A NETWORK APPROACH TO FORCE CONTROL IN ROBOTICS AND TELEOPERATION

Robert John Anderson

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Approved for Public Release. Distribution Unlimited.
# A Network Approach to Force Control in Robotics and Teleoperation

## Abstract

This thesis studies the application of network theory to problems in robot and teleoperator control. Network models, referred to as Hilbert networks, are developed which extend the definition of standard electrical circuit elements, operating over \( \mathbb{R} \), to higher dimensional spaces.

First, Hilbert networks are used to represent manipulators. Kinematic, dynamic, Jacobian, and other relationships are found by applying Kirchoff's laws to these models, leading to considerable simplification over conventional approaches, especially in the case of closed chain systems.

Hilbert networks are next used to derive realistic models for the environment, including effects of contact/noncontact, coulomb friction, and nonlinear springs. It is shown that the Hilbert network can model distributed parameter systems such as transmission lines and flexible beams as well.

(over)
19. ABSTRACT (continued)

Using the Hilbert network approach, passive and nonpassive controller architectures are studied in detail. It is shown that the standard computed torque approach, as well as some hybrid position/force controllers, are nonpassive and can lead to instability. Passive implementations of these controllers are introduced which have desirable stability behavior and transient responses. An entirely new class of "critically damped" controllers with variable bandwidths is introduced which not only guarantee stability but also maintain constant contact stiffness and fully utilize actuators.

The application of Hilbert networks to teleoperators results in the solution of a 20-year-old problem, namely the time-delay instability problem in bilateral teleoperation. By mimicking the behavior of a passive transmission line, a control law is derived which globally stabilizes a teleoperator with force reflection, independent of the time delay in transmission of contact forces between the master and the slave. This result is proved theoretically, and verified experimentally.
A NETWORK APPROACH TO FORCE CONTROL
IN ROBOTICS AND TELEOPERATION

BY

ROBERT JOHN ANDERSON

B.S., University of California, San Diego, 1983
M.S., University of Illinois, 1986

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1989

Urbana, Illinois
A NETWORK APPROACH TO FORCE CONTROL
IN ROBOTICS AND TELEOPERATION

Robert John Anderson, Ph.D.
Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign, 1989

This thesis studies the application of network theory to problems in robot and teleoperator control. Network models, referred to as Hilbert networks, are developed which extend the definition of standard electrical circuit elements, operating over $\mathbb{R}$, to higher dimensional spaces.

First, Hilbert networks are used to represent manipulators. Kinematic, dynamic, Jacobian, and other relationships are found by applying Kirchoff's laws to these models, leading to considerable simplification over conventional approaches, especially in the case of closed chain systems.

Hilbert networks are next used to derive realistic models for the environment, including effects of contact/noncontact, coulomb friction, and nonlinear springs. It is shown that the Hilbert network can model distributed parameter systems such as transmission lines and flexible beams as well.

Using the Hilbert network approach, passive and nonpassive controller architectures are studied in detail. It is shown that the standard computed torque approach, as well as some hybrid position/force controllers, are nonpassive and can lead to instability. Passive implementations of these controllers are introduced which have desirable stability behavior and transient responses. An entirely new class of "critically damped" controllers with variable bandwidths is introduced which not only guarantee stability but also maintain constant contact stiffness and fully utilize actuators.

The application of Hilbert networks to teleoperators results in the solution of a 20-year-old problem, namely the time-delay instability problem in bilateral teleoperation. By mimicking the behavior of a passive transmission line, a control law is derived which globally stabilizes a teleoperator with force reflection, independent of the time delay in transmission of contact forces between the master and the slave. This result is proved theoretically, and verified experimentally.
ACKNOWLEDGEMENTS

I wish to thank my advisor, Dr. Mark Spong, for his guidance, support, encouragement and friendship. Mark is the kind of advisor who would bend over backwards to support his graduate students. For instance, he drove across the country so that his students could attend a robotics conference. His door has always been open, and his chalkboard has been ready to accept even the most unorthodox of ideas. He even loses at poker.

I would like to thank the members of my committee: Professors Kameshwar Poolla, Joseph Bentsman, and Juraj Medanic, as well as the entire control group in the Coordinated Science Laboratory. It is hard to imagine a better environment for the free exchange of ideas, or for the education of graduate students.

I would like to thank the National Science Foundation which supported my research through Grant Number DMC 85-16091.

I would like to thank the many people at the Jet Propulsion Laboratory who helped me in my experimental work. First, I applaud Dr. Antal Bejczy, who not only invited me to work at JPL, but has fostered the work of many other graduate students from across the nation. Second, I thank Dr. Blake Hannaford, with whom I engaged in many hours of discussions about the control of robots. Third, I thank Zoltan Szakaly, Ed Barlow, and Steve Venema, who provided the expertise I needed to get my experiments to work.

I thank my fellow graduate students, especially Russell Ramaker and Scott Bortoff. Their commiseration at the worst of times and their comradeship at the best of times have made my stay at the University an enjoyable one. I will miss the Thursday night volleyball games.

Finally, I would like to thank my wife, Gabriella. She has provided the sweetness, while the university has provided the light.
# TABLE OF CONTENTS

1. INTRODUCTION ........................................................................................................... 1

   1.1. Motivation and History ............................................................................................ 2
       1.1.1. Modeling, and the use of networks .................................................................... 2
       1.1.2. Manipulator and environment modeling ......................................................... 3
       1.1.3. Force control issues ......................................................................................... 4
       1.1.4. Teleoperation and the time delay problem ...................................................... 6

   1.2. Notation and Definitions ......................................................................................... 7
       1.2.1. Hilbert spaces .................................................................................................. 7
       1.2.2. Operator definitions ....................................................................................... 8
       1.2.3. Nonlinear and time-varying vectors and operators ........................................ 10
       1.2.4. Frequency domain definitions ........................................................................ 11
       1.2.5. Robot definitions ............................................................................................ 12
       1.2.6. Distributed parameter systems ...................................................................... 14

2. THE HILBERT NETWORK ............................................................................................. 16

   2.1. Network Elements ................................................................................................. 16
       2.1.1 The passive one-port elements .......................................................................... 17
       2.1.2. The passive two-port elements ........................................................................ 20
       2.1.3. The active elements ....................................................................................... 21
       2.1.4. Generic ports, impedances, admittances, and hybrid forms .......................... 23

   2.2. Network principles ................................................................................................. 24
       2.2.1 Tellegen’s Theorem and passivity .................................................................... 24
       2.2.2. Scattering theory ............................................................................................ 29
       2.2.3. Realization for passive LTI systems ................................................................ 30

   2.3. Network Reduction Rules .................................................................................... 32
       2.3.1. Pulling impedances through transformers and gyrators ............................... 33
       2.3.2. Cascade, series and parallel network reductions ............................................ 38
       2.3.3. Partitioning of the identity operator ............................................................... 40
       2.3.4. Reduction of a ladder network ....................................................................... 41

