for a Climbing Robot..............................................................................28
   4.1 A Prototype Climbing Robot ............................................................................................28
   4.2 The Climbing Robot Jacobian ..........................................................................................30
   4.3 The Dynamic Equations of Motion ..................................................................................33
   4.4 Linearization of System ....................................................................................................39
   4.5 System Root Locus............................................................................................................41
   4.6 Control System Design ....................................................................................................44
   4.7 Simulation results..............................................................................................................58
   4.8 Summary............................................................................................................................71

5 Experimental System.............................................................................................................73
   5.1 System Specifications.........................................................................................................73
   5.2 Mechanical Design description of Climber .......................................................................75
      5.2.1 Leg Description ...........................................................................................................76
      5.2.2 Body Description ........................................................................................................78
      5.2.3 Foot Description .........................................................................................................80
      5.2.4 Wall Description .........................................................................................................81
5.2.5 Summary of Mechanical Design.................................81
5.3 Electrical Design Description and Total System overview........82
  5.3.1 Servo Amplifier Design Description..............................82
  5.3.2 Total System overview........................................83
5.4 Summary.....................................................................84

6 Experimental Implementation of the CMFM Control Algorithm........85
  6.1 Path Planning Algorithm..............................................85
  6.2 CMFM Software Description.........................................88
  6.3 Summary...................................................................96

7 Conclusions and Recommendations for Future Work..................97
  7.1 Control Conclusions...................................................97
  7.2 Climbing Robot..........................................................98
  7.3 Future Research.........................................................99
  7.4 Summary................................................................100

References ........................................................................101
Appendix A LIBRA Jacobian Matrix........................................106
Appendix B Development of the LIBRA Equations of Motion........116
Appendix C LIBRA Drawings................................................131
Appendix D LIBRA Torque Transformation................................148
<p>| Figure 3.1 | Concept of a Multi-Limbed Robotic System with Several Control Points | 12 |
| Figure 3.2 | Coordinated Multi-Point Force and Motion Control Concept | 13 |
| Figure 3.3 | Prototype Robot Mobility Analysis | 18 |
| Figure 3.4 | Prototype Climbing Robot Control Point Definition | 19 |
| Figure 4.1 | A Model of a Climbing Robot | 30 |
| Figure 4.2 | Kp Root Locus | 43 |
| Figure 4.3 | Kd Root Locus | 43 |
| Figure 4.4 | Sample System for Singular Value Analysis | 46 |
| Figure 4.5 | Closed loop system block diagram for the LIBRA using | 47 |
| Figure 4.6 | Desired Shape of the loop transfer function matrix | 55 |
| Figure 4.7 | Loop gain transfer function matrix | 56 |
| Figure 4.8 | Sensitivity transfer function matrix | 57 |
| Figure 4.9 | Complementary sensitivity transfer function matrix | 58 |
| Figure 4.10 | X body position response to step input | 61 |
| Figure 4.11 | Y body position response to a step input | 61 |
| Figure 4.12 | Body Orientation Response to a Step Input in Y | 62 |
| Figure 4.13 | Foot Force Step Response | 62 |
| Figure 4.14 | Control Effort for simultaneous step inputs in each control | 63 |
| Figure 4.15 | X body position response to a step input in X body and a ramp input in Y | 64 |
| Figure 4.16 | Y body position response to a ramp input | 65 |
| Figure 4.17 | Body orientation response to a step input in orientation and a ramp input in Y | 65 |
| Figure 4.18 | Foot Force response to a step input in force and a ramp input | 66 |
| Figure 4.19 | Control Effort for a ramp input in Y | 66 |
| Figure 4.20 | Magnitude Bode Plot based on the nonlinear simulation | 67 |
| Figure 4.21 | X body position response to a step input in X and a 1 hertz | 69 |
| Figure 4.22 | Y body position response to a 1 hertz sinusoidal input in Y | 69 |
| Figure 4.23 | Body orientation response to a step input in orientation and a 1 hertz sinusoidal input in Y | 70 |
| Figure 4.24 | Foot force response to a step input in force and a 1 hertz sinusoidal input in Y | 70 |
| Figure 4.25 | Control effort for a 1 hertz sinusoidal input in Y | 71 |
| Figure 5.1 | Assembly drawing of the LIBRA | 75 |
| Figure 5.2 | Elbow joint of leg | 77 |
| Figure 5.3 | Body drawing of the LIBRA | 79 |
| Figure 5.4 | Foot Design | 80 |
| Figure 5.5 | Servo amplifier circuit diagram | 83 |
| Figure 5.6 | Overview of Experimental System | 84 |
| Figure 6.1 | Motion Path for the climbing of a 3 legged climbing robot | 87 |
| Figure 6.2 | Software Overview Block Diagram | 89 |
| Figure 6.3 | PID Mode System Software Diagram | 91 |
| Figure 6.4 | CMFM Control Options | 93 |
| Figure 6.5 | Clock Interrupt Structure | 94 |
| Figure 6.6 | CMFM Control Algorithm | 95 |
| Figure A.1 | Prototype Climbing Robot Schematic for Development of System Jacobian | 107 |
| Figure B.1 | Prototype Climbing Robot Schematic for Development of System Equations of Motion | 118 |
| Figure D.1 | LIBRA Measured Angles | 148 |
| Figure D.2 | Prototype Climbing Robot Analysis Angles | 149 |</p>
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 4.1</td>
<td>Loop Gain Matrices</td>
<td>55</td>
</tr>
<tr>
<td>Table 5.1</td>
<td>Design specifications for LIBRA project</td>
<td>74</td>
</tr>
<tr>
<td>Table 5.2</td>
<td>Actuator specifications</td>
<td>76</td>
</tr>
<tr>
<td>Table 5.3</td>
<td>Specifications of the LIBRA Design</td>
<td>81</td>
</tr>
</tbody>
</table>
## Nomenclature

### Symbols

- **A, B, C, D** \(\text{linear system model matrices}\)
- **A}^H}** \(\text{complex conjugate transpose of a matrix, } A\)
- **C}^{\Theta, \dot{\Theta}}}** \(\text{Coriolis and Centripetal Acceleration vector}\)
- **C}_{\text{comp}}(s)** \(\text{complementary transfer function matrix}\)
- **d** \(\text{disturbance vector}\)
- **e** \(\text{system error}\)
- **F}_{\text{ext}}** \(\text{column vector of external forces and torques acting on control points}\)
- **F}_e** \(\text{column vector of external forces and torques reflected at the end-effector}\)
- **F}_{\text{imp}}** \(\text{column vector of forces and torques reflected at the control points}\)
- **G}^{\Theta}** \(\text{Gravity vector}\)
- **\hat{G}^{\Theta}** \(\text{estimate of Gravity vector}\)
- **G}(s)\)** \(\text{plant transfer function matrix}\)
- **H}(\Theta)\)** \(\text{Inertia Matrix}\)
- **I** \(\text{identity matrix of appropriate dimension}\)
- **\hat{i}** \(\text{unit vector in horizontal direction}\)
- **\hat{j}** \(\text{unit vector in vertical direction}\)
- **J}(\Theta)\)** \(\text{Jacobian matrix which maps joint velocities to control point velocities}\)
- **J}_{jk}\)** \(\text{element in the } j\text{th row and } k\text{th column of the Jacobian matrix}\)
- **\hat{k}** \(\text{unit vector orthogonal to the plane of climbing}\)
- **K}(s)\)** \(\text{controller transfer function matrix}\)
- **K}_p** \(\text{proportional gain matrix}\)
- **K}_{p\_xb}** \(\text{proportional gain in horizontal body direction}\)
- **K}_{p\_yb}** \(\text{proportional gain in vertical body direction}\)
- **K}_{p\_th}** \(\text{proportional gain of body orientation}\)
- **K}_{p\_foot}** \(\text{proportional gain for foot}\)
- **K}_d** \(\text{derivative gain matrix}\)
- **K}_{d\_xb}** \(\text{derivative gain in horizontal body direction}\)
- **K}_{d\_yb}** \(\text{derivative gain in vertical body direction}\)
- **K}_{d\_th}** \(\text{derivative gain of body orientation}\)
- **K}_{d\_foot}** \(\text{derivative gain for foot}\)
- **K}_s** \(\text{matrix of environmental compliance}\)
- **KE** \(\text{kinetic energy of one link}\)
L \quad \frac{1}{2} \quad \text{length of a link} \\
L \quad \text{Lagrangian} \\
N \quad \text{distance between motors on body link} \\
\mathbf{n} \quad \text{noise vector} \\
r \quad \text{distance from motors to center of mass of body link} \\
S_{\text{sens}}(s) \quad \text{sensitivity transfer function matrix} \\
T^* \quad \text{total system kinetic energy} \\
T(s) \quad \text{loop gain transfer function matrix} \\
\mathbf{u} \quad \text{column vector of robot control points subset of } \mathbf{r} \\
\mathbf{u}_{\text{ext}} \quad \text{column vector of points at which there exist contact between the environment and the robot} \\
\mathbf{\ddot{u}} \quad \text{column vector of system error between actual control point positions and desired control point positions} \\
\mathbf{\ddot{u}}_{\text{ext}} \quad \text{column vector of system error between contact points and actual control point positions} \\
U_i \quad \text{potential energy of } i\text{th link} \\
U^* \quad \text{total system potential energy} \\
U \quad \text{unitary matrix of left singular vectors} \\
u_i \quad \text{ith component of a vector} \\
V \quad \text{unitary matrix of right singular vectors} \\
v \quad \text{linear velocity column vector} \\
\|v\| \quad \text{magnitude of velocity vector} \\
V.I. \quad \text{variational indicator} \\
v_{ix} \quad \text{horizontal component of velocity for } i\text{th link} \\
v_{iy} \quad \text{vertical component of velocity for } i\text{th link} \\
x \quad \text{state vector} \\
x_{\text{body}} \quad \text{horizontal body position measured to the center of mass of the body link of the robot} \\
x_{\text{foot}} \quad \text{horizontal foot position of foot } 1 \text{ of the robot} \\
x_{\text{foot}_3} \quad \text{horizontal foot position of foot } 3 \text{ of the robot} \\
Y \quad \text{vertical distance between foot on left most wall and foot on right most wall while in a pull up motion} \\
y_{\text{body}} \quad \text{vertical body position of the body link of the robot} \\
y_{\text{foot}_3} \quad \text{vertical foot position of foot } 3 \text{ of the robot} \\

\textbf{Greek Symbols} \\
\partial \quad \text{partial derivative} \\
\delta \quad \text{variational operator} \\
\forall \quad \text{for all}
\( \gamma \) small positive number much less than one
\( \lambda_i \) \( i \)th eigenvalue of a matrix
\( \Omega_r \) range of input vector frequencies
\( \Omega_n \) range of noise vector frequencies
\( \Omega_d \) range of disturbance vector frequencies
\( \omega \) angular velocity
\( \omega \) frequency
\( \phi \) absolute angle from horizontal to the link 1
\( \Sigma \) diagonal matrix of singular values
\( \sigma_i \) \( i \)th singular value
\( \sigma_{\text{max}} \) maximum singular value
\( \sigma_{\text{min}} \) minimum singular value
\( \tau \) column vector of joint torques
\( \tau_0 \) equilibrium condition of joint torques
\( \theta_{\text{body}} \) absolute body orientation of the body link of the robot
\( \Theta \) vector of generalize variables--joint angles
\( \dot{\Theta} \) vector of joint angle velocities
\( \dot{\Theta}_0 \) equilibrium condition of joint angles
\( \dot{\Theta}_0 \) equilibrium condition of joint velocities
\( \theta_1 \) absolute angle from horizontal to the link 2
\( \theta_2 \) relative angle between body orientation and link 2
\( \theta_3 \) relative angle between body and link 4
\( \theta_4 \) relative angle between link 4 and link 5
\( \theta_5 \) relative angle between body and link 6
\( \theta_6 \) relative angle between link 6 and link 7

**Acronyms and Abbreviations**

CMFM Coordinated Multi-Point Force and Motion control
LIBRA Limbed Intelligent Basic Robotic Ascender
PMAC Programmable Multi-Axis Controller
1 Introduction

This chapter summarizes the motivation of this research in the control of force and position of limbed robotic systems. Further, the chapter highlights the control issues involved and outlines the organization of the thesis.

1.1 Motivation

Robotic systems have the potential to perform tasks labeled unfeasible for human beings to attempt. For example, within a nuclear facility, tasks include climbing along lengthy pipes to repair damage on reactor core plates and moving radioactive waste containers. Tasks performed during a launch pad chemical spill include escalating to remote locations, excavating contaminated materials, and stabilizing and directing fire hose spray. In the restoration of architecture and artifacts, tasks include mounting scaffolding, and surveying and mapping sites for lichen damage. Assembly and maintenance of space-borne structures will involve scaling and assembling structural truss work; while mining and undersea exploration involve similar tasks which require the system to move in restricted and cramped environments.

To harness this potential, robotic systems will be challenged to exert large controlled forces and maintain desired trajectories in the course of achieving certain task specifications. In addition, the remote locations in which such tasks are performed may require the use of manipulation systems able to scale
over obstacles, climb into crevasses, or ascend scaffolding or conventional ladders. While current mobile manipulation systems consist generally of a standard manipulator bolted to a wheeled or tracked base, the implication is that to effectively accomplish some tasks, robotic systems with multiple limbs will be required.

Many of the pertinent design and control issues for multi-limbed robotic systems, however, are not well understood. While some work has been done for the motion planning of such systems, little to date has been explored in the area of control. Conventional position control has largely been the focus of past research of mobile manipulators. However, this control approach is not designed to address the fundamental issues involved in controlling force exertion while simultaneously following a specified trajectory. While some research has shown the validity of applying control algorithms capable of motion and force tracking, it has largely been limited to fixed-based manipulators or to mobile manipulators on wheeled bases. The extension of these techniques to multi-limbed robotic systems remains to be investigated.

This thesis makes the extension to multi-limbed robotic systems. The focus is on the development of an algorithm for the simultaneous control of forces and motion at multiple locations and coordinate directions of multi-limbed robotic systems.
1.2 Purpose and Contributions of this Thesis

The purpose of this thesis is to present the development of an algorithm to simultaneously impose trajectory following and control the exertion of forces at multiple locations and coordinate directions of multi-limbed systems. These multiple locations and coordinate directions are referred to as control points. The technique is referred to as Coordinated Multi-Point Force and Motion (CMFM) control. The control scheme is a unified approach to control motion and force for multi-limbed systems. The control algorithm is verified in simulations. In addition, the algorithm is verified empirically on a climbing robotic system. The details of the climbing robot design are discussed and its capabilities are outlined. The experimental results presented serve to highlight the capabilities of the control scheme.

This thesis illustrates the ability to control forces and motions at multiple points and directions of multi-limbed systems. The control approach builds on the principles guiding hybrid and impedance controllers and derives carefully the conditions under which it is possible to make the extension to multiple control points. A stability proof is presented to verify the validity of the control approach. An appropriate Lyapunov function is defined to establish the stability of the result.

The control algorithm design is outlined for a prototype climbing robot. A climbing robot is designed and fabricated to serve as a test bed for this control algorithm as well as for the development and testing of future algorithms for the locomotion of multi-limbed robotic systems.
1.3 Outline of Thesis

This thesis details the development of the Coordinated Multi-Point Force and Motion control algorithm and its application to a climbing robot. Chapter One presents the motivation behind the research of multi-limbed machines and details the applications for such robotic systems. Chapter Two surveys past work in the control of robotic manipulators, giving examples of multi-limbed climbing machines which exist. Chapter Three summarizes the development of the Coordinated Multi-Point Force and Motion control algorithm. Chapter Three also presents a mobility analysis and a Lyapunov proof which validates the control algorithm stability and convergence for both position and force control applications. The control theory development is showcased for the case of a planar climbing robot in Chapter Four. Chapter Five introduces the design of the climbing robot. Chapter Six describes the experimental system setup and demonstrates experimentally the capabilities of the control algorithm. Chapter Seven concludes the thesis with a synopsis of relevant results and recommendations for further research.
2 Background and Past Work

This chapter presents an overview of past work in the control of robotic systems such as fixed base manipulators, manipulators on wheeled bases, and multi-limbed robots. The need for control algorithms that simultaneously control force and motion for multi-limbed systems is highlighted.

2.1 Fixed base Manipulator

Many control approaches have been applied to achieve the stable performance of manipulators with accurate position and force control of the manipulator links and end-effector. The success of standard position-derivative controllers, which do not include explicit system parameters, in industrial applications can be largely explained by the high gear reduction ratios found in many industrial robots [2]. More complex Jacobian-based controllers which include kinematic parameters and computed torque controllers which include kinematic and dynamic parameters are increasingly filling the need in high speed applications where highly geared actuators are no longer required. Hogan relays the theory and the implementation of an impedance control scheme which uses the Jacobian kinematic relationship to arrive at the control effort [19-22]. Various other control methods have been implemented including resolved rate, stiffness, hybrid, resolved acceleration, and operational space control in order to achieve accurate positioning of the manipulator end-effector or to exert forces with the end-effector [16,17,30-31].
Typically, these control algorithms have been applied to robotic manipulators operating in highly structured and well defined environments, mounted on fixed, rigid platforms. While motion planning has been the focus of some research in the area of mobile robotic systems, little to date has been explored in the area of control of these systems.

Recently research in mobile manipulators on wheeled bases has sparked interest. Dubowsky and Hootsman presented a control approach to implement robust position control schemes which compensate for the uncontrolled dynamics of a base vehicle [23-24]. These techniques account for the dynamics and kinematics of the manipulator and its base, using both the system governing motion equations as well as the kinematic relationship defined by the system Jacobian.

2.2 Multi-Limbed Robotic Systems

For multi-limbed systems, however, the area of control has not been aggressively researched, although advancements have resulted in some path planning algorithms. Recently, Madhani and Dubowsky formulated a planning approach which can aid in the design of multi-limbed robotic systems as well as plan optimal paths for these systems to follow [35]. Further research contributions have been made in the control of autonomous or telerobotic land vehicles or submersibles to maneuver in unknown environments [5, 13]. The focus has been on 'he planning activities of such
systems in unknown environments. Several studies plan the movement of legged systems on rough terrain [18].