   2.4. The PHIDE network .............................................................................................. 45

   2.5. The Instability Problem for Nonscalar Gain Elements .......................................... 47
3. MODELING THE MANIPULATOR AND THE ENVIRONMENT........................................... 50

3.1. The Manipulator Model ....................................................................................... 50
  3.1.1. Robot spaces and frames .............................................................................. 50
  3.1.2. Transformations between frames ................................................................. 51
  3.1.3. The nondynamic network model ................................................................. 53
    3.1.3.1. The manipulator Jacobian ..................................................................... 54
    3.1.3.2. Finding the joint stiffness ..................................................................... 55
  3.1.4. Manipulator dynamics .................................................................................. 56
    3.1.4.1. Obtaining the dynamics from the model .............................................. 57
    3.1.4.2. Obtaining derivatives ........................................................................... 58
    3.1.4.3. Including gravity in the model ............................................................. 59
  3.1.5. Actuator dynamics ....................................................................................... 60
  3.1.6. Reduction of the manipulator ladder network .............................................. 60
  3.1.7. Closed chain manipulators .......................................................................... 62
  3.1.8. Examples ....................................................................................................... 64
    3.1.8.1. The two-link revolute manipulator ...................................................... 64
    3.1.8.2. The parallel link manipulator ............................................................... 66
  3.2. Representing Environmental Contact ............................................................... 70
    3.2.1. The linearized impedance model ............................................................... 70
    3.2.2. Representing point contact ....................................................................... 72
    3.2.3. Network models for distributed contact .................................................. 76
  3.3. Distributed Parameter Systems ......................................................................... 79
    3.3.1. The transmission line model .................................................................... 81
    3.3.2. Orthonormal decompositions .................................................................. 84
    3.3.3. Beams, membranes, and Maxwell’s equations ......................................... 88
  3.4. Representing Nonlinear Terms in PHIDE Form ............................................... 94

4. FORCE CONTROL PROBLEMS AND STABILITY ........................................... 103

4.1. Network Representations of Robot Control Algorithms .................................... 104
   4.1.1. Joint space, gripper space, and task space .................................................. 104
   4.1.2. Position control .......................................................................................... 105
   4.1.3. Force control and the use of force feedback .............................................. 107
   4.1.4. Computations in gripper space: impedance control ................................... 108
   4.1.5. Hybrid control ......................................................................................... 110
4.2. Computed Torque Approaches ................................................................. 113
  4.2.1. Standard computed torque .............................................................. 113
  4.2.2. The instability problem with standard computed torque ................. 114
  4.2.3. Passive implementation of computed torque .................................... 116
  4.2.4 Achieving critical damping in a PHIDE network ............................... 118
  4.2.5 Variable frequency/constant stiffness computed torque .................... 121
  4.3. Asymptotic Stability for Arbitrary Environments ............................... 123

5. TELEOPERATION ..................................................................................... 127
  5.1. The Teleoperator System .................................................................... 129
  5.2. Frequency Domain Analysis ................................................................ 134
  5.3. Stability Analysis .............................................................................. 137
  5.4. Experimental Results ....................................................................... 140
  5.5. Conclusions ...................................................................................... 148

6. CONCLUSIONS ......................................................................................... 149
  6.1. Pros and Cons of the Network Approach .............................................. 150
  6.2. Work to Be Done .............................................................................. 151
  6.3. A Final Note ..................................................................................... 154

BIBLIOGRAPHY .......................................................................................... 155

VITA ............................................................................................................. 159
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1. Passive circuit elements</td>
<td>17</td>
</tr>
<tr>
<td>2.2. Active circuit elements</td>
<td>21</td>
</tr>
<tr>
<td>2.3. Amplifier element</td>
<td>22</td>
</tr>
<tr>
<td>2.4. Network representation for arbitrary scaling</td>
<td>22</td>
</tr>
<tr>
<td>2.5. Generic networks</td>
<td>23</td>
</tr>
<tr>
<td>2.6. Partition of Hilbert network</td>
<td>25</td>
</tr>
<tr>
<td>2.7. Three-port representation</td>
<td>27</td>
</tr>
<tr>
<td>2.8. Network realization for LTI system</td>
<td>32</td>
</tr>
<tr>
<td>2.9. Equivalent impedance as seen through a transformer</td>
<td>33</td>
</tr>
<tr>
<td>2.10. Equivalent admittance as seen through a gyrator</td>
<td>34</td>
</tr>
<tr>
<td>2.11. Zero-valued Jacobian</td>
<td>35</td>
</tr>
<tr>
<td>2.12. Pulling an inertia through a time-varying transformer</td>
<td>35</td>
</tr>
<tr>
<td>2.13. Pulling a compliance through a time-varying transformer</td>
<td>37</td>
</tr>
<tr>
<td>2.14. Transformers in cascade</td>
<td>38</td>
</tr>
<tr>
<td>2.15. Transformers in parallel</td>
<td>39</td>
</tr>
<tr>
<td>2.16. Transformers in series</td>
<td>40</td>
</tr>
<tr>
<td>2.17. Decomposing a singular Jacobian</td>
<td>41</td>
</tr>
<tr>
<td>2.18. Partition of the identity operator</td>
<td>42</td>
</tr>
<tr>
<td>2.19. Ladder network in $H_1$</td>
<td>42</td>
</tr>
<tr>
<td>2.20. Reduced ladder network in $H_R$</td>
<td>43</td>
</tr>
<tr>
<td>2.21. Equivalent ladder network</td>
<td>46</td>
</tr>
<tr>
<td>2.22. Network subdivided by a power scaling element</td>
<td>48</td>
</tr>
<tr>
<td>2.23. Example of a destabilizing gain element</td>
<td>49</td>
</tr>
<tr>
<td>3.1. Robot link</td>
<td>52</td>
</tr>
<tr>
<td>3.2. Nondynamic circuit model for a single link</td>
<td>53</td>
</tr>
<tr>
<td>3.3. Nondynamic model for serial manipulator</td>
<td>54</td>
</tr>
<tr>
<td>3.4. Joint stiffness model</td>
<td>55</td>
</tr>
</tbody>
</table>
4.3. PD control with velocity feedforward network .............................................. 106
4.4. Force control network .................................................................................. 108
4.5. Gripper space controllers ............................................................................. 109
4.6. Passive hybrid controller implementation ...................................................... 112
4.7. Network representation of standard computed torque .................................... 115
4.8. Passive computed torque network ................................................................. 117
4.9. MBK network in PHIDE form ....................................................................... 119
4.10. Variable bandwidth computed torque networks .............................................. 122
4.11. Input-output stability .................................................................................. 123
4.12. Network diagram for a generic passive force control law .............................. 125
5.1. Block diagram of the teleoperator system ....................................................... 129
5.2. The uncontrolled teleoperator system ........................................................... 131
5.3. The controlled teleoperator system ............................................................... 133
5.4. Circuit representation of teleoperator ............................................................. 134
5.5. Teleoperator demonstration setup .................................................................. 140
5.6. Hard contact test for system without time delay ............................................ 141
5.7. Back-drivability test for system without time delay ......................................... 142
5.8. Back-drivability test for the system with 40 msec time delay ......................... 143
5.9. Back-drivability test for system with 40 msec delay and comp. ..................... 144
5.10. Hard contact test for system with 40 msec of time delay and comp. ............. 145
5.11. Back-drivability test for system with 200 msec delay and comp. ................. 146
5.12. Hard contact test for system with 200 msec of time delay and comp. ............ 147
6.1. Network model for the gradient method of adaptive control ......................... 153
<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1. Redefinition of Ladder Network Jacobians.</td>
<td>80</td>
</tr>
<tr>
<td>3.2. Parameters and Variables for a Cantilever Beam.</td>
<td>90</td>
</tr>
<tr>
<td>3.3. Parameters and Variables for a Flexible Membrane.</td>
<td>92</td>
</tr>
<tr>
<td>3.4. Parameters and Variables for the Electromagnetic Field Equations</td>
<td>93</td>
</tr>
<tr>
<td>3.5. Maxwell’s Equations.</td>
<td>94</td>
</tr>
<tr>
<td>5.1. Teleoperator Variables and Parameters.</td>
<td>130</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