In the arena of control, Woernle describes a scheme of feedback linearization for the hybrid position and force control of a multi-body system [49]. While the proposed algorithm is instructive and accurate when a detailed system model is available, it is not robust to errors in modeling. In one of the earliest discussions of compliance control, Whitney describes a generalized damper approach to control a compliance relation for force control applications [47]. Mason sets the theoretical framework for hybrid control and its relationship to task constraints [36]. Building on this, Lozano-Perez and Mason describe a formal method for planning generalized-damper commands to carry out assembly tasks [32]. Raibert and Craig have conceptualized an approach to control manipulator motions to simultaneously satisfy position and force constraints [40]. Application of this approach to the end-effector of a two link fixed-based manipulator has proven successful, while extension to multi-limbed systems appears feasible. Recently, Khatib defined how generalized joint torques are reflected at the end-effector—a critical understanding of which is crucial for active force control of redundant manipulators [29]. Eppinger and Seering establish a fundamental limit on the efficacy of any force control scheme for a system with non-collocated sensors and actuators [10]. In a related problem, Schneider and Cannon arrive at an object impedance controller for cooperative manipulation [42]. Through theoretical development and experimentation, a control scheme which utilizes the unique ability of impedance control to control the relationship between
displacement and position is demonstrated to successfully coordinate motion between two two-link robots.

Several multi-limbed robotic systems have been fabricated. Akizono et. al. have constructed an underwater inspection robot with a simple path algorithm which uses the legs of the robot as instrumentation devices [1]. Gradetsky and Rachkov discuss a climbing robot using vacuum grippers for actuation [14-15]. The Portsmouth Polytechnic Robug II, similarly uses vacuum grippers to climb [8]. A search of the available literature, however, has not yielded any information on the control of these multi-limbed systems. A unified control approach to control force and motion for multi-limbed systems would expand on the applications of robotic systems.

2.3 Summary

Extension of robotic systems to their potential application dictates the need to control both force and position simultaneously as discussed in Chapter One. Building on the research activity in force and position control of fixed-base or wheeled-base manipulators, it is possible to arrive at a control algorithm for multi-limbed robotic systems.

While recent techniques, such as impedance and hybrid controllers, have combined both force and position control, much of the work has focused exclusively on either position or force. Even hybrid and impedance control research has been largely limited to techniques focused on the end-effector
with little application to multiple control points. Cooperative manipulation techniques involving the coordinated control of an object by several robotic systems are examples of applications closely related to multi-limbed systems. However, applications of coordinated control have been limited to fixed base manipulators.

Anchored in a solid foundation of control algorithms for fixed-based robotic systems, however, the extension of these techniques for the simultaneous coordinated control of force and motion of multi-limbed systems is explored in this thesis. The Coordinated Multi-Point Force and Motion control algorithm is a unified approach to allow multi-limbed robotic systems to perform tasks where motions must be controlled at some location and direction on the robot while force must be controlled at another.
In order to accomplish some tasks in remote locations, robotic systems may be required to exert large forces while maintaining themselves on a desired path. Some tasks will require the use of multi-limbed systems capable of stepping through complex terrains such as scaffolding. Some path planning algorithms have been developed to define desired path and optimal force for such systems [38,48]. It remains, however, to develop a method to allow for the simultaneous control of both force and motion for these multi-limbed robots.

In this chapter, a methodology for achieving the simultaneous control of force and position is presented. First, the control problem is stated. Second the Coordinated Multi-Point Force and Motion (CMFM) control methodology is presented. Next, the expansion of motion and force control beyond the conventional control point at the end-effector of the manipulator is explored. A control point is a location and direction on a robotic system or manipulator for which a desired motion or force has been defined. A mobility analysis for the case of a climbing robot is presented. This analysis governs the capability of extension of the CMFM methodology to arbitrarily control several points and directions on a manipulator. Finally, this chapter details the stability proofs for the proposed coordinated motion and force control. Section 3.4 is a proof of the motion control and force control capabilities of the CMFM control algorithm.
3.1 Control Problem Statement

The control problem which this thesis addresses can be summarized as follows: Given multiple control points for which desired motions or forces have been defined, control the multi-limbed robotic system such that it actively follows the defined trajectories and exerts the desired forces despite the presence of disturbances.

3.2 Development of the Coordinated Multi-Point Force and Motion Control Algorithm

Figure 3.1 depicts a multi-limbed robotic system suggested in Reference [38]. Several control points are labeled. Suppose that certain motions have been specified for the center body of the robot. Specifically, motions have been specified for the x body direction, y body direction. The inertial coordinate frame is shown in Figure 3.1. In addition, suppose it is desired that given forces be exerted by the robot's endpoints to hold its footing. The control problem is to simultaneously exert these forces while maintaining a desired trajectory of the center body.

In References [20] and [21] a methodology for the control of motions and forces for fixed based manipulators. Reference [23] expands this methodology to manipulators on wheeled bases. In this method, the desired motions are modeled as forces through a generalized spring and damper approach while desired forces are transcribed into desired relationships between force and
position. This is referred to as impedance control. The Coordinated Multi-Point Force and Motion control algorithm extends this concept in a unified approach to control robotic system motions and force exertion at multiple control points. The idea is best expressed in a diagram. Figure 3.2 illustrates the control concept.

![Concept of a Multi-Limbed Robotic System with Several Control Points](image)

**Figure 3.1 Concept of a Multi-Limbed Robotic System with Several Control Points**

The control points are labeled. With each control point there is associated a spring and damper. For motion control applications, the generalized spring-damper pair exists between the actual position of the control point and the desired position. The spring and damper constants characterize the relationship which governs the motion of the control position to the desired position. This desired control point position follows the defined trajectory. In this manner motion is controlled. For force control applications, the desired control point resides within the environment. The generalized
spring-damper pair now characterize the relationship which governs the force exertion. As the spring constant is increased, the force exerted increases. This can also be accomplished by moving the control point further into the environment.

![Diagram of coordinated multi-point force and motion control concept]

* = Control Point Position
X = Desired Control Point Position

Figure 3.2 Coordinated Multi-Point Force and Motion Control Concept
Amalgamating these individual forces into one vector of desired forces allows for a unified approach to simultaneously control forces and motions. Let $\mathbf{F}_{imp}$ be a generalized damper and a generalized spring vector:

$$
\mathbf{F}_{imp} = \mathbf{K}_P (\mathbf{u}_{des} - \mathbf{u}) + \mathbf{K}_D (\dot{\mathbf{u}}_{des} - \dot{\mathbf{u}})
$$

(3.2.1)

where $\mathbf{K}_P$ is the diagonal matrix of desired stiffnesses

$\mathbf{K}_D$ is the diagonal matrix of desired damper constants

$\mathbf{u}_{des}$ is the vector of desired control point locations

$\mathbf{u}$ is the vector of actual control point locations

$\dot{\mathbf{u}}_{des}$ is the vector of desired control point velocities

$\dot{\mathbf{u}}$ is the vector of actual control point velocities.

The result is a vector of desired forces, $\mathbf{F}_{imp}$. The distinction between motion and force control is not apparent in the control approach. The distinction is made only in the location of the desired control point position, $\mathbf{u}_{des}$. For force control, the desired control point will lie within some environment. In Figure 3.2, three such control points are shown. The multi-limbed system to which this algorithm is applied will react as if there is some force pulling it into the environment based on the control law. The environment will thus will provide a reaction force opposing this motion. The result is a force at the interface. For motion control applications, the control law will make the robot act as if a force were pulling the control point on its desired trajectory.

The control law follows as:
\[ \tau = J^T(\Theta) \mathbf{F}_{\text{imp}} \]  \hspace{1cm} (3.2.2)

where  \( J(\Theta) \)  is the Jacobian matrix  
\( \mathbf{F}_{\text{imp}} \)  is the vector of desired forces  
\( \tau \)  is the control effort exerted by each joint.

The Jacobian matrix relates small joint motions to the resulting control point motions. The Jacobian for a fixed-based manipulator relates the small joint motions for a given manipulator configuration to its resulting end-effector motions. This convention results from the fact that in most robotic applications it is the end-effector which is of importance since it interacts with the environment and is responsible for manipulation. There is, however, nothing particular about the end-effector, and the Jacobian matrix, more generally, simply refers to the function mapping small motions in the joint space reference frame to motions of some points on the system in the inertial frame.

The Jacobian matrix relation is:

\[ \delta \mathbf{u} = J(\Theta) \delta \Theta \]  \hspace{1cm} (3.2.3)

where  \( J(\Theta) \)  is the Jacobian matrix  
\( \delta \Theta \)  is the vector of joint perturbations  
\( \delta \mathbf{u} \)  is the resulting vector of control point perturbations.
The control law in Equation 3.2.2 follows from the principle of virtual work. The idea is to determine the effort which must be applied by the manipulator joint actuators in order to exert a given force and torque at the control points on an object or an environment. A geometrically admissible configuration of a mechanical system is in equilibrium if and only if the variational indicator, V. I., vanishes for arbitrarily admissible variations of configuration. The variational indicator is defined as the incremental work done under an admissible variation [9]. For the desired force vector derived from the generalized spring and damper formulation, the variational indicator is:

\[ \text{V.I.} = \tau^T \delta \Theta - F_{imp}^T \delta u \]  \hfill (3.2.4)

where \( \delta \) is the variational operator.

Since \( \delta u \) is related to \( \delta \Theta \) through the Jacobian matrix, \( J(\Theta) \), Equation 3.2.6 can be rewritten as:

\[ \text{V.I.} = [\tau^T - F_{imp}^T J(\Theta)] \delta \Theta \]  \hfill (3.2.5)

The requirement is that V.I. must vanish for all geometrically admissible joint displacements, \( \delta \Theta \). This dictates that:

\[ \tau = J^T(\Theta) F_{imp} \]  \hfill (3.2.2)

which is the control law specified earlier.
Incorporating an estimated gravity term, $\hat{G}(\Theta)$, in the control effort in order to reduce steady state errors, the required joint torques can be defined, therefore, as:

$$\tau = J^T(\Theta)F_{\text{imp}} + \hat{G}(\Theta) \quad (3.2.6)$$

This control effort will result in the exertion of the desired force vector, $F_{\text{imp}}$. Therefore, the control points under motion control will act as if there were a force pulling them to the desired locations. The control points for which force is being controlled will exert, therefore, the desired forces.

### 3.3 Mobility Analysis

The extension of the impedance control ideology, however, cannot be made to control motion and force at arbitrarily many points on any closed chain system such as a climbing robot. To ensure that the extension is made properly, an analysis of the mobility of the system must be made. The mobility of a mechanism is defined as the number of independent parameters required to specify the location of each link in the mechanism. The following is an analysis for a climbing robot. The prototype robot is depicted in Figure 3.3.

The Grubler Equation allows the definition of the mobility of a chain of links [34]. The Grubler Equation is:
\[ M = 3(n - 1) - 2f_1 - f_2 \]  \hspace{1cm} (3.3.1)

where

- \( M \) is the mobility, or the number of degrees of freedom
- \( n \) is the total number of links including ground
- \( f_1 \) is the number of one-degree-of-freedom joints
- \( f_2 \) is the number of two-degree-of-freedom joints.

![Diagram of robot mobility analysis](image)

**Figure 3.3 Prototype Robot Mobility Analysis**

The climbing robot has eight links including ground, seven one degree of freedom joints and one joint with two degrees of freedom. Therefore the mobility of the robot is:
The system has six degrees of freedom.

With six degrees of freedom, the implication is that six control points can be controlled. Therefore, looking at the coordinate frame and labels in Figure 3.4, the center body position in x and y as well as the center body orientation can be controlled while simultaneously controlling the force exerted in the x direction by the foot of leg 2 against the wall. In addition, the other two degrees of freedom allow for positioning of the foot of the third leg.
Finally, recognize that force and position cannot be controlled independently in any given control point direction [27]. This would be akin to trying to independently compress a spring by a specified distance and simultaneously controlling the spring force. Position, however, can be controlled in one direction on a robotic system while simultaneously control force in another direction.

This mobility analysis, exemplified here for a prototype climbing robot, is a prerequisite for the application of the CMFM control algorithm. The number of independent control points cannot exceed the number of active degrees of freedom of the system. In addition, it is not physically realizable to control force and position in any given direction. Either force may be controlled or position exclusively.

### 3.4 Lyapunov Stability Criterion

Before attempting to implement the CMFM controller, it is important to perform not only a mobility analysis but also to establish the stability of the controller. As with any stability analysis, the first step is to analyze the system dynamics in conjunction with the dynamics of the controller. Generally, the dynamic equations of motion for a robotic system can be written as follows:

$$H(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + G(\Theta) = \tau + J^T(\Theta)F_{ext} \quad (3.4.1)$$

where \(H(\Theta)\) is the inertia matrix
\( C(\dot{\Theta}) \) is the Coriolis and centripetal force vector
\( G(\Theta) \) is the vector of gravitational potential energy
\( \Theta \) is the vector of joint angles
\( \tau \) is the vector of joint torques
\( J(\Theta) \) is the Jacobian matrix
\( F_{\text{ext}} \) is the vector of external forces acting on the manipulator at the control points.

The external force, \( F_{\text{ext}} \), is defined as follows:

\[
F_{\text{ext}} = K_E \ddot{u}_{\text{ext}} \tag{3.4.2}
\]

where \( K_E \) is the matrix of environmental stiffnesses at the control points of the system
\( \ddot{u}_{\text{ext}} \) is the environmental error vector.

If a control point is under motion control in the \( i \)-th direction, then the compliance in this direction is zero. If a control point in the \( j \)-th direction is under force control, then the \( j \)-th component of this matrix is the actual physical compliance of the interface.

The environmental error vector, \( \ddot{u}_{\text{ext}} \), is given by:

\[
\ddot{u}_{\text{ext}} = u - u_{\text{ext}} \tag{3.4.3}
\]

where \( u_{\text{ext}} \) is the vector of points at which the control points make contact with the environment.
As developed in Section 3.2, the control effort is defined by a Jacobian transpose algorithm which reflects the desired control point forces to the robot joints. Defining this desired force as:

\[ F_{\text{imp}} = K_p (u_{\text{des}} - u) + K_D (\dot{u}_{\text{des}} - \dot{u}). \]  

(3.4.4)

the resulting command torques are:

\[ \tau = J^T(\Theta)F_{\text{imp}} + \hat{G}(\Theta). \]

(3.4.5)

where \( \hat{G}(\Theta) \) is the estimated gravity vector.

Define the system error, \( \ddot{u} \), as:

\[ \ddot{u} = u_{\text{des}} - u. \]

(3.4.6)

The following analysis follows closely the analysis given in [2] for the stability of an impedance control algorithm. Formulating the Lyapunov function:

\[ V = \frac{1}{2} \left[ \ddot{u}^T K_p \ddot{u} + \Theta^T H(\Theta) \dot{\Theta} + \ddot{u}_{\text{ext}}^T K_f \ddot{u}_{\text{ext}} \right] \]

(3.4.6)

gives an energy-based function which satisfies the following criterion. First, the function is positive definite. That is, for any manipulator configuration the function remains positive and is equal to zero only when \( \ddot{u} \) is zero. Second, notice that the function is continuous and that its partial derivatives are continuous in the whole of the state space. Next, examine the first derivative of the function. If this derivative is negative semi-definite, then
the equilibrium point $\bar{u} = 0$ is stable. If, in addition, this derivative is negative definite, then the equilibrium point is asymptotically stable.

The derivative takes the form:

$$
\dot{V} = \bar{u}^T K_p \dot{u} + \dot{\Theta}^T \dot{H}(\Theta) \dot{\Theta} + \frac{1}{2} \dot{\Theta}^T \ddot{H}(\Theta) \Theta + \bar{u}_{ext}^T K_E \dot{u}_{ext}. \quad (3.4.7)
$$

Substituting the governing system equations into Equation 3.4.7 gives:

$$
\dot{V} = \bar{u}^T K_p \dot{u} + \dot{\Theta}^T \left[ \dot{G}(\Theta) - J^T(\Theta) \{ K_p \bar{u} + K_d \dot{u} \} - J^T(\Theta) * K_E \bar{u}_{ext} - G(\Theta) - C(\Theta, \dot{\Theta}) \dot{\Theta} \right] \\
+ \frac{1}{2} \dot{\Theta}^T \ddot{H}(\Theta) \Theta + \bar{u}_{ext}^T K_E \dot{u}_{ext}.
$$

(3.4.8)

For simplicity and without loss of generality, assume that

$$
\dot{G}(\Theta) = G(\Theta). \quad (3.4.9)
$$

In addition, assume that

$$
\bar{u}_{des} = 0. \quad (3.4.10)
$$

The derivative then reduces to the following:

$$
\dot{V} = \bar{u}^T K_p \dot{u} + \dot{\Theta}^T \left[ -J^T(\Theta) \{ K_p \bar{u} + K_d \dot{u} \} - J^T(\Theta) * K_E \bar{u}_{ext} - C(\Theta, \dot{\Theta}) \dot{\Theta} \right] \\
+ \frac{1}{2} \dot{\Theta}^T \ddot{H}(\Theta) \Theta + \bar{u}_{ext}^T K_E \dot{u}_{ext}.
$$

(3.4.11)
Recall that the physics of the manipulator indicate that

\[ \dot{H}(\Theta) - 2C(\Theta, \dot{\Theta})\dot{\Theta} = 0. \quad (3.4.12) \]

Also, the left hand side of Equation 3.4.12 is anti-symmetric. Thus the terms, \( \dot{\Theta}^T C(\Theta, \dot{\Theta})\dot{\Theta} \) and \( \frac{1}{2} \dot{\Theta}^T \dot{H}(\Theta)\dot{\Theta} \) cancel.