A robot or manipulator is a complex, nonlinear, multidegree-of-freedom (DOF) mechanism which performs various tasks on its environment (e.g., car body or aircraft engine). Except for a few cases, such as spraypainting or sandblasting, the robot is dynamically coupled to the environment due to contact forces at the end effector. The correct reaction to these contact forces reduces manipulator and part wear, helps to avoid accidents, and increases the performance of the manipulator. In cases such as grinding and assembly, the manipulator’s reaction to contact forces is critical for the achievement of the task. The contact force is measured by a wrist force sensor, and this measurement is used in the control of the robot. Determining the nature of this control law is the underlying theme of this work.

Before manipulator force control laws are analyzed, models for the system are derived. First, a model of the manipulator is developed, which gives insight into the underlying structure of manipulator equations. Second, a model of the manipulator’s environment is obtained, which faithfully portrays such diverse behavior as contact/loss-of-contact, nonlinear stiffnesses, coulomb friction, and interaction with flexible structures. Finally, models for distributed parameter systems are developed, which can be used for either the manipulator or the environment.

Based on these models, a theorem is given which guarantees that a robot with a passive controller is asymptotically stable when interacting with arbitrary environments. Novel implementations of robot control algorithms, such as computed torque and hybrid control, are developed, which satisfy the conditions of this theorem. Furthermore, some commonly used control strategies which do not satisfy the conditions of this theorem are unstable when interacting with passive environments.

A teleoperator is a dual robot system, in which one robot, the slave, tracks the motion of the other robot, the master. Besides having all of the individual characteristics of two force controlled robots, the teleoperator also requires a communication channel, which causes additional difficulties. One particular difficulty is that the teleoperator tends to go unstable when a communication delay exists between the master and the slave. Although this problem was recognized twenty years ago, no solution has been found. In this work, a theoretically motivated and experimentally verified solution for the time delay problem is presented.

The manipulator and environment models, the stability proofs, the passive controller architectures, and the solution to the teleoperator problem are all based on one unifying theory -- the Hilbert network. The Hilbert network combines the passive structure of networks with the conceptual and notational advantages of vector spaces. Because the concept of the Hilbert
network is fundamental to the development of this work, the basic tenets of the Hilbert network are introduced early, and network reduction rules for the Hilbert network are given in detail.

1.1 Motivation and History

Developing a "network approach to force control in robotics and teleoperation" requires not only specific knowledge in robotics and teleoperation, but also a background in such diverse areas as nonlinear network theory, operator theory on Hilbert spaces, and transmission line theory. In this section a brief historical perspective is given for some of these areas, and some specific problems which arise in the force control of robots and teleoperators are outlined.

1.1.1 Modeling, and the use of networks

To understand the structure of a system, an engineer often resorts to using graphical models. This is especially important in nonlinear systems where the structure affects such fundamental characteristics as controllability, observability, and feedback linearizability.

Gabriel Kron was perhaps the first to apply electrical network theory to a number of problems, which include mechanical systems, biological systems, large-scale systems, distributed parameter systems, etc. (1948). Kron shocked the physics community by using such a powerful concept as tensor analysis (1939) to describe such basic systems as electrical circuits. In his work, he applies tensors to establish equivalent classes for complex systems. For instance, he shows that a wide variety of dc-motor architectures are representable by the same circuit model by using different tensor transformations (1951).

Motivated by electrical engineering examples, C. A. Desoer (1965), L. Chua (1980), and others have done extensive work with nonlinear networks. Resistor-Inductor-Capacitor (RLC) circuits consisting of strictly positive elements (i.e., monotonically nonincreasing nonlinearities restricted to the 1st and 3rd quadrants) have unique, continuable solutions, which vary continuously on the initial conditions. When certain topological conditions are met regarding the placement of resistance in the circuit, the zero equilibria are asymptotically stable. Results regarding passivity, reciprocity, losslessness, existence of hybrid representations, and realizations have all been developed for a large class of nonlinear, lumped parameter systems. Because the motivation for their work is based on electric circuits, however, these results are at times too general and at times too specific. For instance, active transistor circuits give rise to far more complex nonlinear phenomena than occur for the passive robot system, and thus require far more analysis regarding the uniqueness and existence of solutions. On the other
hand, describing the robot/environment behavior with scalar valued network elements would be nearly impossible, due to the large number of variables involved.

In 1960, H. M. Paynter introduced bond diagrams (1960) in an attempt to create an interdisciplinary modeling representation. In bond diagrams, letters are used to denote system elements, and lines or "bonds" are used to connect the elements. Careful attention is paid to the effort (force, voltage) and flow (velocity, current) variables, whose product determines the energy flow for the system. Once a bond diagram is made for a system, the dynamic equations are readily derived. Rosenberg and Karnopp give a good introduction to bond diagrams for general systems (1983).

In the last few years, bond graphs have been extended to multidegree of freedom systems (Tierney and Bos 1985, and Breedveld 1985). These bond graphs have been given the name "multibonds," and have been successfully used to describe such diverse behavior as fluid flow and economic models. Although this modeling approach originated in this country, it has not achieved widespread acceptance.

Functional analysis (e.g., Luenberger 1969, and Epstein 1970) is a well-established branch of mathematics, which is just beginning to be used by engineers. By defining vector spaces to describe system variables and by using operators to represent expressions, an elegant representation of a system is obtained. Problems that occur in robotics, especially when environmental interaction is involved, are well suited to the use of functional analysis. The wide diversity of possible tasks, environments, mechanisms, and control architectures requires the mathematical formalism that the theory provides.

In this work, vector spaces and operators on Hilbert spaces are combined with network representations for the purpose of studying robot control, in particular robot force control. The resulting model is called a "Hilbert Network." In Chapter 2, the Hilbert network is introduced, and model reduction techniques for Hilbert networks are described in detail.

1.1.2 Manipulator and environment modeling

The single body diagrams used for determining force balance equations for elementary dynamic systems are inadequate for determining manipulator equations, due to the complexity of manipulator motion. Instead, the manipulator equations are derived from Lagrangian or Hamiltonian methods. This is unfortunate. Diagrams are the heart and soul of engineering. They support the spatial intuition of the engineer, illustrating the structure of systems while exposing design flaws and solutions. It is said that a picture is worth a thousand words; it is certainly true in engineering, where one simple diagram displaces a complex description.
Any model describing a robot must be complex. A typical 6-link robot has twelve state variables, twenty-four kinematic parameters, and forty-eight inertial parameters. The complete dynamic equations for the manipulator take up pages of text. Prismatic and revolute joints, connected by either serial or parallel links, are possible. Ideally, all the equations for the system should be completely derivable from the model. Furthermore, reduction techniques should be applicable to the model without resorting directly to equations.

Modeling is especially important in the control of nonlinear systems. For example, a good model shows how and where computed torque and feedback linearization approaches should be applied, ensures that errors in deriving equations are avoided, and shows how network properties, such as passivity, should be utilized and maintained.

Some work has been done on applying network methods to robotics. Classical "impedance" control is based on network concepts, whether with bond diagrams (Hogan 1985) or circuit diagrams (Anderson and Spong 1988). The nonlinear dynamics and kinematics have also been represented using bond diagrams (Shahinpoor 1987, Anex 1984), although the more elegant "multibond" approach has not been used to date.

In Chapter 3, the Hilbert network model is used to represent both the manipulator and the environment. The network model for the manipulator shows the complete, nonlinear system. Dynamics and kinematics are found by applying Kirchhoff's laws. Environmental contact is illustrated by many different types of networks. Distributed contact models for the environment are used which lead to a natural understanding of contact tasks. A brief look at distributed parameter modeling is taken, in an attempt to illustrate the scope of the Hilbert network representation. In addition, many nonlinear systems are shown to be representable in network form, using identity valued dynamic elements and nonlinear memoryless elements. This representation proves useful in the study of stability for arbitrary environments.

1.1.3 Force control issues

The term force control in robotics refers to any number of methods in which the end effector contact force is regulated by feedback control of the robot actuators, typically based on measurements taken at the robot’s wrist. The approaches fall into two categories: hybrid position/force control and impedance control.

Hybrid position/force control, or hybrid control for short, was first proposed by Raibert and Craig (1981), based on an understanding of orthogonal position and force subspaces. The idea is that the position of the end effector and the contact force cannot be controlled independently along any one degree-of-freedom. By choosing a coordinate system aligned with the surface of the environment, which defines the so-called task space, the system is split into two
subspaces, called the position controlled and force controlled subspaces. In the position controlled subspace, the manipulator is servoed based on a position error. In the force controlled subspace, the manipulator is servoed based on an error between a desired force command and the actual contact force.

Impedance control, introduced by Hogan (1985), does not delineate between position and force controlled subspaces. Instead, the "impedance," or compliance of the manipulator is chosen to ensure the appropriate behavior of the contact forces. By closing position and velocity loops in gripper space rather than in joint space, it is possible to maintain a constant amount of damping and stiffness with respect to the environment. Thus, whatever the configuration of the manipulator, a given applied force at the gripper always results in the same deflection.

Since these two approaches were introduced, little fundamental work has been done regarding robot force control, and yet many problems remain. The modeling of the environment, whose development is critical to understanding contact tasks, is crude at best. Typical models include one-dimensional springs and dampers. For impedance control, it is assumed that the model is represented by a 6-DOF impedance operator, but little work is done showing whether this representation is valid, and, if so, how it is derived. For hybrid control, there is little distinction made between task space and world space. Task space implementations are problematic since no absolute task space measurement of position exists. Furthermore, many force control algorithms require the ideal assumption that the manipulator contains 6-DOF and is operating in a 6-DOF world space in a region free of manipulator Jacobian singularities. Because many manipulators contain more or less than 6-DOF, and because manipulator Jacobian singularities are often entered, a more general force control theory needs to be developed.

Finally, the area of stability for arbitrary environments has not yet been adequately addressed. A robot is not a machine tool destined to endlessly repeat a given task in a highly structured environment. Instead, the robot should be able to interact with a wide variety of possible environments. Furthermore, it should do so without going unstable. Many force control architectures developed to date, however, cannot make such a claim, even though achieving such a condition does not reduce the performance of the manipulator.

In Chapter 4 of this work, these problems are addressed. By using network models and "passive controller" architectures, it is possible to guarantee stability for the manipulator when operating with a wide class of arbitrary environments. It is shown how some contemporary controller approaches, such as hybrid control and computed torque control, do not have these passive controller architectures, and new implementations of these control laws are given.
which do. Motivated by the Hilbert network structure, an entirely new class of computed torque controllers is developed.

1.1.4 Teleoperation and the time delay problem

Teleoperation represents one of the first domains of robotics, and one of the most challenging (Vertut and Coiffet 1986). In teleoperation a human operator moves a master manipulator, and a slave manipulator is controlled to follow the motion while manipulating a remote environment. Providing contact information to the human operator improves task performance. Although this information can be viewed on displays, it is more useful when provided directly, by reflecting the measured force to motors on the master. When this is done, the operator is said to be kinesthetically coupled to the environment, and the teleoperator is said to be controlled bilaterally. (Bejczy and Handlykken 1981, Vertut et al. 1984).

When teleoperation is performed over a great distance, such as in undersea and outer space operations, a time delay is incurred in the transmission of information from one site to another. This time delay destabilizes a bilaterally controlled manipulator. Whitney states, that the time delay problem in teleoperation "caused the demise of a proposed use for force-feedback manipulators: their utilization in space from ground-based control stations." (Whitney 1985).

In 1965, the first work (Ferrell 1965) dealing with time delay in teleoperation appeared. Because no force feedback was being used however, instability was not a problem. In 1966, force feedback was used in the presence of time delay (Ferrell 1966). Here, the instability was apparent. Delays on the order of a tenth of a second destabilized the teleoperator. Recently, Vertut and co-workers experimented with a force-reflecting system with time delay, and achieved stability only when the bandwidth was severely reduced, allowing velocities of only 10 cm/sec (Vertut et al. 1981). Since this pioneering work, little has appeared in the literature regarding the stability of bilateral teleoperators with time delay. One reason for this is the difficulty of the analysis. The time delay turns an otherwise finite-dimensional system into an infinite-dimensional system, and the nonlinearities in manipulators further complicate the analysis.

By applying the network modeling approach and using passivity ideas, a solution is developed for the instability of teleoperation caused by time delay. Teleoperation and the time-delay solution are presented in Chapter 5.
1.2 Notation and Definitions

The notation and definitions used in this work can be divided into six different areas: Hilbert spaces; operators; nonlinear and time-varying systems; frequency domain analysis; robots; and distributed parameter systems. Each of these areas is covered in the following subsections.