Also notice that

\[ \dot{\Theta}^T J^T(\Theta) = \ddot{u}^T. \quad (3.4.13) \]

Further, recall

\[ \ddot{u}_{ext}^T K_E = F_{ext}^T. \quad (3.4.14) \]

The Conservation of Energy dictates that:

\[ F_{ext}^T \dot{u}_{ext} = 0. \quad (3.4.15) \]

The derivative in Equation 3.4.11 becomes:

\[ \dot{V} = -\ddot{u}^T K_D \ddot{u} \leq 0. \quad (3.4.16) \]

From this we see that the equilibrium point \( \ddot{u} = 0 \) is stable. The implication is that if the manipulator configuration begins near the desired configuration, then it will not move far away from that desired configuration. This is a part
of the type of stability desired. It is desired not only that the system not “blow up” for arbitrary configurations approximating the desired one, but that the manipulator converge on the desired configuration for any arbitrary initial configuration.

This is the concept of asymptotic stability [43]. To show that the stability is asymptotic, it is necessary to prove that the derivative is negative definite. That is, the derivative is equal to zero if and only if

\[ \ddot{u} = 0 \quad (3.4.17) \]

which implies that

\[ u = u_{\text{des}}. \quad (3.4.18) \]

To prove this, it is necessary to examine the conditions for which the acceleration term vanishes. For non-singular configurations,

\[ \ddot{\Theta} = 0 \quad (3.4.19) \]

if and only if

\[ \dot{\Theta} = 0 \quad (3.4.20) \]

which implies that
\[ \ddot{\Theta} = -H(\Theta)^T J^T(\Theta) \{ K_p \ddot{u} + K_h \ddot{u}_{ext} \}. \quad (3.4.21) \]

Solving this equation for \( u \) yields:

\[ u = [K_p + K_h]^{-1} \{ K_p u_{des} + K_h u_{ext} \}. \quad (3.4.22) \]

For position control applications of the \( i \)-th control point, the \( i \)-th component of the environmental compliance matrix, is zero:

\[ k_{e_i} = 0. \quad (3.4.23) \]

Therefore, the \( i \)-th component of the position vector converges by Equation 3.4.22 to its desired position

\[ u_i = u_{des_i}. \quad (3.4.24) \]

Thus, the system will converge to the desired configuration and the stability is asymptotic.

For force control applications in the \( i \)-th control point, if the proportional gain matrix stiffness in the \( j \)-th direction is chosen so that it is much smaller compared to the actual compliance of the environmental interface:

\[ k_{e_j} \gg k_{p_j} \quad (3.4.25) \]

then
First, this ensures that proper footing will be maintained. Second, it verifies that the desired force at the wall will be achieved as long as this force is proportional to the compliance of the wall.

3.5 Summary

This chapter has developed the Coordinated Multi-Point Force and Motion control algorithm. The algorithm extends the concepts of impedance control to multiple control points. The chapter also indicates the need to perform a mobility analysis for a robotic system on which the CMFM control algorithm will be implemented. It cautions that the number of control points cannot exceed the number of degrees of freedom of the robotic system. Finally, the chapter presents a Lyapunov proof which verifies the asymptotic stability of the CMFM control method. The CMFM control algorithm is, thus, a unified approach to the simultaneous control of force and motion problem present in many applications of multi-limbed systems.
4 Application—Design of the Coordinated Multi-Point Force and Motion Control Algorithm for a Climbing Robot

This Chapter discusses the control system design for a climbing robot. First, a prototype climbing robot is presented. Second, the kinematic equations of the robotic system are derived, and a system Jacobian matrix is formulated. Third, the derivation of the dynamic equations which govern the system motions is outlined. Next, the control system design for the climbing robot is presented. Finally, simulation results are discussed.

4.1 A Prototype Climbing Robot

A prototype climbing robot is described below.

The prototype system consists of three legs and a center body. Each leg consists of two limbs and a foot. Figure 4.1 illustrates the system model. Labeled in the illustration are the joint angles used in the subsequent system analysis. The convention used is that all angles and torques are positive counter clockwise. The angles are defined as follows:

\( \phi \) is the absolute angle measured from the horizontal to link 1
\( \theta_1 \) is the absolute angle measured from the horizontal to link 2
$\theta_2$ is the relative angle between link 2 and the body link
$\theta_3$ is the relative angle between link 3 and link 4
$\theta_4$ is the relative angle between link 4 and link 5
$\theta_5$ is the relative angle between link 3 and link 6
$\theta_6$ is the relative angle between link 6 and link 7.

The following are the assumptions of the model. The first assumption is that the climbing motion occurs in one plane. Second, the compliance at the wall-foot interface is assumed to be 100 lb./in. This is in accordance with the measured compliance of the proposed experimental system wall as discussed in Section 5.2. Third, an actuator saturation in each joint of 20 Nm is assumed. Fourth, in all the analysis it is assumed that the links of the climbing robot are rigid bodies.

Figure 4.1 indicates the inertial coordinate frame for the prototype system. Link 1 is modeled as a hinged joint fixed to the wall. The axis origin is fixed to the hinged joint of link 1. The robot is assumed to be climbing along two parallel walls as shown. The planted feet on the walls in Figure 4.1 are foot 1 and foot 2. The vertical distance between these two feet along the y-direction is assumed to be a system constraint. The wall separation is constant; however, the assumption is that there exists a compliance at the foot wall interface which allows for deflection in this x-axis direction. The degree of freedom for the system is six as explained in Section 3.2. Therefore, the CMFM control algorithm can be applied to control forces and motions at six control points. The control points are labeled in Figure 3.3 and repeated in Figure 4.1 for clarity. These points include the x body direction, the y body
direction, the body orientation, the second foot force against the wall, the x
direction of the foot 3, and the y direction of foot 3.

![Diagram of a model of a climbing robot](image)

**Figure 4.1 A Model of a Climbing Robot**
Notice that the model consists of 3 legs and a body and is planar

### 4.2 The Climbing Robot Jacobian

The formulation of the kinematic equations which govern the prototype
climbing robot motions enables the application of the CMFM control
algorithm for this system. The following is an explanation of the derivation
of the Jacobian matrix.
For the prototype climbing robot, the vector of control points is:

\[ u = \begin{bmatrix}
  x_{\text{body}} \\
  y_{\text{body}} \\
  \theta_{\text{body}} \\
  x_{\text{foot}_2} \\
  x_{\text{foot}_3} \\
  y_{\text{foot}_3}
\end{bmatrix} \quad (4.2.1) \]

In Figure 4.1,

- \text{L} is the one-half the link length
- \text{N} is the distance between centers of actuators on the robot body
- \text{r} is the distance from the actuator to the center of mass of the body.

The \( x_{\text{body}} \) position can be formulated as a function of the robot joint angles:

\[ x_{\text{body}} = 2*\text{L}*\cos(\phi) + 2*\text{L}*\cos(\theta_1) + r*\cos\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right) \]

\[ (4.2.2) \]

where the angle \( \phi \) is defined as:

\[ \phi = \text{asin}(D) \quad (4.2.3) \]

and where \( D \) is given by:

\[ D = \frac{Y - 2*\text{L}\left[ \sin(\theta_1) + \sin(\theta_1 + \theta_2 + \theta_3) + \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right] - \text{N}\sin(\theta_1 + \theta_2)}{2*\text{L}}. \]

\[ (4.2.4) \]
Similarly, the equation relating joint angles to the \( y_{\text{body}} \) position is:

\[
y_{\text{body}} = 2*L*\sin(\phi) + 2*L*\sin(\theta_1) + r*\sin\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right).
\]  
(4.2.5)

The orientation of the body is simply:

\[
\theta_{\text{body}} = \theta_1 + \theta_2.
\]  
(4.2.6)

Finally, the foot position is given as:

\[
x_{\text{foot,2}} = 2*L*\cos(\phi) + 2*L*\cos(\theta_1) + N*\cos(\theta_1 + \theta_2) + 2*L*\cos(\theta_1 + \theta_2 + \theta_3) + \frac{2}{3}N*\cos(\theta_1 + \theta_2 + \theta_3 + \theta_4).
\]  
(4.2.7)

To formulate the Jacobian matrix, simply take the derivative of \( \dot{u} \):

\[
\dot{u} = \begin{bmatrix}
\dot{x}_{\text{body}} \\
\dot{y}_{\text{body}} \\
\dot{\theta}_{\text{body}} \\
\dot{x}_{\text{foot,2}} \\
\dot{x}_{\text{foot,3}} \\
\dot{y}_{\text{foot,3}}
\end{bmatrix} = J(\Theta)\dot{\Theta}.
\]  
(4.2.8)

where \( \Theta \) is the vector of joint angles shown in Figure 4.1.

The following is a sample element of the Jacobian:
\[ J_{11} = \frac{D}{\cos(\arcsin(D))} \left[ 2*L* \begin{pmatrix} \cos(\theta_1) \\ +\cos(\theta_1 + \theta_2 + \theta_3) \\ +\cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \end{pmatrix} + N*\cos(\theta_1 + \theta_2) \right] \\
-2*L*\sin(\theta_1) - r*\sin\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right) \]

(4.2.9)

The subscript \( J_{jk} \) indicates the element in the \( j \)-th row and the \( k \)-th column of the Jacobian matrix. The entire Jacobian can be found in Appendix A.

### 4.3 The Dynamic Equations of Motion

Two methods for the derivation of the equations of motion of a manipulator are traditionally employed, the Lagrangian method and the Newton-Euler method. The following is a brief synopsis of this derivation for the six degree of freedom planar climbing robot using the Lagrangian method.

The first step in the Lagrangian method is to formulate a group of generalized variables. The number of variables is usually equal to the number of degrees of freedom of the system.

Choose the following generalized variables:
\[ \Theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{bmatrix} \quad (4.3.1) \]

These are the angles of the joints of the robot as indicated in Figure 4.1. The generalized efforts associated with these variables are the joint torques.

\[ \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \\ \tau_5 \\ \tau_6 \end{bmatrix} \quad (4.3.2) \]

Lagrange's method requires the formulation of the kinetic and potential energy of the climbing robot. In the following, the kinetic energy of each link of the robot is formulated. The potential energy is developed later in this section.

Formulation of the kinetic energy of the system necessitates the development of the inertial velocities of the center of mass of each link and the inertial angular velocities. Since for each link the total kinetic energy is given by:

\[ KE = \frac{1}{2} m* \|v\|^2 + \frac{1}{2} I* \|\omega\|^2. \quad (4.3.3) \]

Note that the following notation will be used for the unit directional vectors:
\[
\hat{i} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \hat{j} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \hat{k} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.
\] (4.3.4)

The definition of the following variables eases the derivation:

\[
c_{1} = \cos(\theta_{1}), \quad s_{1} = \sin(\theta_{1})
\]
\[
c_{12} = \cos(\theta_{1} + \theta_{2}), \quad s_{12} = \sin(\theta_{1} + \theta_{2})
\]
\[
c_{p1} = \cos(\theta_{1} + \theta_{2} - \frac{\pi}{6}), \quad s_{p1} = \sin(\theta_{1} + \theta_{2} - \frac{\pi}{6})
\]
\[
c_{123} = \cos(\theta_{1} + \theta_{2} + \theta_{3}), \quad s_{123} = \sin(\theta_{1} + \theta_{2} + \theta_{3})
\]
\[
c_{1234} = \cos(\theta_{1} + \theta_{2} + \theta_{3} + \theta_{4}), \quad s_{1234} = \sin(\theta_{1} + \theta_{2} + \theta_{3} + \theta_{4}).
\] (4.3.5)

The velocity of the i-th link is given as:

\[
v_{i} = (v_{ix}) \hat{i} + (v_{iy}) \hat{j}.
\] (4.3.8)

The magnitude of this velocity is written as:

\[
\|v_{i}\| = \sqrt{(v_{ix})^2 + (v_{iy})^2}.
\] (4.3.9)

Notice that the square root is eliminated in taking the kinetic energy making it unnecessary to expand the components of velocity.

The velocity components in terms of the generalized variables for link 1 are:
\[
\begin{bmatrix}
2* L*[c1 + c123 + c1234] + N*c12 \\
2* L*[c123 + c1234] + N*c12 \\
2* L*[c123 + c1234] \\
2* L*c1234 \\
0 \\
0
\end{bmatrix}^T * \Omega
\]

\[
\begin{bmatrix}
2* L*[c1 + c123 + c1234] + N*c12 \\
2* L*[c123 + c1234] + N*c12 \\
2* L*[c123 + c1234] \\
2* L*c1234 \\
0 \\
0
\end{bmatrix}^T * \Omega
\]

4.3.10

\[
\begin{align*}
\dot{v}_{1x} &= \frac{D}{2*\sqrt{1 - D^2}} \\
\dot{v}_{1y} &= -\cos(a \sin(D)) \frac{1}{2*\sqrt{1 - D^2}} 
\end{align*}
\]

(4.3.11)

where

\[
\dot{\Omega} = \begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2 \\
\dot{\theta}_3 \\
\dot{\theta}_4 \\
\dot{\theta}_5 \\
\dot{\theta}_6
\end{bmatrix}.
\]

(4.3.12)

The angular velocity of link 1 in terms of the generalized variables is:

\[
\begin{bmatrix}
2* L*[c1 + c123 + c1234] + N*c12 \\
2* L*[c123 + c1234] + N*c12 \\
2* L*[c123 + c1234] \\
2* L*c1234 \\
0 \\
0
\end{bmatrix}^T * \Omega.
\]
The remaining velocity terms are found in Appendix B. Together with the system mass properties and Equation 4.3.3 the velocities characterize the system total kinetic energy.

The system potential energy incorporates the gravitational potential energy of the robot only. The potential energy due to the compliance at the wall-foot interface is modeled as an external force acting on the system.

For link 1, the potential energy is:

\[ U_1 = m \cdot g \cdot L \cdot D. \]  \hspace{1cm} (4.3.14)

The remaining potential energy terms are found in Appendix B.

The total system kinetic energy is given by \( T^* \). The total system potential energy is \( U^* \). Formulating the Lagrangian

\[ L = T^* - U^* \]  \hspace{1cm} (4.3.15)

and deriving according to the Lagrange equations:

\[ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_i} \right) - \frac{\partial L}{\partial \theta_i} = \tau_i \]  \hspace{1cm} (4.3.16)
will give at a complex set of interdependent equations.

The velocity and potential energy terms for all the links are given in Appendix B. The dynamics of the system are used in this thesis for purposes of simulation. Matlab M-files are used to construct the exact form of the equations of motion. The control method does not rely on the dynamic equations of motion. Instead, the control is a function simply of the Jacobian matrix formulated in Section 4.2.

The equations of motion can be summarized in the following:

\[ H(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + G(\Theta) = \tau + J^T(\Theta)F_{ext} \quad \text{(4.3.17)} \]

where
- \( H(\Theta) \) is the inertia matrix
- \( C(\Theta, \dot{\Theta}) \) is the Coriolis and centripetal force vector
- \( G(\Theta) \) is the vector of gravitational potential energy
- \( \Theta \) is the vector of joint angles
- \( \tau \) is the vector of joint torques
- \( J(\Theta) \) is the Jacobian matrix
- \( F_{ext} \) is the vector of external forces acting on the manipulator at the control points.

The vector of external forces is described in Section 3.2. The potential energy term from the modeled compliance of the wall is incorporated in this external force term rather than in the dynamics of the climbing robot. With these equations, a nonlinear simulation was developed to demonstrate the
control capabilities of the CMFM control algorithm. A linear system model is developed next from which a set of system gains for the generalized spring and damper constants are chosen to meet certain criteria.

4.4 Linearization of System

The nonlinear equations are linearized in this section which allows for the application of several tools for control system design. The linear system description follows.

Let,

\[
\Theta_0 = \begin{bmatrix}
\theta_{1_0} \\
\theta_{2_0} \\
\theta_{3_0} \\
\theta_{4_0} \\
\theta_{5_0} \\
\theta_{6_0}
\end{bmatrix}
\]

be any arbitrary configuration about which the system is to be linearized. At a static equilibrium position, the velocity terms vanish:

\[
\dot{\Theta}_0 = \begin{bmatrix}
\dot{\theta}_{1_0} \\
\dot{\theta}_{2_0} \\
\dot{\theta}_{3_0} \\
\dot{\theta}_{4_0} \\
\dot{\theta}_{5_0} \\
\dot{\theta}_{6_0}
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}.
\]
Thus, the equilibrium torques must offset the gravitational potential energy terms and the external efforts exerted:

$$\tau_0 = G(\Theta_0) + J^T(\Theta_0) \ast F_{ext}.$$  \hspace{1cm} (4.4.3)

To linearize the system first perturb each generalized variable in the equation of motion about the equilibrium position such that:

$$\Theta = \Theta_0 + \delta \Theta.$$  \hspace{1cm} (4.4.4)

For example,

$$\sin(\theta_1) = \sin(\theta_{10} + \delta \theta_1) = \sin(\theta_{10}) \cos(\delta \theta_1) + \cos(\theta_{10}) \sin(\delta \theta_1)$$
$$= \sin(\theta_{10}) + \cos(\theta_{10}) \ast \delta \theta_1.$$  \hspace{1cm} (4.4.5)

Similarly, each torque is perturbed about its equilibrium.

$$\tau = \tau_0 + \delta \tau$$  \hspace{1cm} (4.4.6)

In this fashion, the following matrices are formulated. Let

$$x = \begin{bmatrix} \Theta \\ \dot{\Theta} \end{bmatrix}$$  \hspace{1cm} (4.4.7)

be the state variable vector for the system. Equation 4.3.33 reduces to the following set of linear equations.
\[ \dot{x} = Ax + Bz \quad (4.4.8) \]
\[ u = Cx + Dz \quad (4.4.9) \]

where

\[
A = \begin{bmatrix}
0 & 1 \\
-H^{-1}(\Theta_0) * (G(\Theta_0) + J^T(\Theta_0) * F_{ext}) & 0
\end{bmatrix}
\]
\[ (4.4.10) \]

\[
B = \begin{bmatrix}
0 \\
H^{-1}(\Theta_0)
\end{bmatrix}
\]
\[ (4.4.11) \]

\[
C = \begin{bmatrix}
J(\Theta) & 0
\end{bmatrix}
\]
\[ (4.4.12) \]

\[
D = [0]
\]
\[ (4.4.13) \]

The formulation of a linear system model facilitates the control system design. The gains for the controller are chosen for the linear model and simulations of these gains for the nonlinear model are performed.

### 4.5 System Root Locus

The linear system model provides a basis for selecting system gains. The following is a root locus for the system. Since the gains which are introduced by the control effort simulate passive generalized springs and generalized
dampers at the control points, intuitively the closed loop system ought to be stable. Figure 4.2 gives a root locus for of the linear system closed loop model with the CMFM controller described in Section 3.2. Shown are the root locus for varying the body proportional gain from 0 to 2000 N/m. The figure indicates that the roots of the system have negative real parts for all positive values of proportional gain. Thus, the closed loop system is stable.

Figure 4.3 is a root locus plot for body derivative gains varying from 0 to 200 N*sec/m. With constant position gains for the body and the foot. The body proportional gain used is 1000 N/m. The foot gain is 500 N/m. For the derivative gain of the foot the gain is 120 N*sec/m.

The conclusion therefore, is that for any positive definite gain matrices, the closed loop system will be stable.
4: Stability of the Non Collocated Motion and Force Control

Figure 4.2 Kp Root Locus; notice stability for all positive Kp is guaranteed

Figure 4.3 Kd Root Locus; notice increased damping pulls locus to real axis
4.6 Control System Design

The system gain matrices are selected to meet the following specifications.

1. Steady State position error of less than 1 mm.
2. Steady State force error of less than 50%.
3. Bandwidth of 1 Hz.
4. Critically damped system.
5. Steady State orientation errors of less than 1 degree.
6. Torque requirements of less than 10 Nm.

A position steady state position error of less than 1 mm and less than 1 degree in orientation is reasonable since sensing accuracy problems such as backlash limit the accuracy of the feedback received to ± 0.7 mm and 0.72 degrees. The steady state force error allowance of less than 50% while high, is indicative of the flexibility in the control of force in the climbing robot. To maintain footing the horizontal force exerted must be greater than a minimum value. Any force exceeding that value will suffice to maintain steady footing. The experimental system described in Chapter Five has a sampling rate of 100 Hz. Thus, a bandwidth of 1 Hz is chosen. The critical damping criterion indicates that the system will not have large overshoots in position, orientation, and force. Finally, the torque limit of 20 Nm is determined by the torque saturation limit on the motors of the experimental robot.
Before proceeding with the selection of system gains to meet these criterion, a brief overview of singular value decomposition is given. The singular value decomposition of a transfer function matrix for multi-input multi-output systems, such as a climbing robot, is a valuable design tool for defining system gains. The following discussion is based on notes from reference [3].

A singular value of a matrix \( A \) is defined as:

\[
\sigma_i(A) = \sqrt{\lambda_i(A^H A)} > 0 \quad (4.6.1)
\]

where \( A^H \) is defined as the complex conjugate transpose of the matrix, \( A \).

The matrix, \( A \), can be recast as product of three matrices forming the singular value decomposition of the matrix \( A \). If \( \Sigma \) is a diagonal matrix of the singular values of \( A \) and if \( U \) is the unitary matrix of left singular vectors of \( A \) or right eigenvectors of \( AA^H \), and if \( V \) is the unitary matrix of right singular vectors of \( A \) or the right eigenvectors of \( A^H A \), then \( A = U \Sigma V^H \) is a singular value decomposition of \( A \).

Another important concept is that of the size of a matrix. This is given by the spectral norm of the matrix. For a matrix \( A \) the spectral norm is given as:

\[
\| A \|_2 = \max_{\| x \|_2 \neq 0} \frac{\| Ax \|_2}{\| x \|_2} = \max_{\| x \|_1} \| Ax \|_2 \quad (4.6.2)
\]

The relationship between the matrix spectral norm and the singular values of the matrices is specified as follows:
\[ \sigma_{\text{max}}(A) = \max_{\|x\|_2 = 1} \|Ax\|_2 = \|A\|_2 \]  \hspace{2cm} (4.6.3) \\
\[ \sigma_{\text{min}}(A) = \min_{\|x\|_2 = 1} \|Ax\|_2 = \frac{1}{\|A^{-1}\|_2} \]  \hspace{2cm} (4.6.4)

The singular values of a transfer function matrix determine the maximum and minimum amplifications of the system output. The singular values of a transfer function matrix are, therefore, indicative of the system bandwidth. For example, consider the system depicted in Figure 4.4.

![Figure 4.4 Sample System for Singular Value Analysis](image)

The output response of a system is given by:

\[ y = G(j\omega)u. \]  \hspace{2cm} (4.6.5)

Suppose that the input is limited by:

\[ \|u\|_2 = 1. \]  \hspace{2cm} (4.6.6)

This is the case when each element of the input vector is a complex sinusoid:

\[ u_i(t) = |u_i|e^{j\phi_i}e^{j\omega t}. \]  \hspace{2cm} (4.6.7)
From the relationship of singular values to the spectral norm of a matrix explained above, the maximum and minimum singular values of $G(j\omega)$ define maximum and minimum amplifications of unit sinusoidal input at frequency $\omega$.

$$\sigma_{\text{min}}(G(j\omega)) = \min_{\|u\|_2=1} \|G(j\omega)u\|_2 = \|y_{\text{min}}(\omega)\|$$  \hspace{1cm} (4.6.8)

$$\sigma_{\text{max}}(G(j\omega)) = \max_{\|u\|_2=1} \|G(j\omega)u\|_2 = \|y_{\text{max}}(\omega)\|$$  \hspace{1cm} (4.6.9)

Figure 4.5 illustrates the closed loop linearized climbing robot system with the Coordinate Multi-Point Force and Motion controller.

![Block Diagram of Closed Loop System](image)

**Figure 4.5** Closed loop system block diagram for the LIBRA using simultaneous motion and force control

The plant transfer function matrix, $G(s)$, is defined as:

$$G(s) = C(sI - A)^{-1}B$$  \hspace{1cm} (4.6.10)

where $s$ is the complex frequency.
The controller transfer function matrix, $K(s)$, is:

$$K(s) = K_p + K_D \cdot s$$  \hspace{1cm} (4.6.11)

where
- $K_p$ is the diagonal generalized spring gain matrix for the CMFM control algorithm,
- $K_D$ is the diagonal generalized damper gain matrix for the CMFM control algorithm.

The loop transfer function matrix, $T(s)$, is:

$$T(s) = G(s) \cdot K(s)$$  \hspace{1cm} (4.6.12)

$$T(s) = C \cdot (s \cdot I - A)^{-1} \cdot B \cdot [K_p + K_D \cdot s].$$  \hspace{1cm} (4.6.13)

The system performance equation, is:

$$\epsilon(s) = S_{sens}(s) \cdot [u_{des}(s) - d(s)] + C_{comp}(s) \cdot n(s)$$  \hspace{1cm} (4.6.14)

where
- $\epsilon(s)$ is the error vector,
- $d(s)$ is the vector of disturbances,
- $n(s)$ is the vector of noise inputs,
- $S_{sens}(s)$ is the sensitivity transfer function matrix,
- $C_{comp}(s)$ is the complementary sensitivity transfer function matrix.

The sensitivity transfer function matrix is defined as:
\[ S_{\text{sens}}(s) = [I + T(s)]^{-1} \] (4.6.15)

and the complementary sensitivity is defined as:

\[ C_{\text{comp}}(s) = [I + T(s)]^{-1} \ast T(s). \] (4.6.16)

For accurate command following and disturbance rejection, therefore, the sensitivity transfer function matrix must be analyzed. The relationship between error and the commanded input is given by:

\[ e = S_{\text{sens}}(j\omega) \ast u_{\text{des}}. \] (4.6.17)

From the previous analysis, the magnitude of the error is determined by the following relation:

\[ \|e\|_2 \leq \sigma_{\text{max}}[S_{\text{sens}}(j\omega)] \ast \|u_{\text{des}}\|_2. \] (4.6.18)

Therefore, to accurately following commanded inputs, that is, to drive the error to zero requires that

\[ \sigma_{\text{max}}[S_{\text{sens}}(j\omega)] \ll 1 \forall \omega \in \Omega, \] (4.6.19)

where \( \Omega \) is the set of all input command frequencies. For the climbing robot application, an input command frequency range of 1 Hz is desired. The general requirement for the maximum singular value of the sensitivity transfer function matrix can be formulated as an equivalent requirement for the loop gain transfer function matrix according to Equation 4.6.15.
Specifically, for the set of all commanded frequencies, the specification for good command following requires that:

$$\sigma_{\text{min}}[T(j\omega)] \gg 1 \forall \omega \in \Omega_r.$$  \hspace{1cm} (4.6.20)

This follows from certain facts about singular values. First, notice that the maximum singular value of the inverse of a matrix is the reciprocal of the minimum singular value of the matrix. Simply:

$$\sigma_{\text{max}}(A^{-1}) = \frac{1}{\sigma_{\text{min}}(A)}.$$  \hspace{1cm} (4.6.21)

Also, the minimum singular value of the sum of a matrix and the identity matrix, $I$, is bounded by the following relation:

$$\sigma_{\text{min}}(A) - 1 \leq \sigma_{\text{min}}(I + A) \leq \sigma_{\text{min}}(A) + 1.$$  \hspace{1cm} (4.6.22)

Thus by Equation 4.6.15 since:

$$S_{\text{sens}}(s) = [I + T(s)]^{-1}$$  \hspace{1cm} (4.6.15)

the requirement for good command following defined in Equation 4.6.19 can be formulated into a loop gain transfer function matrix requirement using Equation 4.6.21:

$$\sigma_{\text{max}}(S(j\omega)) = \frac{1}{\sigma_{\text{min}}(I + T(j\omega))} \ll 1 \quad \forall \omega \in \Omega_r.$$  \hspace{1cm} (4.6.23)
From this, it follows that:

\[ \sigma_{\text{min}}(I + T(j\omega)) >> 1 \quad \forall \omega \in \Omega_r. \]  \hspace{1cm} (4.6.24)

Equation 4.6.22 then implies that:

\[ \sigma_{\text{min}}(I + T(j\omega)) < \sigma_{\text{min}}(T(j\omega)) + 1. \]  \hspace{1cm} (4.6.25)

In conjunction with Equation 4.6.24 this yields the requirement stipulated in Equation 4.6.20, repeated here for completeness:

\[ \sigma_{\text{min}}(T(j\omega)) >> 1 \quad \forall \omega \in \Omega_r. \]  \hspace{1cm} (4.6.20)

Similarly, the relation between disturbance and error is given by the sensitivity function as follows:

\[ e = S_{\text{sens}}(j\omega)*d. \]  \hspace{1cm} (4.6.26)

This implies that the magnitude of the error due to a disturbance is bounded by:

\[ \|e\|_2 \leq \sigma_{\text{max}} [S_{\text{sens}}(j\omega)]*\|d\|_2. \]  \hspace{1cm} (4.6.27)

Therefore, to attenuate disturbances requires that:

\[ \sigma_{\text{max}} [S_{\text{sens}}(j\omega)] << 1 \quad \forall \omega \in \Omega_d. \]  \hspace{1cm} (4.6.28)
where \( \Omega_d \) is the set of all disturbance input frequencies. For the climbing robot, disturbances below 20 rad/sec are to be attenuated. Again this requirement is reformulated as a loop gain transfer function matrix requirement.

\[
\sigma_{\min} \left[ T(j\omega) \right] >> 1 \forall \omega \in \Omega_d. \quad (4.6.29)
\]

The relationship between sensor noise and error is given as a function of the complementary sensitivity transfer function:

\[
e = C_{\text{comp}}(j\omega) \ast n. \quad (4.6.30)
\]

The error is bounded above by:

\[
\|e\|_2 \leq \sigma_{\max} \left[ C_{\text{comp}}(j\omega) \right] \ast \|n\|_2. \quad (4.6.31)
\]

Therefore, it is desirable to place the following limit for the maximum singular value of the complementary sensitivity transfer function matrix:

\[
\sigma_{\max} \left[ C_{\text{comp}}(j\omega) \right] << 1 \forall \omega \in \Omega_n \quad (4.6.32)
\]

where \( \Omega_n \) is the set of all noise frequencies. Inherently, however, there exists a performance conflict due to the constraint equation:

\[
S_{\text{sens}}(s) + C_{\text{comp}}(s) = I \quad (4.6.33)
\]
The constraint equations dictates that it is not possible to do good command following and disturbance rejection with noisy sensors that make low frequency errors such as drift and bias. This requires a separations between \( \Omega_r \cup \Omega_d \) and \( \Omega_n \).

As in the case of disturbance rejection and command following, this noise attenuation specification can be transformed into a specification for the loop gain transfer function matrix. Let

\[
0 < \gamma \ll 1 \tag{4.6.34}
\]

and suppose that

\[
\sigma_{\text{max}} \left[ C_{\text{comp}}(j\omega) \right] \ll \gamma \quad \forall \omega \in \Omega_n. \tag{4.6.35}
\]

Formulating the inverse of the complementary sensitivity transfer function matrix gives insight into the desired specification in terms of the loop gain transfer function matrix. The inverse is:

\[
C_{\text{comp}}^{-1}(j\omega) = T^{-1}(j\omega) \ast [I + T(j\omega)] = T^{-1}(j\omega) + I \tag{4.6.36}
\]

Since the minimum singular value of the inverse is the maximum singular value of the matrix, a relation is obtained.

\[
\sigma_{\text{min}} \left( C_{\text{comp}}^{-1}(j\omega) \right) = \sigma_{\text{min}} \left( T^{-1}(j\omega) + I \right) \leq \sigma_{\text{min}} \left( T^{-1}(j\omega) \right) + 1 = \sigma_{\text{max}} \left( T(j\omega) \right) + 1. \tag{4.6.37}
\]
Thus,

$$\sigma_{\text{min}}(C_{\text{comp}}^{-1}(j\omega)) - 1 \leq \sigma_{\text{max}}(T(j\omega)). \quad (4.6.38)$$

Moreover, by assumption 4.6.35:

$$\sigma_{\text{min}}(C_{\text{comp}}^{-1}(j\omega)) - 1 = \sigma_{\text{max}}(C_{\text{comp}}(j\omega)) - 1 < \gamma - 1 < \gamma. \quad (4.6.39)$$

Finally, the desired relationship for noise attenuation in terms of the loop gain transfer function matrix is given by:

$$\sigma_{\text{max}}(T(j\omega)) < \gamma \ll 1. \quad (4.6.40)$$

The desired shape has a high minimum singular value at frequencies up to 8 Hertz and a low maximum singular value at frequencies above the range of frequencies at which inputs will be commanded. Any low frequency noise will cause the control system to fail. Figure 4.6 is a schematic of the desired shape for the loop transfer function matrix.
Desired Loop Transfer Function Matrix \( T(j\omega) \)
Singular Value Decomposition

![Graph showing frequency response with Maximum Singular Value and Minimum Singular Value](image)

Figure 4.6 Desired Shape of the loop transfer function matrix

Figure 4.7 is the loop gain transfer function converged on using an iterative search to arrive at the desired shape. The gains used are:

<table>
<thead>
<tr>
<th>Kp Gains</th>
<th>Kd Gains</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K_p_{xb} = 1000 )</td>
<td>( K_d_{xb} = 160 )</td>
</tr>
<tr>
<td>( K_p_{yb} = 1000 )</td>
<td>( K_d_{yb} = 160 )</td>
</tr>
<tr>
<td>( K_p_{th} = 22 )</td>
<td>( K_d_{th} = 4 )</td>
</tr>
<tr>
<td>( K_p_{foot}=500 )</td>
<td>( K_d_{foot}=120 )</td>
</tr>
</tbody>
</table>

Table 4.1 Loop Gain Matrices; \( K_p \) and \( K_d \) are diagonal matrices
Figure 4.7 Loop gain transfer function matrix singular values plot

Figure 4.8 illustrates directly the command following and disturbance rejection capabilities of the system. Notice that the design specification of a bandwidth of 1 Hertz is satisfied while simultaneously allowing disturbance rejection up to 8 Hertz.
Figure 4.8 Sensitivity transfer function matrix singular values plot

Figure 4.9 shows that the system is able to attenuate only noises greater than 300 Hertz due to the constraint equation which does not allow noise attenuation for the range of frequencies of commanded inputs. Any noise in the commanded frequency range results in an inability to accurately follow a commanded signal.
4.6 Simulation results

While capable of executing position control to a certain degree of accuracy, the inherent problem with PID control for force requirements is that small errors in position can result in loss of contact with the environment. In a situation where the robot is climbing, loss of contact results in the robot losing footing and can cause the robot to fall. Combining force control with position control creates a more robust scheme. The control of force rather than position at the
wall has proven to be an appropriate strategy. The following are the results of simulations. Depicted are the step, ramp, and sinusoidal responses of the system under CMFM control algorithm. The control gains used are given in Table 4.1.

The response of the system to a step input is illustrated in the following four graphs. The control effort exerted is also shown. Figure 4.10 shows the x body position step response. The rise time for the x body position is 0.375 seconds. The steady state error converges to zero in 1.062 seconds and the response is critically damped. Figure 4.11 shows the y body position step response. The rise time is 0.375 seconds again. The settling time is 1.125 seconds and the steady state error is zero.

Figure 4.12 shows the body orientation step response. Notice that the response is not critically damped as in the case for the translational body positions. This soft control of the orientation response minimizes the magnitude of the control effort while still allowing for stable climbing motions. The settling time is less than 1 second.

Figure 4.13 shows the foot force response for a step input in force. The force is commanded to go from 14 N pulling out of the wall to -1 N pushing into the wall. The overshoot is 38.4% and the settling time is 0.25 seconds. The fast rise time and settling characteristics of this response are crucial for maintaining stable footing. Increased damping can decrease the overshoot of the response, but does so at the expense of greater actuator effort. Figure 4.14 shows the actuator efforts expended in the step response. Notice that the
torque limit of 20 Nm is not exceeded. The operating range of the control actuators is -5 Nm to -3 Nm which adheres with the rated dynamic performance of several DC motors.
Figure 4.10 X body position response to step input

Figure 4.11 Y body position response to a step input
Figure 4.12 Body Orientation Response to a Step Input in Y

Figure 4.13 Foot Force Step Response
A ramp input of 0.051 meters per second, approximately two inches per second, was given for the desired y body position of the robot. Concurrently, a step input was given to the remaining control points of the body. Figure 4.15 shows the x body response. The rise time is 0.0625 seconds and the error settles to zero after 0.5 secs. Notice that the response is critically damped. Figure 4.16 shows the ramp response of the y body position notice that the body is able to follow the command exactly. The error begins at a maximum level of 0.2 mm and decreases to zero is 0.0625 seconds.
Figure 4.17 shows the body orientation response. Again to minimize actuator effort, the control for the orientation for the body is soft. However, the error is bounded by 0.0015 radians, approximately 0.09 degrees.