1.2.1 Hilbert spaces

The analysis in this work is based on Hilbert spaces. Although there are many cases where the requirement that a vector belongs to a Hilbert space is either too general (e.g., when $H_i = \mathbb{R}$) or not general enough, (e.g., when differentiability is needed), the advantages of Hilbert space notation and theoretical development far outweigh the disadvantages.

Standard Hilbert spaces are represented by bold letters, e.g., $\mathbf{R}^n$, $\mathbf{L}_2[0,1]$. Particular Hilbert spaces are represented by $H_i$, where the subscript $i$ is used to delineate individual spaces. Vectors are represented by lower case letters, e.g., $x$, $v_i$, and $f_T$, using the same subscript as the underlying Hilbert space. For example, the statement, $v_a \in H_a = \mathbb{R}^n$, implies that the vector $v_a$ is a member of the "actuator" Hilbert space $\mathbb{R}^n$.

Associated with each Hilbert space, $H_i$, is an inner product, $\langle \cdot, \cdot \rangle : H_i \times H_i \rightarrow \mathbb{R}$, which satisfies the four axioms:

1. $\langle x_1, x_2 \rangle_i = \langle x_2, x_1 \rangle_i$, 
2. $\langle x_1 + x_2, x_3 \rangle_i = \langle x_1, x_3 \rangle_i + \langle x_2, x_3 \rangle_i$, 
3. $\langle \lambda x_1, x_2 \rangle_i = \lambda \langle x_1, x_2 \rangle_i$, 
4. $\langle x_i, x_i \rangle_i \geq 0$ and $\langle x_i, x_i \rangle = 0$ if and only if $x_i = 0$.

The inner product on the Hilbert space, $H_i$, induces the norm, $\| \cdot \| : H_i \rightarrow \mathbb{R}$, defined by

$$\| x_i \|_i = \langle x_i, x_i \rangle_i^{1/2}. \quad (1.1)$$

Some of the commonly used Hilbert spaces are listed below:

1. $\mathbb{R}^n$ denotes the usual $n$-dimensional Euclidean space over $\mathbb{R}$ with inner product,
$$\langle x_i, y_i \rangle_i = x_i^T y_i,$$
where $x_i, y_i \in H_i \triangleq \mathbb{R}^n$, and the superscript, $T$, denotes the transpose.

2. $\mathbf{L}_2^a[0,\infty)$ denotes the Hilbert space of Lebesgue measurable functions, $x : [0,\infty) \rightarrow \mathbb{R}^n$, which are square integrable over the indefinite interval, $[0,\infty)$, with inner product,
$$\langle x_i, y_i \rangle_i = \int_0^\infty x_i(t)^T y_i(t) dt,$$
where \( x_i, y_i \in H_i \triangleq L^2([0, \infty)). \)

3. \( L^2([0,1] \times [0,1]) \) denotes the Hilbert space of Lebesque measurable functions, \( x : ([0,1] \times [0,1]) \rightarrow \mathbb{R}^n \), which are square integrable over the unit square with inner product,

\[
\langle x_i, y_i \rangle_i = \int_0^1 \int_0^1 x_i(r_1, r_2)^T y_i(r_1, r_2) dr_1 dr_2,
\]

where \( x_i, y_i \in H_i \triangleq L^2([0,1] \times [0,1])^n \).

1.2.2 Operator definitions

Operators which map Hilbert spaces into Hilbert spaces are represented by capital letters, e.g., \( A : \mathbb{R}^n \rightarrow \mathbb{R}^n \), \( J : \mathbb{R}^n \rightarrow L^2([0,1]) \). The subscript notation is extended to operators with the following understanding. If the operator maps a space into itself, then a single subscript is used, e.g., \( A_i : H_i \rightarrow H_i \) is defined by \( y_i = A_i x_i \), where \( y_i, x_i \in H_i \). On the other hand, if the operator maps a space into a different space, then both subscripts separated by a slash (/) are used, e.g., \( A_{ji} : H_i \rightarrow H_j \) is defined by \( x_j = A_{ji} x_i \), where \( x_i \in H_i \) and \( x_j \in H_j \).

Let \( A_{ji} : H_i \rightarrow H_j \) be an operator between two spaces, and let \( x_i \) and \( x_j \) represent vectors in the two respective spaces. Then, consider the following definitions:

**Definition 1.1**: The adjoint of \( A_{ji} \), denoted \( A^*_{ji} \), is the unique operator mapping \( H_j \rightarrow H_i \) which satisfies

\[
\langle x_j, A_{ji} x_i \rangle_j = \langle A^*_{ji} x_j, x_i \rangle_i
\]

for all \( x_i \in H_i \) and \( x_j \in H_j \).

**Definition 1.2**: The null operator, \( 0_{ji} : H_i \rightarrow H_j \), is the unique operator satisfying \( 0_{ji} x_i = 0 \) for all \( x_i \in H_i \).

**Definition 1.3**: The induced norm on the operator \( A_{ji} \), denoted \( \| A_{ji} \| \), is given by

\[
\| A_{ji} \| = \sup_{\| x_i \|_i = 1} \| A_{ji} x_i \|_i.
\]

Let \( A_i : H_i \rightarrow H_i \) be an operator mapping a Hilbert space into itself, and let \( x_i \) and \( y_i \) be two vectors in \( H_i \). Then, consider the following definitions:
Definition 1.4: The operator \( A_i \) is Hermitian or symmetric if \( A_i = A_i^* \).

Definition 1.5: The operator \( A_i \) is skew-symmetric if \( \langle A_i x_i, x_i \rangle_i = 0 \) for all \( x_i \in \mathcal{H}_i \). Skew-symmetric operators are denoted by the capital letter \( S \).

Definition 1.6: The operator \( A_i \) is positive definite if \( \langle x_i, A_i x_i \rangle_i > 0 \) for all \( x_i \neq 0 \).

Definition 1.7: If the operator \( A_i \) is one-to-one and onto, then \( A_i \) has an inverse, denoted \( A_i^{-1} \), such that if \( A_i x_i = y_i \), then \( A_i^{-1} y_i = x_i \).

Definition 1.8: The Identity Operator, \( I_i : \mathcal{H}_i \rightarrow \mathcal{H}_i \), is the unique operator satisfying \( I_i x_i = x_i \) for all \( x_i \in \mathcal{H}_i \).

Definition 1.9: An operator, \( A_i \), is unitary if \( A_i^* A_i = A_i A_i^* = I_i \). Unitary operators are denoted by the capital letters, \( U \) and \( V \).

Theorem 1.1: If \( A_i \) is unitary and time-varying, then \( \dot{A}_i = S_i A_i \), where \( S_i \) is skew-symmetric.