Figure 4.18 depicts the force response. Again the overshoot of 38.4% can be minimized by increase damping in the controller at the expense of increased actuator effort. The settling time is 0.2565 seconds and the steady state error converges to zero. Figure 4.19 shows the control effort expended in the move. Notice that the control efforts steer clear of the 20 Nm specification for the torque limit.

![Graph showing ramp response and desired body position](image-url)
Ramp Response
Desired Y Body Position == 0.0508 m/sec
Body Position (y) vs. Time

Figure 4.16 Y body position response to a ramp input

Ramp Response
Desired Y Body Position == 0.0508 m/sec
Body Orientation vs. Time

Figure 4.17 Body orientation response to a step input in orientation and a ramp input in Y
Figure 4.18 Foot Force response to a step input in force and a ramp input in Y

Figure 4.19 Control Effort for a ramp input in Y
To test the bandwidth of the system, the y body position was given an input of several sinusoids at varying frequencies. Figure 4.20 is a magnitude Bode plot. Notice that the system bandwidth is 6.4 Hz. The system can be expected to faithfully track sinusoidal unit inputs of up to 6.4 Hz.

![Magnitude Bode Plot](image)

**Figure 4.20** Magnitude Bode Plot based on the nonlinear simulation sinusoidal response data

Depicted below are the results for a sinusoidal input of 1 Hz in the y body desired position. Figure 4.21 shows that the x body position has a rise time of 0.375 seconds and is critically damped. The settling time is 1.063 seconds and the steady state error is bounded by 0.00025 m. Figure 4.22 shows the y body position response. Notice that the robot is able to track the position with a steady state error of less than 0.1 mm. Figure 4.23 shows the body orientation response. Again the control of the orientation is flexible allowing for
minimal control effort. The error however is bounded by 0.09 degrees. Recall that the specification for steady state error in the body orientation is 1 degree.

Figure 4.24 shows an overshoot for the desired force which could be eliminated with increased damping. The overshoot is 38.4%. The settling time is 0.25 secs with a steady state error of less that .002 N. The steady state error in force is only 0.02 % which is well within the specification. Figure 4.25 shows the torque actuation effort expended in the 1 Hertz move. Again notice that the torque limit is not violated.
Figure 4.21 X body position response to a step input in X and a 1 hertz sinusoidal input in Y

Figure 4.22 Y body position response to a 1 hertz sinusoidal input in Y
Figure 4.23 Body orientation response to a step input in orientation and a 1 hertz sinusoidal input in Y.

Figure 4.24 Foot force response to a step input in force and a 1 hertz sinusoidal input in Y.
4.7 Summary

This Chapter has presented the design and simulation results for a force and motion controller based on the CMFM control algorithm for a prototype climbing robot. The Jacobian matrix for the system is presented in Appendix A. The equations of motion of the system and the simulation software is incorporated into Appendix B. The system specifications are given and the design is formulated to meet these specifications. The results of the simulation shows that the CMFM controller designed is able to faithfully track a sinusoidal input of 1 Hz. For motion control applications, the system
has a steady state error of less than 1 mm and less than 1 degree in orientation. Also, the force steady state error is well within the allotted 50% error specification. The step responses of the system are critically damped, except in the force control application. This is a trade off made in order to keep the torque control efforts below the saturation level of 20 Nm.
A description of the experimental system is detailed in this chapter. After a discussion of system specifications, a brief outline of the mechanical design of the system is presented. Finally, the electronics hardware is discussed along with a total system overview.

5.1 System Specifications

The objective of the Limbed Intelligent Basic Robotic Ascender (LIBRA) design program was to develop a system to serve as a testbed for the qualitative and quantitative study of climbing path planning and control strategies.

The approach taken emphasized simplicity in design and for analysis. The mobility analysis of Section 3.2 indicates that to control a climbing robot's body position and orientation and to exert a specified force against an environment while having the capability to climb upwards, requires 6 degrees of freedom. Thus, the designed robot consists of six actuators and three legs. While more legs may provide greater kinematic stability, a three legged mechanism is capable of climbing and provides insight into issues of stability and challenge path planning and control schemes.
Simplicity dictated that DC motors be used for the LIBRA. These motors needed to have a high torque-to-weight ratio since the robot is required to support these actuators. High gear reduction ratios were chosen to meet the high-torque-low-weight constraints.

In addition, it was desired that the system have a span of at least 2 feet. The total robot weight should not exceed 15 pounds. Further, the environment which the robot climbs consists of two parallel walls. The walls are made of "L"-shaped steps made of angle iron.

Table 5.1 summarizes the mechanical design specifications.

<table>
<thead>
<tr>
<th>Item</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Legs</td>
<td>3</td>
</tr>
<tr>
<td>Actuators</td>
<td>DC Motors</td>
</tr>
<tr>
<td></td>
<td>High torque to weight ratio</td>
</tr>
<tr>
<td>Total System Weight</td>
<td>less than 15 lb.</td>
</tr>
<tr>
<td>Span of limbs</td>
<td>2 feet</td>
</tr>
<tr>
<td>Wall</td>
<td>2 parallel ladders of stepped angle iron</td>
</tr>
<tr>
<td></td>
<td>6 inch step size</td>
</tr>
<tr>
<td></td>
<td>12 inch wall separation</td>
</tr>
</tbody>
</table>

Table 5.1 Design specifications for LIBRA project

The following is a mechanical design description of the LIBRA.
5.2 Mechanical Design description of Climber

The mechanical system design is presented in following categories: leg and actuator description; body and foot descriptions of the LIBRA; and a brief environment description. The governing principles of the mechanical design were flexibility and simplicity. Therefore, much of the mechanical design is homogeneous and modular in scope. In addition, the design was shaped by the requirements of a planar system. This is accentuated in the design of the system joints as discussed in the next section. Figure 5.1 is a partial assembled view of the LIBRA. Depicted are two limbs, body, and a foot of the robot. One assembled leg is shown. The other two legs follow similarly.

![Diagram of LIBRA](image)

Figure 5.1 Assembly drawing of the LIBRA
5.2.1 Leg Description

The leg design consists of two similar limb members. Each limb consists of a leg tube, two yokes, and an actuator.

Figure 5.2 is a schematic of the elbow joint. Notice that each joint part is modular. The yoke is made of three parts which are fastened together using 1/8" dowel pins and 6-32 socket head screws. A modular design approach was chosen, thus allowing for the use of various actuators of different dimensions without many changes to the yoke design. This flexibility, however, allows the potential for misalignment in assembly. Notice that dowels are used to minimize misalignment. Future designs would benefit from the use of one piece yoke members. The elbow joints house the system actuators. The face plate of the gearhead is used to secure the motor to the yoke. The requirement that the LIBRA be planar was conducive this type of actuation housing. Table 5.2 shows the actuator specifications for the chosen actuators (Portescap US, Inc. of Hauppauge, New York model number: 23DT12-R32).

<table>
<thead>
<tr>
<th>Measuring Voltage</th>
<th>18 V</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Load Speed</td>
<td>10 rpm</td>
</tr>
<tr>
<td>Stall Torque</td>
<td>20 Nm</td>
</tr>
<tr>
<td>Power Output</td>
<td>8.1 W</td>
</tr>
<tr>
<td>No Load Current</td>
<td>0.020 A</td>
</tr>
<tr>
<td>Terminal Resistance</td>
<td>10 Ohms</td>
</tr>
<tr>
<td>Torque Constant</td>
<td>23.3 mN/A</td>
</tr>
</tbody>
</table>

Table 5.2 Actuator specifications
A high gear reduction ratio, 792:1, was chosen in order to produce torques adequate to support the entire climbing mechanism. Motion is transferred via a set screw on the motor shaft. Each joint is equipped with a 0.25 inch inner diameter precision ball bearing capable of handling loads up to 40 N.
5.2.2 Body Description

Figure 5.3 is a schematic of the body. The body houses three motors, with each motor affixed to the body in a manner similar to that described in Section 5.2.1. The body consists of two .25" thick plates shaped to enhance the kinematic workspace of the climber. These body plates are separated by a 4.5" spacer tube which serves to house the system electronics. A connector for the system umbilical cord to a junction panel is affixed to the body.

Also incorporated into the body is an inclinometer consisting of a damped encoder and a pendulum. The device senses an absolute angle from which to determine the overall system configuration. Due to the nature of its design, however, it is best suited for low speed applications.
Figure 5.3 Body drawing of the LIBRA; note the housing of the actuators for planar climbing
5.2.3 Foot Description

Again, a modular approach to the design of the foot led to several designs. Figure 5.4 depicts one. As is illustrated in the figure, the foot consists of two 2 inch diameter model car tires and an aluminum rod which rotates in the bearings of the yoke member. As Section 5.2.4 explains, the flexibility to climb with round feet results from the "L" shaped parallel ladder environment structure. Depending on the environment/wall a different foot design can be used. Chantal Moore has designed and built other feet which can be interchanged with the ones depicted in Figure 5.4.

Figure 5.4 Foot Design
5.2.4 Wall Description

For the purposes of experimentation, a wall was designed and built. Each wall consists of L-shaped steps similar to the rungs on an industrial ladder at adjustable intervals. The two walls are parallel and separated by one foot.

5.2.5 Summary of Mechanical Design

Table 5.3 gives a summary of the final design of the LIBRA. Notice that the specifications of Table 5.1 are all met.

<table>
<thead>
<tr>
<th>Item</th>
<th>Final Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link Length</td>
<td>5.625 inches</td>
</tr>
<tr>
<td></td>
<td>Total Span: 14.5 inches</td>
</tr>
<tr>
<td>Weight</td>
<td>12.5 lb.</td>
</tr>
<tr>
<td>Joint Workspace</td>
<td>Each Shoulder Link: 123 degrees</td>
</tr>
<tr>
<td></td>
<td>Each Elbow Link: 210 degrees</td>
</tr>
<tr>
<td>Wall</td>
<td>2 Parallel ladders 12 inches apart</td>
</tr>
<tr>
<td></td>
<td>6 inch rung separation</td>
</tr>
</tbody>
</table>

Table 5.3 Specifications of the LIBRA Design

Appendix C contains all the mechanical design drawings used in the fabrication of the LIBRA.
5.3 Electrical Design Description and Total System overview

This section briefs the design of the servo amplifiers and relays a system overview. The servo amplifiers were designed by Craig Sunada. The design is discussed here briefly.

5.3.1 Servo Amplifier Design Description

The servo amplifier design revolves around the LM12 operational amplifier. A schematic is shown in Figure 5.5. The power operational amplifier is driven by two power supplies operating at +24 Volts and -24 Volts. The input signal to the op amp is received from the Programmable Multi-Axis Controller (PMAC), a Delta Tau Data Systems controller board capable of controlling eight axis, over the J1-PM1 line. The board is able to run lower level control algorithms such as proportional-derivative control schemes. For the purposes of this study, however, the PMAC is used as an I/O device with the control relegated to a 68030 microprocessor based computer board, as is discussed in Section 5.3.2. The servo amplifier circuit is equipped with safety devices to insure the safe transmission of power to the actuators. These include heat sinks and a fan to hasten the dissipation of the heat produced by the LM12 chip. In addition simple diodes and an RC circuit are added for safety as the schematic illustrates. These insure that a sudden jump in current will be dissipated through a resistor and not through the DC motors. The gain of the amplifier is set by adjusting the variable resistor value. For the purposes of experimentation, this gain was set at 0.29 amps/volt. To account for the input bias current of the LM12, the PMAC is adjusted. The
output current lines J2-M1 and J2-M2 are connected to the leads of each motor. There are 6 servo amplifier cards, one for each actuator.

![Servo amplifier circuit diagram]

Figure 5.5 Servo amplifier circuit diagram

5.3.2 Total System overview

The software to implement high level control is written in C programming language on a Sun workstation in the control room. See Figure 5.6. This program is downloaded onto the 68030 microprocessor-based computer board via an Ethernet link. The board resides on a VME bus line through which it is able to communicate to the Programmable Multi-Axis Controller, which is a dedicated eight axis controller board capable of performing simple position-derivative-integral control on each axis. For purposes of this experiment, however, the PMAC is used as in input/output device. From it the 68030 reads encoder positions and defines from them the current position of each control point. The encoder positions are feed from the LIBRA to a junction
panel and then to the PMAC via a 30 pin ribbon cable. In addition, the 68030 downloads to the PMAC a DAC number corresponding to the desired voltage output to each motor. These commands are routed through a junction panel to the servo amplifiers. The servos then magnify the input voltage to an output current to be sent to the actuators on the robot. This is done via a power cable umbilical cord.

![Diagram of Experimental System](image)

Figure 5.6 Overview of Experimental System

### 5.4 Summary

This chapter has discussed the LIBRA, a three legged planar climbing robot. The design is detailed. Both mechanical and electronics hardware are discussed.
6 Experimental Implementation of the CMFM Control Algorithm

This chapter presents the results of experimental implementation of the Coordinated Multi-Point Force and Motion control algorithm developed in Chapter Three and designed in Chapter Four. The Experimental system is described in Chapter Five. The chapter begins by introducing the path planning algorithm for the LIBRA. The chapter next introduces the software written in the C programming language which is responsible for the real time control of the LIBRA. The software is developed on the SUN workstation and subsequently downloaded and executed by the VxWorks operating system on a Heurikon 68020 control board. The software facilitates the interface between the high level control on the 68020 and the low level output and feedback PMAC board. Next the Chapter presents preliminary results obtained from a pull up motion executed by the LIBRA. The response to a step input is shown.

6.1 Path Planning Algorithm

As described in Chapter Five, the environment which the LIBRA will climb is composed of two parallel walls of angle iron steps. The following path planning scheme is used to move the LIBRA through this environment. Figure 6.1 illustrates the various steps involved in the climbing sequence. The steps are indicated to the right of the figure. Each elbow joint is
numbered either 1, 2, or 3 in order to identify which leg with which the joint corresponds.

Step One involves a pull up motion which elevates the body position from the initial level one rung below the upper limbs to a level even with the feet of the robot. Step Two moves the third leg to the wall on which the second leg exerts a horizontal force. Step Three transfers the load from leg two to leg three, thus freeing leg two to take a step up which is the motion involved in step four. Step Four is a series of three commanded motions. First, leg two is commanded to move away from the wall. It is then commanded to move up one step length. Finally, leg two is move over to the wall. Here the transition to Step Five occurs. In this phase, the load is transferred back to leg two from leg three. Step Six sees the motion of leg three to the opposite wall. Step Seven liberates leg one for motion by transferring its load to leg three. Step Eight mirrors Step Four by moving first leg one away from the wall. Next in Step Eight leg one is moved upwards and finally over to the rung. Step Nine transfers the load from leg three to leg one. Step Ten moves leg three back to its home position. Here the sequence begins again. This position is referred to in the Figure 6.1 as the home position.

In this manner, the robot is able to scale the environment described in Chapter Five. This describes only one mode of climbing. The LIBRA is capable of climbing in different manners which will not be discussed in this thesis. One major constraint of the LIBRA design is that the legs are not able to cross over each other. As is discussed in Chapter Five, each joint is limited to move within a workspace which is less than the full 360 degrees of motion.
Figure 6.1 Motion Path for the climbing of a 3 legged climbing robot—10 step cyclical function
The path described here is implemented on the experimental system and does not violate this joint constraint.

6.2 CMFM Software Description

The software to control the LIBRA is discussed below. Currently, the two control algorithms are implemented. The first is conventional PID control. This software was written by Tom Corrigan, Craig Sunada, and myself. The second is the software to implement CMFM control. An interface program allows the user to toggle between these control algorithms as well as to define parameters peculiar to each algorithm, such as control matrix gains.

Figure 6.2 diagrams the flow of the system software. The "LibraMove" routine declares the global variables of the software. Next, the "PMACinitialize" routine defines the parameters on the PMAC required for communications. Included in this routine are the communication functions required to interface with the PMAC over the VME bus via its 16 register locations. Also, this routine defines the interrupt service routine which is vectored when the PMAC sends out a Level Four interrupt. The PMAC does so each time it is prompted for information or has successfully read data. The "libra_setup" routine sets up the particular Programmable Logic Control routines for the LIBRA. These Routines reside in the PMAC memory and are executed at each clock cycle of the PMAC. The PMAC manual should be referenced before any changes to this software are implemented.
Figure 6.2 Software Overview Block Diagram

In addition, the "libra_setup" routine defines the PMAC parameters which control such things as encoder decode methodology and proportional,
derivative, and integral gains for joint PID motions. These gains for the
current system were determined using the Executive Software program
supplied by Delta Tau Data Systems. This software resides on an AT&T
computer which interfaces with the PMAC via an R32 serial port cable.

The "init_offset" vector is cleared on execution of the "LibraMove" routine.
This parameter is set when the LIBRA is put into its home position. The
program now branches into one of three user selected subroutines:

- "Manual_Mode": PID control of joint angles
- "Jac_transpose": begins the CMFM control algorithm
- Help menu

Figure 6.3 depicts the functions which control of each joint under PID control.
These functions are used to set the initial conditions of the LIBRA before
climbing motions begin. "Turn_on_pid" shifts the control of the robot from
the 68020 board to the PMAC. In essence, the user supplies desired positions
and the PMAC closes the loop around the encoders and performs the
standard PID control. The PMAC is equipped with an integral windup
prevention device and can be programmed to include filters. The CMFM
control algorithm does not use these control capabilities of the PMAC board.
Instead, it renders the PMAC an input output device, using its counters to
track joint positions and its DAC output to produce the control voltage
proportional to the control torque which is determined in the manner
described in Chapters Three and Four. The "get_new_command_position"
and "send_new_command_position" commands are well named and
documented in code. The user should have no difficulty in understanding these routines. The "read_manipulator_position" routine interacts with the Dual Ported Ram to obtain the position of each joint and a display routine shuffles this to the user's screen. The "redefine_home" routine simply takes the current position of the robot and subtracts it from all subsequent feedback information by setting the init_offset vector.