Proof

\[
\frac{d}{dt}(A_i A_i^*) = \dot{A}_i A_i^* + A_i \dot{A}_i^* = \dot{A}_i A_i^* + (A_i \dot{A}_i^*)^* = \frac{d}{dt}I_i = 0.
\]

Define \( S_i = \dot{A}_i A_i^* \), then

\[ S_i A_i = \dot{A}_i A_i^* A_i = \dot{A}_i. \quad \square \]

Definition 1.10: A matrix can be decomposed using a singular value decomposition (SVD) so that \( A_{ji} = U_j \Sigma_{ji} V_i \), where \( U_j \) and \( V_i \) are unitary operators, and \( \Sigma_{ji} \) contains the terms \( (\sigma_1, \ldots, \sigma_n) \) on the main diagonal.

The terms, \( (\sigma_1, \ldots, \sigma_n) \), are called the singular values of \( A_{ji} \). It is well known that the maximum singular value is equal to the norm, \( \| A_{ji} \| \). Furthermore, if a matrix is positive definite and symmetric, then \( V_i = U_j^T \).

Definition 1.11: The square root of a positive definite symmetric matrix, \( A_i \), written \( A_i^{\frac{1}{2}} \), is the unique positive definite matrix satisfying \( A_i^{\frac{1}{2}} A_i^{\frac{1}{2}} = A_i \).
From the SVD decomposition, the square root is computed as

\[ A_i^{1/2} = U_i \Sigma_i^{1/2} U_i^T. \]  

Similarly, the inverse of a positive definite matrix, \( A_i \), is written as

\[ A_i^{-1} = U_i \Sigma_i^{-1} U_i^T. \]

Square brackets, ([,]), which are commonly used for vector and matrix notation over \( \mathbb{R}^n = \mathbb{R} \times \mathbb{R} \times \cdots \times \mathbb{R} \), are used here to represent operations involving the Cartesian product of arbitrary Hilbert spaces. For instance, if \( x_{i1} \in H_{i1}, x_{i2} \in H_{i2}, x_{j1} \in H_{j1}, x_{j2} \in H_{j2} \), and the operators, \( A_{j1/i1}, A_{j1/i2}A_{j2/i1}, A_{j2/i2} \) are defined so that \( x_{j1} = A_{j1/i1}x_{i1} + A_{j1/i2}x_{i2} \) and \( x_{j2} = A_{j2/i1}x_{i1} + A_{j2/i2}x_{i2} \) and by defining \( H_i = H_{j1} \times H_{j2} \), and \( H_j = H_{j1} \times H_{j2} \), then using the bracket notation:

\[ x_j \triangleq \begin{bmatrix} x_{j1} \\ x_{j2} \end{bmatrix} = \begin{bmatrix} A_{j1/i1} & A_{j1/i2} \\ A_{j2/i1} & A_{j2/i2} \end{bmatrix} \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix} = A_{j/i} x_i, \]

where \( x_j \in H_j, x_i \in H_i \), and \( A_{j/i} : H_i \rightarrow H_j \).

1.2.3 Nonlinear and time-varying vectors and operators

Until now, definitions have been given for linear operators operating on a linear Hilbert space. In order to gain all this "linearity" in a nonlinear world, a certain notational trick needs to be applied. Functional dependence which is not linear is deemed "parametric" dependence. For example, the equation, \( y_i = z_i x_i \), could be written as \( y_i = f(z_i, x_i) \), where \( f : H_i \times H_i \rightarrow H_i \) is a nonlinear operator. Instead, this is interpreted as \( y_i = A_i(z_i)x_i \), where \( A_i(z_i) : H_i \rightarrow H_i \) depends parametrically on \( z_i \in H_i \).

Often the temporal variable, \( t \), is subjugated in this fashion. Furthermore, because almost all variables in the network vary with time, the notational device showing this parametric dependence, i.e., the variable, \( t \), in closed parentheses, is frequently dropped. Thus when it is written that \( x_i \in H_i = \mathbb{R}^n \), what is often meant is that \( x_i(t) \in H_i = \mathbb{R}^n \ \forall t \in [0, \infty) \).

When time is included directly some additional constructs are necessary. First, because many variables in the robot equations do not approach zero as time goes to infinity, it is impossible to use Hilbert spaces based on the indefinite interval, \( t \in [0, \infty) \). Instead, with regard to the time variables, the extended Hilbert space approach is applied (Vidyasagar 1978).

**Definition 1.12:** Let \( x_i(t) \in H_i \ \forall t \in [0, \infty) \), be measurable. Then for all \( T \in [0, \infty) \) the function \((·)_T \) defined by
\[ (x_i(t))_T = \begin{cases} x_i(t) & 0 \leq t \leq T \\ 0 & T < t \end{cases} \]

is called the **truncation** of \( x_i \).

**Definition 1.13:** Let \( H_i \) be a spatial Hilbert space. The set of all measurable functions, \( x_i(t) \in H_i \), for all time, \( t \in [0, \infty) \), which satisfy

\[ \int_0^{\infty} \| x_i(t) \|^2_T dt = \int_0^T \| x_i(t) \|^2_T dt < \infty, \]

is denoted by \( IE_i \) and is called the **time extension** of \( H_i \).

Thus, when it is written that \( x_i \in H_i \), in reality \( x_i \in IE_i \), and is a member of \( H_i \) for each time, \( t \). Additional restrictions may also be made regarding the time variation of \( x_i(t) \). In many cases, differentiability is necessary. In this case it is assumed that \( x_i(t) \) is in a differentiable subspace of \( IE_i \).

If the variables in the system do approach zero as \( t \to \infty \), then it is possible to define a new space, denoted by \( H_i(t) \). The inner product in \( H_i(t) \) is defined by

\[ \int_0^{\infty} < x_1(t), x_2(t) >_i dt, \tag{1.5} \]

and thus the norm on \( H_i(t) \) is given by

\[ ||| x_i |||_i = ( \int_0^{\infty} < x_i(\tau), x_i(\tau) >_i d\tau )^{\frac{1}{2}} = ( \int_0^{\infty} \| x_i(\tau) \|^2_i d\tau )^{\frac{1}{2}}. \tag{1.6} \]

The three bar notation \( ||| \cdot ||| \) has been introduced to distinguish between the norm on the space, \( H_i \), and the norm on the space, \( H_i(t) \). This norm is called the **time norm** on the space, \( H_i(t) \).

The time norm leads to an induced norm for operators in \( H_i(t) \).

\[ ||| A_i |||_i = \sup_{x_i \in H_i} \int_0^{\infty} \| A_i x_i \|^2_T d\tau / \int_0^{\infty} \| x_i \|^2_T. \tag{1.7} \]

**1.2.4 Frequency domain definitions**

In general, the type of parametric dependency described in the previous section makes a frequency domain approach impossible. Nevertheless, there are many instances in which frequency domain techniques are still useful.