Figure 6.3 PID Mode System Software Diagram

Figure 6.4 illustrates graphically the software options available for the CMFM Jacobian transpose controller. Again, global variables are declared initially. Next, the system clock is initialized to interrupt every 10.1 milliseconds. This
is the determined cycle time of the control software. At each interrupt, the control routine is executed and a flag is set which tells the interrupt routine that the program has executed completely. As Figure 6.4 shows the interrupt service routine looks for this flag and if it is not set sends an error message to the user indicating that the specified interrupt rate is too fast.

In addition, before starting the interrupt driven control software, the control parameters must be set. The environment parameters such as wall separation and step size are assumed to be known and given in this subroutine. Option Two allows the user to define the system gain matrices. Also, Option Four allows the user to use a jig to place the robot in a desired configuration and set the initial conditions. Option Five is a data gathering routine written by Andrew Kuklinski. Finally, a help option describes the functionality of each option.
Figure 6.4 CMFM Control Options

Figure 6.5 shows the clock interrupt structure governing the execution of the CMFM control algorithm. Once the system is started, the control algorithm is faithfully executed every 10.1 milliseconds.
The various functions which are called in the execution of the control algorithm are shown in Figure 6.6.
Figure 6.6 CMFM Control Algorithm
The joint angles are measured by the PMAC and read over the Dual Ported RAM into the 68020 board. These angles are then transcribed into the analysis angles used in the development of the control algorithm in Chapter 4. The control point positions are calculated and desired positions are defined as a function of the path described in Chapter 4. The Jacobian matrix is evaluated at the current position and a gravity estimate is obtained. Finally the control torques are calculated using the algorithm defined in Chapter 3. These torques are then transformed into the torques actually applied to the LIBRA. Appendix D details the required transformation. Next, these control torques are converted to an equivalent DAC output number. This conversion is accomplished taking into account the gearhead reduction ratio, the geartrain efficiency, the servo amplifier gains, the torque constant, and a safety limit of 1 amp for the maximum allowable current going to the motors.

6.3 Summary

This chapter has detailed the experimental system. The path plan algorithm implemented in code is illustrated in Section 6.1. Section 6.2 discusses the software both for the PID control of the joints of the LIBRA and the implementation of the Jacobian based CMFM control algorithm. Currently, the LIBRA is being debugged.
The goal of this thesis has been to generate a unified algorithm for the simultaneous control of both force and motion of multi-limbed robotic systems and to demonstrate the capabilities of the algorithm in simulations and through experimentation. A prototype multi-limbed climbing robot was designed and fabricated for the implementation of the control algorithm. This chapter summarizes the control technique developed and the results obtained. Further, this chapter provides recommends areas for future research.

7.1 Control Conclusions

This thesis has developed a new control technique for the control of multi-limbed robotic systems. The control algorithm is known as Coordinated Multi-Point Force and Motion (CMFM) control. The motivation guiding the development of the control algorithm is that in future applications, robotic systems may be required to exert controlled forces against environments while simultaneously maintaining a desired orientation and motion. In addition, these robotic system may be required to work in environments which necessitate the use of multiple limbs capable of scaling or climbing.

The control algorithm builds on the principles established in impedance and hybrid control techniques. It distinguishes itself from these control techniques in that it applies the techniques to control force and motion at
multiple points and directions of multi-limbed systems in a unified manner. The application of this control algorithm extends beyond the control of a manipulator end point.

This thesis establishes the conditions under which it is possible to apply the CMFM control strategy and cautions that the mobility of a multi-limbed robotic system must be analyzed before applying the technique. This analysis is demonstrated for the case of a climbing robot.

This thesis demonstrates the stability of the control technique. Using the theory developed by Lyapunov, the convergence of control points of multi-limbed robotic systems to the desired positions and forces is proven.

### 7.2 Climbing Robot

The CMFM control strategy is applied in simulation to a prototype climbing robot. A complete design of the climbing robot controller demonstrates the techniques used to decide system gains. The results indicate the control strategy is capable of good command following for both motion and force control applications.

This thesis results in the design and fabrication of a climbing robot. A planar three legged robot is fabricated to serve as a testbed for this control algorithm as well as for the future algorithms for path planning and control of multi-limbed robotic systems. The robot is called the Limbed Intelligent Basic Robotic Ascender (LIBRA).
Currently, the CMFM control strategy is being applied to this climbing robot.

7.3 Future Research

The continued study of the LIBRA is important for the complete understanding of the implementation of the CMFM control algorithm developed in this thesis. While this simulation results indicate the linear model's ability to do a push up motion, the LIBRA needs to be able to climb up the stepped environment to demonstrate completely the facets of the control algorithm. The digital control ramifications of the control algorithm remain to be studied. These issues include sampling rate which may have a significant effect on the ability to implement the technique in real time.

Currently, a power optimization strategy is being developed in order to allow for the autonomous climbing of the LIBRA. The application of robotic systems to perform tasks in remote locations is often inhibited by the use of umbilical cords.

In the area of simulation, a complete computer graphic simulation is desirable for the visual understanding of climbing issues. A model of the system non backdrivability is needed to define the total effort actually required. The idea is that the non backdrivability of the actuators on the LIBRA provides some of the effort required to climb, the remaining
difference can be supplied by the system hardware. This would eliminate some of the power usage.

In the area of design, there exist several improvements which can be made on the LIBRA. First, the yoke members design, while allowing for actuator flexibility, results in a potential for misalignment. The use of one piece yokes for the joints is advisable. In addition, with the use of Computer Numerical Control machinery, precise parts can be made to fit together through a series of grooves and keys. This would eliminate the need for dowel pins while maintaining the flexibility required of a prototype system. Further, the method of actuation transmission is to use a set screw on the output shaft of the gearhead. While functional, this technique may slippage to occur at the set screw-shaft interface.

7.4 Summary

This thesis results in the development of a control technique to simultaneously control motion and force of multiple points and directions on multi-limbed robotic systems. The technique is proven theoretically and in simulation. The thesis results further in the design and fabrication of a climbing robot. The control technique gains are designed for this system to meet specifications and subsequently implemented on the climbing robot. Future research direction is given in the areas of experimentation, control, power, and design.
References


References


References


References


References


Appendix A details the development of the Jacobian for the prototype climbing robot described in Section 4.1. The Jacobian matrix maps small joint motions into small control point motions. The vector of control points for the prototype climbing system is:

\[
\mathbf{u} = \begin{bmatrix}
    x_{\text{body}} \\
    y_{\text{body}} \\
    \theta_{\text{body}} \\
    x_{\text{foot}_2} \\
    x_{\text{foot}_3} \\
    y_{\text{foot}_3}
\end{bmatrix}
\]  \hspace{1cm} (A1)

Figure A.1 depicts the prototype system described in Chapter Four. The joint angle definitions are repeated here for clarity. The convention used is that all angles and torques are positive counter clockwise. The angles are defined as follows:

- $\phi$ is the absolute angle measured from the horizontal to link 1
- $\theta_1$ is the absolute angle measured from the horizontal to link 2
- $\theta_2$ is the relative angle between link 2 and the body link
- $\theta_3$ is the relative angle between link 3 and link 4
- $\theta_4$ is the relative angle between link 4 and link 5
- $\theta_5$ is the relative angle between link 3 and link 6
- $\theta_6$ is the relative angle between link 6 and link 7.
In Figure A.1,

- \( L \) is the one-half the link length
- \( N \) is the distance between centers of actuators on the robot body
- \( r \) is the distance from the actuator to the center of mass of the body.

![Diagram of Prototype Climbing Robot Schematic](image)

**Figure A.1** Prototype Climbing Robot Schematic for Development of System Jacobian

The \( x_{\text{body}} \) position can be formulated as a function of the robot joint angles:
\[ x_{\text{body}} = 2L \cos(\phi) + 2L \cos(\theta_1) + r \cos\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right) \]  
(A2)

where the angle \( \phi \) is defined as:

\[ \phi = \text{asin}(D) \]  
(A3)

and where \( D \) is given by:

\[ D = \frac{Y - 2L [\sin(\theta_1) + \sin(\theta_1 + \theta_2 + \theta_3) + \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4)] - N \sin(\theta_1 + \theta_2)}{2L}. \]  
(A4)

The equation relating joint angles to the \( y_{\text{body}} \) position is:

\[ y_{\text{body}} = 2L \sin(\phi) + 2L \sin(\theta_1) + r \sin\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right). \]  
(A5)

The orientation of the body is simply:

\[ \theta_{\text{body}} = \theta_1 + \theta_2. \]  
(A6)

The foot position on leg 2 is given as:

\[
x_{\text{foot}_2} = 2L \cos(\phi) + 2L \cos(\theta_1) + N \cos(\theta_1 + \theta_2) \\
+ 2L \cos(\theta_1 + \theta_2 + \theta_3) + 2L \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4).
\]  
(A7)

The foot \( x \) position of leg 3 is given as:
\[
\begin{align*}
x_{\text{foot}_3} &= 2*L*\cos(\phi) + 2*L*\cos(\theta_1) + N*\cos\left(\theta_1 + \theta_2 - \frac{\pi}{3}\right) \\
&+ 2*L*\cos\left(\theta_1 + \theta_2 + \theta_5 - \frac{\pi}{2}\right) + 2*L*\cos\left(\theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2}\right).
\end{align*}
\]  
(A8)

Finally, the \(y\) position of the foot of leg 3 is given as:

\[
\begin{align*}
y_{\text{foot}_3} &= 2*L*\sin(\phi) + 2*L*\sin(\theta_1) + N*\sin\left(\theta_1 + \theta_2 - \frac{\pi}{3}\right) \\
&+ 2*L*\sin\left(\theta_1 + \theta_2 + \theta_5 - \frac{\pi}{2}\right) + 2*L*\sin\left(\theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2}\right).
\end{align*}
\]  
(A9)

To formulate the Jacobian matrix, simply take the derivative of \(u\):

\[
\dot{u} = \begin{bmatrix}
\dot{x}_{\text{body}} \\
\dot{y}_{\text{body}} \\
\dot{\theta}_{\text{body}} \\
\dot{x}_{\text{foot}_2} \\
\dot{x}_{\text{foot}_3} \\
\dot{y}_{\text{foot}_3}
\end{bmatrix} = J(\Theta)\dot{\Theta}.
\]  
(A10)

The following are the elements of the Jacobian. The subscript \(J_{jk}\) indicates the element in the \(j\)-th row and the \(k\)-th column of the Jacobian matrix.
\[
J_{11} = \frac{D}{\cos(asin(D))} \left[ 2L^* \left( \cos(\theta_1) \right) + \cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right] + N^* \cos(\theta_1 + \theta_2) \\
-2L^* \sin(\theta_1) - r^* \sin(\theta_1 + \theta_2 - \frac{\pi}{6}) 
\]

(A11)

\[
J_{12} = \frac{D}{\cos(asin(D))} \left[ 2L^* \left( \cos(\theta_1 + \theta_2 + \theta_3) \right) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right] + N^* \cos(\theta_1 + \theta_2) \\
-r^* \sin(\theta_1 + \theta_2 - \frac{\pi}{6}) 
\]

(A12)

\[
J_{13} = \frac{D}{\cos(asin(D))} \left[ 2L^* \left( \cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right) \right] 
\]

(A13)

\[
J_{14} = \frac{D}{\cos(asin(D))} \times 2L^* \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) 
\]

(A14)

\[
J_{15} = 0 
\]

(A15)

\[
J_{16} = 0 
\]

(A16)
\[ J_{21} = -2^* L^* \left[ \cos(\theta_1) + \cos(\theta_1 + \theta_2 + \theta_3) + N^* \cos(\theta_1 + \theta_2) \right] \\
+2^* L^* \cos(\theta_1) + r^* \cos\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right) \tag{A17} \]

\[ J_{22} = -2^* L^* \left[ \cos(\theta_1 + \theta_2 + \theta_3) + N^* \cos(\theta_1 + \theta_2) \right] \\
+\cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) + r^* \cos\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right) \tag{A18} \]

\[ J_{23} = -\left[2^* L^* \cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4)\right] \tag{A19} \]

\[ J_{24} = -\left[2^* L^* \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4)\right] \tag{A20} \]

\[ J_{25} = 0 \tag{A21} \]

\[ J_{26} = 0 \tag{A22} \]

\[ J_{31} = 1 \tag{A23} \]

\[ J_{32} = 1 \tag{A24} \]

\[ J_{33} = 0 \tag{A25} \]
\[ J_{34} = 0 \quad (A26) \]

\[ J_{35} = 0 \quad (A27) \]

\[ J_{36} = 0 \quad (A28) \]

\[
J_{41} = \frac{D}{\cos(\text{asin}(D))} \left[ 2 \cdot L \left( \cos(\theta_1) + \cos(\theta_1 + \theta_2 + \theta_3) \right) + N \cdot \cos(\theta_1 + \theta_2) \right]
- 2 \cdot L \cdot \sin(\theta_1) - N \cdot \sin(\theta_1 + \theta_2) - 2 \cdot L \cdot \left( \sin(\theta_1 + \theta_2 + \theta_3) + \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right)
\quad (A29) \]

\[
J_{42} = \frac{D}{\cos(\text{asin}(D))} \left[ 2 \cdot L \left( \cos(\theta_1 + \theta_2 + \theta_3) \right) + N \cdot \cos(\theta_1 + \theta_2) \right]
- N \cdot \sin(\theta_1 + \theta_2) - 2 \cdot L \cdot (\sin(\theta_1 + \theta_2 + \theta_3) + \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4))
\quad (A30) \]

\[
J_{43} = \frac{D}{\cos(\text{asin}(D))} \left[ 2 \cdot L \cdot (\cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4)) \right]
- 2 \cdot L \cdot (\sin(\theta_1 + \theta_2 + \theta_3) + \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4))
\quad (A31) \]

\[
J_{44} = \frac{D}{\cos(\text{asin}(D))} \left[ 2 \cdot L \cdot \cos(\theta_1 + \theta_2 + \theta_3) \right] - 2 \cdot L \cdot \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4)
\quad (A32) \]
\[ J_{45} = 0 \quad (A33) \]

\[ J_{46} = 0 \quad (A34) \]

\[ J_{51} = \frac{D}{\cos(\text{asin}(D))} \left[ 2L \left( \cos(\theta_1) \begin{pmatrix} +\cos(\theta_1 + \theta_2 + \theta_3) \\ +\cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \end{pmatrix} + N\cos(\theta_1 + \theta_2) \right) \\
-2L\sin(\theta_1) - N\sin\left(\theta_1 + \theta_2 - \frac{\pi}{3}\right) - 2L\sin\left(\theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2}\right) \right] \quad (A35) \]

\[ J_{52} = \frac{D}{\cos(\text{asin}(D))} \left[ 2L\left( \cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right) + N\cos(\theta_1 + \theta_2) \right] \\
-N\sin\left(\theta_1 + \theta_2 - \frac{\pi}{3}\right) - 2L\sin\left(\theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2}\right) \quad (A36) \]

\[ J_{53} = \frac{D}{\cos(\text{asin}(D))} \left[ 2L\left( \cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right) \right] \quad (A37) \]

\[ J_{54} = \frac{D}{\cos(\text{asin}(D))} \left[ 2L\cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right] \quad (A38) \]
\[ J_{55} = -2L \left[ \sin \left( \frac{\theta_1 + \theta_2 + \theta_5}{2} - \frac{\pi}{2} \right) + \sin \left( \theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2} \right) \right] \]  
(A39)

\[ J_{56} = -2L \left[ \sin \left( \frac{\theta_1 + \theta_2 + \theta_5 + \theta_6}{2} - \frac{\pi}{2} \right) \right] \]  
(A40)

\[ J_{61} = - \left[ 2L \left( \begin{array}{c} \cos(\theta_1) \\ + \cos(\theta_1 + \theta_2 + \theta_3) \\ + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \end{array} \right) + N \cos(\theta_1 + \theta_2) \right] - 2L \left( \begin{array}{c} \cos \left( \frac{\theta_1 + \theta_2 + \theta_5}{2} - \frac{\pi}{2} \right) \\ + \cos \left( \theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2} \right) \end{array} \right) \]  
(A41)

\[ J_{62} = - \left[ 2L \left( \cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right) + N \cos(\theta_1 + \theta_2) \right] - N \cos \left( \frac{\theta_1 + \theta_2}{3} - \frac{\pi}{2} \right) - 2L \left( \begin{array}{c} \cos \left( \frac{\theta_1 + \theta_2 + \theta_5}{2} - \frac{\pi}{2} \right) \\ + \cos \left( \theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2} \right) \end{array} \right) \]  
(A42)

\[ J_{63} = - \left[ 2L \left( \cos(\theta_1 + \theta_2 + \theta_3) + \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right) \right] \]  
(A43)

\[ J_{64} = - \left[ 2L \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \right] \]  
(A44)
Appendix A

\[ J_{65} = -2L \left[ \cos \left( \theta_1 + \theta_2 + \theta_5 - \frac{\pi}{2} \right) + \cos \left( \theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2} \right) \right] \]  
(A45)

\[ J_{66} = -2L \left[ \cos \left( \theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2} \right) \right] \]  
(A46)
Appendix B presents the method of the derivation of the equations of motion for the prototype climbing robot discussed in Section 4.1. Chapter Four begins the derivation. Included here are the velocity terms and the gravitational potential energy terms for each link. The Matlab M-files written for the simulation of the prototype system construct the system model based on these terms.