Let \( s \) denote the Laplace domain variable, and let \( \omega \) represent the Fourier domain variable. If the vector, \( x_i(t) \in H_i(t) \) is Laplace transformable, then the Laplace transform of \( x_i(t) \) is defined by
\[ x_i(s) = \int_0^\infty e^{-s\tau}x_i(\tau)d\tau. \] (1.8)

It is assumed that the Laplace transform maps from a time domain space, \( H_i(t) \), to a frequency domain space, \( H_i(s) \).

Convolution operators, which depend parametrically on the differential and integral operators, such as \( A_i(d/dt, \int_0^\infty d\tau) \), are represented in the frequency domain with direct substitution as, \( A_i(s, 1/s) \) or simply, \( A_i(s) \).

**Definition 1.14:** The frequency norm on the space, \( H_i(s) \), \( ||| \cdot |||_i \) is given by

\[ ||| x_i |||_i = \int_\omega^\infty <x_i(j\omega), x_i(j\omega)>_i d\omega = \int_\omega^\infty (|| x_i(j\omega)||_i )d\omega. \]

The use of identical notation for both the frequency norm and the time norm does not cause a problem because of the following theorem.

**Theorem 1.2 (Parseval's Identity)**

\[ ||| x_i |||_i = \int_\omega^\infty (|| x_i(j\omega)||_i )d\omega = \int_\omega^\infty (|| x_i(t)||_i )dt. \]

If \( A_i : H_i(t) \rightarrow H_i(t) \) is a convolution operator defining an LTI system with Laplace transform, \( A_i(s) \), then the induced norm on \( A_i \) is given by

\[ ||| A_i |||_i = \sup_{x_i \in H_i} \int_0^\infty \frac{|| A_i x_i ||^2}{|| x_i ||^2} dt = \sup_{\omega} || A_i(j\omega)||_i. \] (1.9)

Another property of the frequency domain operator, or transfer function, is related to the passivity of the system.

**Definition 1.15:** The frequency domain operator, \( A_i(s) \), is **positive real** if and only if \( \text{Re}(A_i(j\omega)) \) is positive semidefinite for all \( \omega \).

### 1.2.5 Robot definitions

Because this work is concerned with robot control, and not just the development of the Hilbert network concepts, many notational devices are used which are peculiar to robot equation development. For instance, the dynamics of a robot are defined with respect to certain reference frames. Additional subscript and superscript conventions are used to relate vectors in different reference frames. The vector, \( d_{ij} \in \mathbb{R}^3 \), represents the Cartesian distance between frame \( i \) and frame \( j \) computed in a fixed reference frame. Often the concept of a reference frame and the underlying space associated with vectors operating in the reference frame is obscured, since the subscripts used to denote the space denotes the reference frame as well.
Manipulators typically operate with six degrees-of-freedom (DOF), and thus many of the vectors of interest are defined on $\mathbb{R}^6$. The six DOF consist of three linear or prismatic DOF, denoted by the subscript $p$, and three angular or revolute DOF, denoted by the subscript, $\theta$. In order to distinguish between components of these vectors the subscripts, $x, y, z, \xi, \psi, \nu$, and $\zeta$, are used to denote the vectors oriented along and around the three Cartesian axes, $x, y$ and $z$. For instance, the vector, $v_i \in \mathbb{R}^6$, is represented as

$$v_i = \begin{bmatrix} v_{pi} \\ v_{qi} \end{bmatrix} = \begin{bmatrix} v_{xi} \\ v_{yi} \\ v_{zi} \\ v_{\xi_i} \\ v_{\psi_i} \\ v_{\zeta_i} \end{bmatrix}.$$  \hspace{1cm} (1.10)

A few special relationships occur on the Cartesian space, $\mathbb{R}^3$. First, consider the skew-symmetric operator, $S(r): \mathbb{R}^3 \rightarrow \mathbb{R}^3$, $r \in \mathbb{R}^3$, given by

$$S(r) = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}.$$  \hspace{1cm} (1.11)

The operator $S$ is linear, and thus for $r_1, r_2, \alpha_1, \alpha_2 \in \mathbb{R},$

$$\alpha_1 S(r_1) + \alpha_2 S(r_2) = S(\alpha_1 r_1 + \alpha_2 r_2),$$

and for $r(t) \in L^2(\mathbb{R}_+)$

$$\frac{d}{dt} S(r(t)) = S \left( \frac{dr(t)}{dt} \right).$$

The operator $S$ takes the place of the cross product in the standard development of the dynamic equations, since

$$S(r_1) r_2 = r_1 \times r_2.$$  \hspace{1cm} (1.12)

Let $SO(\mathbb{R}^3, 3)$ represent the special orthogonal group. Elements of $SO(\mathbb{R}^3, 3)$ are called rotations.

**Proposition 1.1** (Arnold 1978) Let $R \in SO(\mathbb{R}^3, 3)$. There exists a vector, $x_\theta \in \mathbb{R}^3$, such that the skew-symmetric matrix, $S$,

$$S(x_\theta) = \begin{bmatrix} 0 & -x_{\theta z} & x_{\theta y} \\ x_{\theta z} & 0 & -x_{\theta x} \\ -x_{\theta y} & x_{\theta x} & 0 \end{bmatrix}$$
is related to \( R \) by

\[
R = e^{S(x_0)}.
\]

Using these properties, the derivative of the rotation matrix, \( R(t) \), is computed as follows:

\[
\frac{d}{dt} R(t) = \frac{d}{dt} e^{S(x_0(t))} = \left( \frac{d}{dt} S(x_0(t)) \right) e^{S(x_0(t))}
\]

\[
= S \left( \frac{d}{dt} x_0(t) \right) R(t) = S(v_0(t)) R(t),
\]

where \( v_0(t) \triangleq \frac{d}{dt} x_0(t) \).

### 1.2.6 Distributed parameter systems

A number of definitions are needed for dealing with distributed parameter systems, such as transmission lines and flexible structures.

A distributed parameter system is defined by differential equations which are valid on an internal interval and boundary conditions which are valid on the end of an interval. In a network representation of the same system, however, it should be possible to derive both the differential equations and the boundary conditions from an application of Kirchoff’s laws. In order to do this, the concept of partial differential needs to be extended to include the boundary conditions.

Let \( H_c \triangleq L^2([0,1]) \), and assume that the signals, \( f_c, v_c \in H_c \), are differentiable in the open interval, \((0,1)\), and continuous over the closed interval, \([0,1]\).

**Definition 1.16:** The impulse operator, \( \delta_c(z-z_0):H_c \rightarrow H_c \), is defined as follows:

\[
\int_0^1 v_c(z) \delta_c(z-z_0) dz = v_c(z_0).
\]

**Definition 1.17:** The central partial derivative, \( \frac{\partial}{\partial z}:H_c \rightarrow H_c \), is defined as follows:

\[
\frac{\partial}{\partial z} v_c = \begin{cases} 
\frac{\partial}{\partial z} v_c(z,t) & z \in (0,1) \\
\lim_{z \to 0^+} \frac{\partial}{\partial z} v_c(z,t) & z = 0 \\
\lim_{z \to 1^-} \frac{\partial}{\partial z} v_c(z,t) & z = 1
\end{cases}
\]

where \( \frac{\partial}{\partial z} \) is the standard partial derivative.
From this definition, the left and right partial derivatives are defined.