The following generalized variables are chosen:

\[
\Theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{bmatrix}
\]  \hspace{1cm} (B1)

These are the relative angles of the joints of the robot as indicated in Figure B.1. The convention used is that all angles and torques are positive counter clockwise. The angles are defined as follows:

- \( \phi \) is the absolute angle measured from the horizontal to link 1
- \( \theta_1 \) is the absolute angle measured from the horizontal to link 2
- \( \theta_2 \) is the relative angle between link 2 and the body link
- \( \theta_3 \) is the relative angle between link 3 and link 4
\[ \theta_4 \] is the relative angle between link 4 and link 5
\[ \theta_5 \] is the relative angle between link 3 and link 6
\[ \theta_6 \] is the relative angle between link 6 and link 7.

The generalized efforts associated with these variables are the joint torques.

\[
\tau = \begin{bmatrix}
\tau_1 \\
\tau_2 \\
\tau_3 \\
\tau_4 \\
\tau_5 \\
\tau_6 \\
\end{bmatrix}
\]  \hspace{1cm} \text{(B2)}

In Figure B.1,

\[ L \] is the one-half the link length
\[ N \] is the distance between centers of actuators on the robot body
\[ r \] is the distance from the actuator to the center of mass of the body.

Note that the following notation will be used for the unit directional vectors.

\[
\hat{i} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \hat{j} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad \hat{k} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\]  \hspace{1cm} \text{(B3)}

For ease of notation, define the following variables:

\[ c1 = \cos(\theta_1) \hspace{1cm} s1 = \sin(\theta_1) \]  \hspace{1cm} \text{(B5)}
c_{12} = \cos(\theta_1 + \theta_2) \quad \quad \quad \quad \quad \quad s_{12} = \sin(\theta_1 + \theta_2)

c_{pi} = \cos\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right) \quad \quad \quad \quad \quad \quad s_{pi} = \sin\left(\theta_1 + \theta_2 - \frac{\pi}{6}\right)

c_{123} = \cos(\theta_1 + \theta_2 + \theta_3) \quad \quad \quad \quad \quad \quad s_{123} = \sin(\theta_1 + \theta_2 + \theta_3)

c_{1234} = \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) \quad \quad \quad \quad \quad \quad s_{1234} = \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4)

c_{12 - \frac{\pi}{3}} = \cos\left(\theta_1 + \theta_2 - \frac{\pi}{3}\right) \quad \quad \quad \quad \quad \quad s_{12 - \frac{\pi}{3}} = \sin\left(\theta_1 + \theta_2 - \frac{\pi}{3}\right)

c_{125 - \frac{\pi}{2}} = \cos\left(\theta_1 + \theta_2 + \theta_5 - \frac{\pi}{2}\right) \quad \quad \quad \quad \quad \quad s_{125 - \frac{\pi}{2}} = \sin\left(\theta_1 + \theta_2 + \theta_5 - \frac{\pi}{2}\right)

c_{1256 - \frac{\pi}{2}} = \cos\left(\theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2}\right) \quad \quad \quad \quad \quad \quad s_{1256 - \frac{\pi}{2}} = \sin\left(\theta_1 + \theta_2 + \theta_5 + \theta_6 - \frac{\pi}{2}\right)

Figure B.1 Prototype Climbing Robot Schematic for Development of System Equations of Motion

For each link the total kinetic energy is given by:
Appendix B

\[ KE = \frac{1}{2} m \mathbf{v}^2 + \frac{1}{2} I \omega^2 \]  

(B5)

The velocity of the i-th link is given as:

\[ \mathbf{v}_i = (v_{ix}) \hat{i} + (v_{iy}) \hat{j} \]  

(B6)

The magnitude of this velocity is written as:

\[ \| \mathbf{v}_i \| = \sqrt{(v_{ix})^2 + (v_{iy})^2} \]  

(B7)

Notice that the square root is eliminated in taking the kinetic energy making it unnecessary to expand the components of velocity.

The velocity components in terms of the generalized variables for link 1 are derived as:

\[ v_{ix} = \frac{D}{2 \sqrt{1 - D^2}} \begin{bmatrix} 2L[c1 + c123 + c1234] + Nc12 \\ 2L[c123 + c1234] + Nc12 \\ 2L[c123 + c1234] \\ 2Lc1234 \\ 0 \\ 0 \end{bmatrix}^T \hat{\Theta} \]  

(B8)
Appendix B

\[ v_{1y} = \frac{-\cos(a \sin(D))}{2\sqrt{1 - D^2}} \times \begin{bmatrix} 2* L^* (c_1 + c_{123} + c_{1234}) + N*c_{12}^T \cr 2* L^* (c_{123} + c_{1234}) + N*c_{12} \cr 2* L^* (c_{123} + c_{1234}) 
\end{bmatrix} * \dot{\Theta} \]  
\[ \text{(B9)} \]

where

\[ \dot{\Theta} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix} \]  
\[ \text{(B10)} \]

The angular velocity of link 1 in terms of the generalized variables is:

\[ \omega_1 = \frac{-1}{2* L* \sqrt{1 - D^2}} \times \begin{bmatrix} 2* L^* (c_1 + c_{123} + c_{1234}) + N*c_{12}^T 
\end{bmatrix} * \dot{\Theta} \]  
\[ \text{(B11)} \]

The similarly velocity components of link 2 is given as:
\[ v_{2x} = \left( \frac{D}{\sqrt{1 - D^2}} \right)^* \begin{pmatrix} 2L[c_1 + c_{123} + c_{1234}] + Nc_{12} \\ 2L[c_{123} + c_{1234}] + Nc_{12} \\ 2L[c_{123} + c_{1234}] \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} Ls_1 \end{pmatrix}^T \begin{pmatrix} \hat{\Theta} \end{pmatrix} \] (B12)

\[ v_{2y} = \left( \frac{-\cos(\sin(D))}{2\sqrt{1 - D^2}} \right)^* \begin{pmatrix} 2L[c_1 + c_{123} + c_{1234}] + Nc_{12} \\ 2L[c_{123} + c_{1234}] + Nc_{12} \\ 2L[c_{123} + c_{1234}] \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} Lc_1 \end{pmatrix}^T \begin{pmatrix} \hat{\Theta} \end{pmatrix} \] (B13)

The angular velocity of link 2 is:

\[ \dot{\omega}_2 = [1 \ 0 \ 0 \ 0 \ 0] \begin{pmatrix} \hat{\Theta} \end{pmatrix} \] (B14)

The velocity components for the body link are:

\[ v_{3x} = \left( \frac{D}{\sqrt{1 - D^2}} \right)^* \begin{pmatrix} 2L[c_1 + c_{123} + c_{1234}] + \frac{N}{L}c_{12} \\ 2L[c_{123} + c_{1234}] + \frac{N}{L}c_{12} \\ 2L[c_{123} + c_{1234}] \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} Ls_1 + r*spi \\ r*spi \\ 0 \end{pmatrix}^T \begin{pmatrix} \hat{\Theta} \end{pmatrix} \] (B15)
\[ v_{3y} = \begin{pmatrix} \frac{-\cos(asin(D))}{2*\sqrt{1-D^2}} & (2*L*[c1+c123+c1234]+N*c12) \\ & (2*L*[c123+c1234]+N*c12) \end{pmatrix} \begin{pmatrix} 2*L*[c123+c1234] \\ 2*L*c1234 \end{pmatrix} \begin{bmatrix} L*c1 + r*cpi \\ r*cpi \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \] * \( \hat{\Theta} \)

(B16)

The angular velocity of link 3 is:

\[ \omega_3 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\Theta} \end{bmatrix} \]  

(B17)

For link 4 the velocity is:

\[ v_{4x} = \begin{pmatrix} D \\ \frac{D}{\sqrt{1-D^2}} \end{pmatrix} \begin{bmatrix} (2*L*[c1+c123+c1234]+N*c12) \\ (2*L*[c123+c1234]+N*c12) \end{bmatrix} \begin{pmatrix} 2*L*[c123+c1234] \\ 2*L*c1234 \end{pmatrix} \begin{bmatrix} L*(s1+s123)+N*spi \\ L*(s123)+N*spi \end{bmatrix} \begin{bmatrix} L*(s123) \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \]  

(B18)
\[
\begin{align*}
V_{4y} &= \begin{pmatrix}
-\frac{\cos(\arcsin(D))}{2\sqrt{1 - D^2}} \\
\frac{(2L*[c1 + c123 + c1234] + N*c12)}{2L*[c123 + c1234] + N*c12} \\
\frac{2L*[c123 + c1234]}{2L*[c123 + c1234]}
\end{pmatrix}^{T} \begin{pmatrix}
L*(c1 + c123) + N*c12 \\
L*c123 + N*c12 \\
L*c123 + N*c12 \\
0 \\
0 \\
0
\end{pmatrix} \ast \dot{\Theta} \\
\end{align*}
\]

(B19)

The angular velocity of link 4 is:

\[
\omega_4 = \begin{pmatrix}
1 \\
1 \\
1 \\
0 \\
0 \\
0
\end{pmatrix} \ast \dot{\Theta}
\]

(B20)
For link 5, the velocity is:

\[
\mathbf{v}_{5x} = \begin{bmatrix}
\frac{D}{\sqrt{1 - D^2}} & \left(2L^*[c1 + c123 + c1234] + N*c12\right) \\
L^*[s1 + s123 + s1234] + N*s12 & 2L^*[c123 + c1234] \\
L^*[s123 + s1234] & 2L^*[c123 + c1234] \\
L^*[s1234] & 0 \\
0 & 0 \\
0 & 0
\end{bmatrix}^T \hat{\mathbf{\theta}}
\]

(B21)

\[
\mathbf{v}_{5y} = \begin{bmatrix}
\frac{-\cos(\alpha \sin(D))}{2\sqrt{1 - D^2}} & \left(2L^*[c1 + c123 + c1234] + N*c12\right) \\
L^*[c1 + c123 + c1234] + N*c12 & 2L^*[c123 + c1234] \\
L^*[c123 + c1234] & 2L^*[c123 + c1234] \\
L^*[c1234] & 0 \\
0 & 0 \\
0 & 0
\end{bmatrix}^T \hat{\mathbf{\theta}}
\]

(B22)

For link 5, the angular velocity is:
\[ \omega_5 = [1 \ 1 \ 1 \ 1 \ 0 \ 0] \hat{\Omega} \] (B23)

For Link 6, the velocity components are:

\[
v_{6x} = \left( \begin{array}{c}
\frac{D}{\sqrt{1-D^2}} \\
(2L^*[c1 + c123 + c1234] + N*c12) \\
(2L^*[c123 + c1234] + N*c12) \\
2L^*[c123 + c1234] \\
2L^*[c1234] \\
0 \\
0 \\
0 \\
0 \\
L*s125 - \frac{\pi}{3} \\
0 \\
0 \\
L*s125 - \frac{\pi}{2} \\
0 \\
\end{array} \right)^T \hat{\Omega}
\] (B24)
\[
v_{6y} = \begin{bmatrix}
-\cos(a \sin(D)) \\
\frac{2*\sqrt{1 - D^2}}{2*\sqrt{1 - D^2}}
\end{bmatrix}^T
\begin{bmatrix}
(2*L*[c1 + c123 + c1234] + N*c12) \\
(2*L*[c123 + c1234] + N*c12) \\
2*L*[c123 + c1234] \\
2*L*c1234 \\
0 \\
0 \\
L*[c125 - \frac{\pi}{2}] \\
0 \\
0 \\
L*[c125 - \frac{\pi}{2}]
\end{bmatrix}^*\hat{\Theta}
\]

(B25)

The angular velocity of link 6 is:

\[
\omega_6 = [1 \ 1 \ 0 \ 0 \ 1 \ 0]^*\hat{\Theta}
\]

(B26)

For Link 7, the velocity components are:
Appendix B

\[
v_{7x} = \left( \frac{D}{\sqrt{1 - D^2}} \right)^* \begin{bmatrix}
(2L*[c1 + c123 + c1234] + N*c12) \\
(2L*[c123 + c1234] + N*c12) \\
2L*[c123 + c1234] \\
2L*[c1234] \\
0 \\
0 \\
2L*[s125 - \frac{\pi}{2} + \frac{1}{2}s1256 - \frac{\pi}{2}] + N*s12 - \frac{\pi}{3} \\
0 \\
2L*[s125 - \frac{\pi}{2} + \frac{1}{2}s1256 - \frac{\pi}{2}] + N*s12 - \frac{\pi}{3} \\
L*s1256 - \frac{\pi}{2}
\end{bmatrix}^T \hat{\theta}
\]

(B27)

\[
v_{7y} = \left( \frac{-\cos(\text{asin}(D))}{2\sqrt{1 - D^2}} \right)^* \begin{bmatrix}
(2L*[c1 + c123 + c1234] + N*c12) \\
(2L*[c123 + c1234] + N*c12) \\
2L*[c123 + c1234] \\
2L*[c1234] \\
0 \\
0 \\
2L*[c1 + c125 - \frac{\pi}{2} + \frac{1}{2}c1256 - \frac{\pi}{2}] + N*c12 - \frac{\pi}{3} \\
0 \\
2L*[c125 - \frac{\pi}{2} + \frac{1}{2}c1256 - \frac{\pi}{2}] + N*c12 - \frac{\pi}{3} \\
0 \\
2L*[c125 - \frac{\pi}{2} + \frac{1}{2}c1256 - \frac{\pi}{2}] + N*c12 - \frac{\pi}{3} \\
L*c1256 - \frac{\pi}{2}
\end{bmatrix}^T \hat{\theta}
\]

(B28)
The angular velocity of link 7 is:

\[ \omega_7 = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \end{bmatrix} \hat{\Theta} \]  

(B29)

Together with the system mass properties and Equation 4.3.3 the velocities characterize the system total kinetic energy.

The system potential energy incorporates the gravitational potential energy of the robot. The following describes the potential energy in each link.

For link 1, the potential energy is:

\[ U_1 = m \ast g \ast L \ast D \]  

(B30)

For link 2, the potential energy is:

\[ U_2 = m \ast g \ast (2 \ast L \ast D + L \ast s1) \]  

(B31)

The potential energy for the body is given as:

\[ U_3 = m_{body} \ast g \ast (2 \ast L \ast D + 2 \ast L \ast s1 + r \ast \text{spi}) \]  

(B32)
Link 4 has potential energy $U_4$ given by:

$$U_4 = m \cdot g \cdot (2 \cdot L \cdot D + 2 \cdot L \cdot s_1 + N \cdot s_{12} + L \cdot s_{123})$$

(B33)

Link 5 has potential energy described as:

$$U_5 = m \cdot g \cdot (2 \cdot L \cdot D + 2 \cdot L \cdot [s_1 + s_{123}] + N \cdot s_{12} + L \cdot s_{1234})$$

(B34)

The potential energy of link 6 is:

$$U_6 = m \cdot g \cdot \left( 2 \cdot L \cdot D + 2 \cdot L \cdot s_1 + N \cdot s_{12} - \frac{\pi}{3} + L \cdot s_{125} - \frac{\pi}{2} \right)$$

(B35)

The potential energy of link 7 is:

$$U_7 = m \cdot g \cdot \left( 2 \cdot L \cdot D + 2 \cdot L \cdot s_1 + N \cdot s_{12} - \frac{\pi}{3} + 2 \cdot L \cdot s_{125} - \frac{\pi}{2} + L \cdot s_{1256} - \frac{\pi}{2} \right)$$

(B36)

The total system kinetic energy is given by $T^*$. The total system potential energy is $U^*$. Formulating the Lagrangian

$$L = T^* - U^*$$

(B37)

and deriving according to the Lagrange equations:
\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_i} \right) - \frac{\partial L}{\partial \theta_i} = \tau_i
\]  \hspace{1cm} \text{(B38)}

will arrive at a complex set of interdependent equations. The calculations are done in several Matlab M files written for this simulation.
Appendix C contains all the mechanical drawings for the LIBRA.
Appendix C

Upper Arm Assembly Drawing
Appendix C

Ø.5 Drill

Ream to φ.1250 2 PL.

.193"

No. 6-32UNC-2B 6 PL.

.193"

3.750"

.250"

I do not care what this looks like

<table>
<thead>
<tr>
<th>Plan Climbing Robot</th>
<th>Drawn By: Dalila Argaez</th>
<th>Scale 1:1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part Leg Tube</td>
<td>Material Extruded AL 1&quot; SQ.</td>
<td>Quantity 6</td>
</tr>
<tr>
<td>Weight 0.087 lbs</td>
<td></td>
<td>Date 8-31-92</td>
</tr>
</tbody>
</table>

Comments:

dimensions to hold after black anodized

133
Appendix C

Chamfer 45° x 0.06

Ream to Ø0.1250 2 PL.

6-32 UNC-2B 0.367" DP, 2PL

<table>
<thead>
<tr>
<th>Plan Climbing Robot</th>
<th>Drawn By: Dalila Argaez</th>
<th>Scale 1:1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yoke Idler Shaft</td>
<td>Material AL 6061</td>
<td>Quantity 6</td>
</tr>
<tr>
<td>Member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weight 0.041187 lbs</td>
<td>Date 8-14-92</td>
<td></td>
</tr>
</tbody>
</table>

Comments: dimensions to hold after black anodized

Note: The hole for the idler shaft is to be a close fit tolerance
6-32UNC-2B 0.367" DP. 2PL

Ream to Ø0.1250 2 PL.

Drill and Ream Ø8.000 mm

No. 6-32UNC-2B

<table>
<thead>
<tr>
<th>Plan</th>
<th>Drawn By:</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Climbing Robot</td>
<td>Dalila Argaez</td>
<td>1:1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part</th>
<th>Material</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yoke Motor Shaft Member--R32</td>
<td>AL 6061</td>
<td>3</td>
</tr>
</tbody>
</table>

Weight 0.0418 lbs

Comments:
Dimensions to hold after black anodized

Date 8-14-92
Appendix C

Drill and Countersink for 6-32 0.137" DP 2PL.

R0.125 8PL.

R.125

Ream to Ø.1250 6 PL.

1/2" Drill

3.419"

2.805"

.375"

.125"

3.109"

6.219"

.56"

2.61"

.250"

.13"

---

<table>
<thead>
<tr>
<th>Plan</th>
<th>Drawn By:</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Climbing Robot</td>
<td>Dalila Arguez</td>
<td>1:1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part Description</th>
<th>Material</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shoulder Yoke Main Member</td>
<td>AL 6061</td>
<td>3</td>
</tr>
</tbody>
</table>

Weight 0.1042 lbs

Date 8-14-92

Comments: dimensions to hold after black anodized
### Appendix C

**Dimensions:**
- Drill and Counterbore .137" DP. for 6-32 2 PL.
- Ream to Ø.1250" 6 PL.
- 1/2 Drill
- R.125" 8 PL.
- R.125" 8 PL.
- .500" 8 PL.
- .875" 8 PL.
- .807" 8 PL.
- .188" 8 PL.
- .125" 8 PL.
- .193" 8 PL.
- 1.000" 8 PL.

**Table:**

<table>
<thead>
<tr>
<th>Plan Climing Robot</th>
<th>Drawn By: Dalila Argaez</th>
<th>Scale 1:1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part</td>
<td>Material AL 6061</td>
<td>Quantity 6</td>
</tr>
<tr>
<td>Yoke Motor Upper Member-R32</td>
<td></td>
<td>Date 11-4-92</td>
</tr>
<tr>
<td>Weight 0.0418 lbs</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Comments:**
- dimensions to hold after black anodized

---

138
Appendix C

Drill and Counterbore 0.137" DP. for 6-32 2 PL.

Ream to Ø.1250" 6 PL.

R.125
1/2 Drill

Plan
Climbing Robot

<table>
<thead>
<tr>
<th>Plan Climbing Robot</th>
<th>Drawn By: Dalila Argez</th>
<th>Scale 1:1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part</td>
<td>Material</td>
<td>Quantity 3</td>
</tr>
<tr>
<td>Elbow Yoke Main</td>
<td>AL 6061</td>
<td></td>
</tr>
<tr>
<td>Member – R22B</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weight 0.0948 lbs</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comments:
Appendix C

Ream to Ø.1250 2 PL.

6-32UNC-2B 0.367" DP. 2PL

R.062 2 PL. EQ. SP.

Detail A

<table>
<thead>
<tr>
<th>Plan</th>
<th>Drawn By:</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Climbing</td>
<td>Dalila Argaez</td>
<td>1:1</td>
</tr>
<tr>
<td>Robot</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part</th>
<th>Material</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shaft Bearing</td>
<td>AL 6061</td>
<td>9</td>
</tr>
<tr>
<td>Member</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Weight 0.041187 lbs

Comments:
dimensions to hold after black anadized
Drill and Counterbore for 6-32 0.137" DP. 4 PL EQ Spaced on a 0.625" BC

.2500" +.0000
-.0001

.500"

.250"

.875"

R0.01 MAX

---

<table>
<thead>
<tr>
<th>Plan Climbing Robot</th>
<th>Drawn By: Dalila Argaez</th>
<th>Scale 1:1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part Yoke Idler Shaft</td>
<td>Material AL 6061</td>
<td>Quantity 6</td>
</tr>
<tr>
<td>Weight 0.01855 lbs</td>
<td>Date 8-14-92</td>
<td></td>
</tr>
</tbody>
</table>

Comments:
Note: The idler shaft dia. is such that it requires a push fit on the inner race of the bearing.

dimensions to hold after black anodized
Forearm Assembly Drawing
Drill and Counterbore 0.137" DP for 6-32 2 PL.

Ream to Ø.1250" 6 PL.

1/2 Drill

.250"

.125"

.375"

.943" 1.557"

2.500"

Plan Climbing Robot

<table>
<thead>
<tr>
<th>Part</th>
<th>Material</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wrist Yoke</td>
<td>AL 6061</td>
<td>3</td>
</tr>
<tr>
<td>Main Member</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Weight 0.0495 lbs

Comments:

Drawn By: Dalila Argacz

Scale: 1:1

Date: 8-14-92

143
Appendix C

Drill Thru for M3 on a 26.000 mm BC 4 PL.

Ø32.000 mm
Ø21.000 mm
1.000"

A

Section A-A

.250"

.585"

1.500"

.250"

.500"

.188"

.813"

.125"

.375"

6-32UNC-2B 0.367" DP. 2PL

Ream to .1250" 2PL

<table>
<thead>
<tr>
<th>Plan</th>
<th>Drawn By: Dalila Argaez</th>
</tr>
</thead>
<tbody>
<tr>
<td>Climbing Robot</td>
<td></td>
</tr>
<tr>
<td>Part</td>
<td></td>
</tr>
<tr>
<td>Yoke Motor</td>
<td></td>
</tr>
<tr>
<td>Member--R32</td>
<td></td>
</tr>
<tr>
<td>Material</td>
<td>AL 6061</td>
</tr>
<tr>
<td>Quantity</td>
<td>3</td>
</tr>
<tr>
<td>Weight</td>
<td>0.0418 lbs</td>
</tr>
<tr>
<td>Date</td>
<td>11-4-92</td>
</tr>
</tbody>
</table>

Comments: dimensions to hold after black anodized
Drill and Counterbore 0.137" DP. for a 6-32 EQ. SP. on a 1.5" BC

Drill Thru for M3 12 PL. EQ. SP. on a 26 mm BC

Ø21.000 mm 3 PL. EQ. SP. on a 3.8" Circle

<table>
<thead>
<tr>
<th>Part</th>
<th>Material</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body Motor Member</td>
<td>AL 6061</td>
<td>1</td>
</tr>
<tr>
<td>Weight</td>
<td>0.32lbs</td>
<td></td>
</tr>
</tbody>
</table>

Comments:

Scale: 1:1

Drawing Date: 8-14-92
Drill and Countersink 0.137" DP. for 6-32 EQ. SP. on a 1.5" BC

7/16 Drill 3 PL. EQ. SP. on a 3.8" BC.

Ø 0.5000" +0.0002
-0.0000

120.0°

R 0.000"

See Detail A

Detail A

R 6.22 PL. EQ. SP.

<table>
<thead>
<tr>
<th>Part</th>
<th>Material</th>
<th>Quantity</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body Bearing Member</td>
<td>6061 AL</td>
<td></td>
<td>8-14-92</td>
</tr>
<tr>
<td>Weight</td>
<td>&lt; 0.33 lbs</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Plan</th>
<th>Drawn By:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Climbing</td>
<td>Dalis Arzaz</td>
</tr>
<tr>
<td>Robot</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>1:1</td>
</tr>
</tbody>
</table>
Appendix C

Drill and Tap 6-32 0.367" DP. 3 PL. EQ. SP. on a 1.5" BC

Drill Ø1.125

Ø1.875

I do not care what this looks like

5.275"

.500"

<table>
<thead>
<tr>
<th>Plan</th>
<th>Drawn By:</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Climbing Robot</td>
<td>Dalila Arguez</td>
<td>1:1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part</th>
<th>Material</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body Spacer</td>
<td>2&quot; SQ.1 Extruded AL</td>
<td>1</td>
</tr>
</tbody>
</table>

Weight 0.171 lbs

Date 8-14-92

Comments:
Appendix D describes the coordinate transformation required to implement Coordinated Multi-Point Force and Motion control on the LIBRA described in Chapter Five. The transformation is required because the joint angles which are actually available for feedback are not the same angles used in the analysis of Chapter Four. Hence, the control effort torques which the CMFM control algorithm prescribes based on the analysis angles must be transformed into the actual domain in which the torques are to be applied based on the measured angles.

Figure D.1 LIBRA Measured Angles: angles are positive as shown
Figure D.2 Prototype Climbing Robot Analysis Angles angles are positive counterclockwise

Figure D.1 shows the angles which are measured using encoders on the LIBRA. Figure D.2 repeats the angles used for the analysis of the climbing robot. Recall that for all the analysis, the convention used is that all analysis angles and torques are positive counterclockwise. The angles are defined as follows:

\[ \phi \] is the absolute angle measured from the horizontal to link 1
Appendix D

\( \theta_1 \) is the absolute angle measured from the horizontal to link 2

\( \theta_2 \) is the relative angle between link 2 and the body link

\( \theta_3 \) is the relative angle between link 3 and link 4

\( \theta_4 \) is the relative angle between link 4 and link 5

\( \theta_5 \) is the relative angle between link 3 and link 6

\( \theta_6 \) is the relative angle between link 6 and link 7.

The joint torques associated with these analysis angles are:

\[
\tau = \begin{bmatrix}
\tau_1 \\
\tau_2 \\
\tau_3 \\
\tau_4 \\
\tau_5 \\
\tau_6
\end{bmatrix}
\]

Recall, also, that:

L is the one-half the link length

N is the distance between centers of actuators on the robot body

r is the distance from the actuator to the center of mass of the body.

In order to proceed, it is necessary that the system constraints be defined. First, vertical distance between two planted feet in Figure D.2 is assumed to be fixed. In addition, the wall distance is also assumed to be fixed.

Y is the fixed vertical distance between the two planted feet. Then:
\[
Y = 2L \cdot \sin(\phi) + 2L \cdot s1 + N \cdot s12 + 2L \cdot (s123 + s1234).
\]  
(D1)

Similarly, \( W \) is the fixed wall distance. Then:

\[
W = 2L \cdot \cos(\phi) + 2L \cdot c1 + N \cdot c12 + 2L \cdot (c123 + c1234)
\]  
(D2)

Based on these constraints, the transformation between the measured angles and the analysis angles is derived.

\[
\theta_1 = \acos \left( \frac{Y \cdot C_2 + W \cdot C_1}{C_1^2 + C_2^2} \right)
\]  
(D3)

where

\[
C_1 = 2L \cdot \left[ \cos(\zeta_1) + \cos(\zeta_2 - \zeta_3) + \cos(\zeta_2 - \zeta_3 - \zeta_4) + 1 \right] + N \cdot \cos(\zeta_2)
\]  
(D4)

\[
C_2 = 2L \cdot \left[ \sin(\zeta_1) + \sin(\zeta_2 - \zeta_3) + \sin(\zeta_2 - \zeta_3 - \zeta_4) \right] + N \cdot \sin(\zeta_2)
\]  
(D5)

\[
\theta_2 = \zeta_2
\]  
(D6)

\[
\theta_3 = -\zeta_3
\]  
(D7)

\[
\theta_4 = -\zeta_4
\]  
(D8)
\[ \theta_5 = -\zeta_5 \]  \hspace{1cm} \text{(D9)}

\[ \theta_6 = \zeta_6 \]  \hspace{1cm} \text{(D10)}

The transformation from the analysis torques produced by the CMFM control algorithm which is based on the analysis angles to the real joint torques which should be applied is based on the fundamental fact that the work done in the analysis coordinate frame must be equal to the work done in the actual measured coordinate frame.

The total work done in the analysis coordinate frame is given as:

\[ \text{Work}_\theta = \tau_{1\theta} \dot{\theta}_1 + \tau_{2\theta} \dot{\theta}_2 + \tau_{3\theta} \dot{\theta}_3 + \tau_{4\theta} \dot{\theta}_4 + \tau_{5\theta} \dot{\theta}_5 + \tau_{6\theta} \dot{\theta}_6 \]  \hspace{1cm} \text{(D11)}

where \( \tau_{i\theta} \) is the torque corresponding to the \( i \)-th analysis angle.

Similarly, the work done in the measured coordinate from is given as:

\[ \text{Work}_\zeta = \tau_{1\zeta} \dot{\zeta}_1 + \tau_{2\zeta} \dot{\zeta}_2 + \tau_{3\zeta} \dot{\zeta}_3 + \tau_{4\zeta} \dot{\zeta}_4 + \tau_{5\zeta} \dot{\zeta}_5 + \tau_{6\zeta} \dot{\zeta}_6 \]  \hspace{1cm} \text{(D12)}

where \( \tau_{i\zeta} \) is the torque corresponding to the \( i \)-th measured angle.

From D11 and D12, it is clear that to transform the desired torque output in the analysis frame to a real set of torques to exert in the measured coordinate frame, a relationship is needed between joint velocities in the analysis frame and the joint velocities in the measured coordinate frame.
This relationship is derived in the following equations. The joint velocity of link 1 in is given as:

\[
\dot{\theta}_1 = \frac{-1}{\sqrt{1 - \left(\frac{Y*Y + W*C_1 + 2\lambda C_1}{C_1^2 + C_2^2}\right)^2}} \left(\begin{array}{c}
\frac{W}{C_1^2 + C_2^2} - \frac{Y*Y + W*C_1 + 2\lambda C_1}{C_1^2 + C_2^2} C_1 \\
+ \frac{Y}{C_1^2 + C_2^2} - \frac{Y*Y + W*C_1 + 2\lambda C_1}{C_1^2 + C_2^2} C_2
\end{array}\right) \dot{C}_1
\]

\[(D13)\]

where

\[
\dot{C}_1 = \begin{bmatrix}
-2*Y*\sin(\zeta_1) \\
-2*Y*[\sin(\zeta_2 - \zeta_3) + \sin(\zeta_2 - \zeta_3 - \zeta_4)] - N*\sin(\zeta_2) \\
2*Y*[\sin(\zeta_2 - \zeta_3) + \sin(\zeta_2 - \zeta_3 - \zeta_4)] \\
2*Y*[\sin(\zeta_2 - \zeta_3 - \zeta_4)]
\end{bmatrix}^T \begin{bmatrix}
\zeta_1 \\
\zeta_2 \\
\zeta_3 \\
\zeta_4
\end{bmatrix}
\]

\[(D14)\]

\[
\dot{C}_2 = \begin{bmatrix}
2*Y*\cos(\zeta_1) \\
2*Y*[\cos(\zeta_2 - \zeta_3) + \cos(\zeta_2 - \zeta_3 - \zeta_4)] + N*\cos(\zeta_2) \\
-2*Y*[\cos(\zeta_2 - \zeta_3) + \cos(\zeta_2 - \zeta_3 - \zeta_4)] \\
-2*Y*[\cos(\zeta_2 - \zeta_3 - \zeta_4)]
\end{bmatrix}^T \begin{bmatrix}
\zeta_1 \\
\zeta_2 \\
\zeta_3 \\
\zeta_4
\end{bmatrix}
\]

\[(D15)\]

The joint velocities for the remaining links are simply given as:

\[
\dot{\theta}_2 = \dot{\zeta}_2
\]

\[(D16)\]

\[
\dot{\theta}_3 = -\dot{\zeta}_3
\]

\[(D17)\]
\[
\dot{\theta}_4 = -\dot{\zeta}_4 \\
\dot{\theta}_5 = -\dot{\zeta}_5 \\
\dot{\theta}_6 = \dot{\zeta}_6.
\]

Substituting Equations D11 through D16 into Equation D9, gives the following relationship between analysis torques, and the joint torques which need to be applied to the LIBRA joints:

\[
\tau_{1\zeta} = h_{11} \tau_{1\theta} + h_{12} \tau_{2\theta} + h_{13} \tau_{3\theta} + h_{14} \tau_{4\theta} + h_{15} \tau_{5\theta} + h_{16} \tau_{6\theta}.
\]

where

\[
h_{11} = \frac{-1}{\sqrt{1 - \left(\frac{Y \cdot C_2 + W \cdot C_1}{C_1^2 + C_2^2}\right)^2}} \left(\frac{W}{C_1^2 + C_2^2} - \frac{Y \cdot C_2 + W \cdot C_1 \cdot 2 \cdot C_1}{C_1^2 + C_2^2}\right)^{-2} \cdot L \cdot \sin(\zeta_1)
\]

\[
\left(\frac{Y}{C_1^2 + C_2^2} - \frac{Y \cdot C_2 + W \cdot C_1 \cdot 2 \cdot C_2}{C_1^2 + C_2^2}\right)^{-2} \cdot L \cdot \cos(\zeta_1)
\]

(D22)
\[ h_{12} = \frac{-1}{\sqrt{1 - \left(\frac{Y*C_2 + W*C_1}{C_1^2 + C_2^2}\right)^2}} \left( \begin{array}{c} \frac{W}{C_1^2 + C_2^2} \\ -\frac{Y*C_2 + W*C_1}{C_1^2 + C_2^2} * 2*C_1 \\ +N*\sin(\zeta_2) \\ +N*\cos(\zeta_2) \end{array} \right) \cdot 2*L* \left( \begin{array}{c} \sin(\zeta_2 - \zeta_3) \\ +\sin(\zeta_2 - \zeta_3 - \zeta_4) \end{array} \right) \]  

(D23)

\[ h_{13} = \frac{-1}{\sqrt{1 - \left(\frac{Y*C_2 + W*C_1}{C_1^2 + C_2^2}\right)^2}} \left( \begin{array}{c} \frac{W}{C_1^2 + C_2^2} \\ -\frac{Y*C_2 + W*C_1}{C_1^2 + C_2^2} * 2*C_1 \\ +N*\sin(\zeta_2) \\ +N*\cos(\zeta_2) \end{array} \right) \cdot 2*L* \left( \begin{array}{c} \cos(\zeta_2 - \zeta_3) \\ +\cos(\zeta_2 - \zeta_3 - \zeta_4) \end{array} \right) \]  

(D24)

\[ h_{14} = \frac{-1}{\sqrt{1 - \left(\frac{Y*C_2 + W*C_1}{C_1^2 + C_2^2}\right)^2}} \left( \begin{array}{c} \frac{W}{C_1^2 + C_2^2} \\ -\frac{Y*C_2 + W*C_1}{C_1^2 + C_2^2} * 2*C_1 \\ +N*\sin(\zeta_2) \\ +N*\cos(\zeta_2) \end{array} \right) \cdot 2*L* \left( \begin{array}{c} \cos(\zeta_2 - \zeta_3) \\ +\cos(\zeta_2 - \zeta_3 - \zeta_4) \end{array} \right) \]  

(D25)

\[ h_{15} = 0 \]  

(D26)
\[ h_{16} = 0 \tag{D27} \]

\[ \tau_{2\zeta} = \tau_{2\theta} \tag{D28} \]

\[ \tau_{3\zeta} = -\tau_{3\theta} \tag{D29} \]

\[ \tau_{4\zeta} = -\tau_{4\theta} \tag{D30} \]

\[ \tau_{5\zeta} = -\tau_{5\theta} \tag{D31} \]

\[ \tau_{6\zeta} = \tau_{6\theta} \tag{D32} \]

These joint torques are then applied to the LIBRA to produce the desired motions and forces.