**Definition 1.18:** The left partial derivative, \( \frac{\partial l}{\partial z} : \mathbf{H}_c \to \mathbf{H}_c \), is defined by

\[
\frac{\partial l}{\partial z} v_c = \frac{\partial c}{\partial z} v_c + \delta(z) v_c(z).
\]

**Definition 1.19:** The right partial derivative, \( \frac{\partial r}{\partial z} : \mathbf{H}_c \to \mathbf{H}_c \), is defined by

\[
\frac{\partial r}{\partial z} v_c = \frac{\partial c}{\partial z} v_c - \delta(z-1) v_c(z).
\]

Notice that all of the definitions for partial derivatives are identical on the open interval. The advantage of defining the left and right partial derivatives is seen in the next lemma.

**Lemma 1.1** The adjoint of the right partial derivative is equal to the negative of the left partial derivative, i.e., \( \frac{\partial r}{\partial z} ^* = -\frac{\partial l}{\partial z} \).

**Proof** The proof follows directly from the definition of the adjoint of the operator in the space, \( \mathbf{H}_c = L_2^n[0,1] \):

\[
\langle \frac{\partial r}{\partial z} v_c, f_c \rangle_c = \int_0^1 \frac{\partial r}{\partial z} v_c^T f_c \, dz
\]

\[
= \int_0^1 \left( \frac{\partial c}{\partial z} v_c + \delta_c(z) v_c(z) \right)^T f_c \, dz
\]

\[
= \int_0^1 \frac{\partial c}{\partial z} v_c^T f_c \, dz + v_c(0)^T f_c(0).
\]  \hspace{1cm} (1.14)

Integrating the first term by parts gives

\[
\int_0^1 \frac{\partial c}{\partial z} v_c^T f_c \, dz = v_c^T f_c \big|_0^1 - \int_0^1 v_c^T \frac{\partial c}{\partial z} f_c \, dz.
\]

Inserting this into (1.14) gives

\[
\langle \frac{\partial r}{\partial z} v_c, f_c \rangle_c = -\int_0^1 v_c^T \frac{\partial c}{\partial z} f_c \, dz + v_c^T(1)f_c(1)
\]

\[
= -\int_0^1 v_c^T \frac{\partial c}{\partial z} f_c \, dz + \int_0^1 v_c^T(\delta_c(z-1)) f_c(z) \, dz
\]

\[
= -\int_0^1 v_c^T \frac{\partial l}{\partial z} f_c \, dz
\]

\[
= -\langle v_c, \frac{\partial l}{\partial z} f_c \rangle_c. \quad \square
\]
2. THE HILBERT NETWORK

The premise of this work is that networks serve as a useful model for robot/environment interaction. Networks can be used to model complex nonlinear, distributed phenomena, without requiring unrealistic assumptions. Unlike some recent modeling approaches, such as bond graphs, networks are familiar to almost every engineer, and a significant body of knowledge already exists for dealing with networks, albeit for the scalar variety. Networks are easy to manipulate and can lead to substantial reduction in the effort involved in deriving dynamic equations. Finally, networks maintain the passive structure of a system, which is useful for proving stability of control algorithms. In this section, the Hilbert network model and an important subclass, the PHIDE network, are introduced, and a number of useful results for these networks are stated.

In order to model the nonlinear, multidegree of freedom behavior of a manipulator, manipulator/environment interactions, and materials with distributed characteristics, it is necessary to leave the scalar domain of standard circuits. By adding the additional structure of Hilbert spaces and the notational advantages of operator theory, a powerful tool for representing complex dynamic behavior is developed. This tool is a Hilbert network.

An operator theoretic description of nonlinear networks has been developed by Dolezal and Zemanian (1975) and is summarized in the book (Dolezal 1977). Here the term "Hilbert Network" was first used. Unfortunately, this book a mathematical treatise rather than a useful tool for an engineer doing research in robotics. The Hilbert network developed here is similar in philosophy, but altogether different in applications.

The Hilbert network is constructed by connecting a finite number of basic elements, where each element is associated with an effort vector (voltage or force), \( f_i \), and a flow vector (current or velocity), \( v_i \). Both \( f_i \) and \( v_i \) belong to a Hilbert space, \( H_i \), for all time, \( t \), and obey Kirchhoff's laws. For a standard electrical network, the space, \( H_i \), would be \( \mathbb{R} \), but for a rigid body the space, \( H_i \), would be given as \( \mathbb{R}^3 \times SO(3) \), and for a flexible beam the space, \( H_i \), might be given as \( L_2[0,1] \).

2.1 Network Elements

In this section the basic elements are defined. The basic elements are either one-ports or two-ports. Each of the one-port operators, \( A_i \), relate efforts, \( f_i \), and flows, \( v_i \), belonging to a single space, \( H_i \), and are represented by a single subscript. The two-port elements, \( A_{ij} \), on the other hand, relate efforts, \( f_i \) and \( f_j \), and flows, \( v_i \) and \( v_j \), belonging to two different spaces, \( H_i \)
and $H_j$, and thus require the slash notation. The one-port and two-port elements can be further divided into two general classes, passive elements and active elements.

2.1.1 The passive one-port elements

There are three basic one-port passive elements: the inertial element, $M$; the resistance (conductance) element, $B$ ($G$); and the stiffness (compliance) element, $K$ ($C$). The inertial element and the stiffness element are said to be dynamic elements, since they are associated with energy storage. The damping element, on the other hand, is memoryless and can only dissipate energy.

The inertial one-port, $M_i: H_i \rightarrow H_i$, is represented by an inductor (Fig. 2.1a), and has the constitutive relationship:

$$f_i = \frac{d}{dt}(M_i v_i).$$

(2.1)

It is assumed that the element is both positive definite and symmetric. The energy stored in $M_i$, i.e., kinetic energy, is given by

$$K.E. = \frac{1}{2} <v_i, M_i v_i>.$$  

(2.2)

![Diagram of passive circuit elements](image)

Fig. 2.1. Passive circuit elements: a) inertia; b) damping; c) stiffness; d) transformer; e) gyrator.

To insure that the energy entering $M_i$ minus the energy leaving $M_i$ is equal to the energy stored in $M_i$, the condition

$$<v_i, M_i v_i> = 0 \text{ almost everywhere (a.e.)}$$

(2.3)

must hold true for all $v_i \in H_i$. This condition is proved below.
Theorem 2.1: The equality,
\[ \frac{1}{2} \langle v_i(t), M_i v_i(t) \rangle_i - \frac{1}{2} \langle v_i(0), M_i v_i(0) \rangle_i = \int_0^t \langle f_i, v_i \rangle_i dt, \]
implies that \( \langle v_i, \dot{M}_i v_i \rangle_i = 0 \) a.e. for all \( v_i \in H_i \).

Proof

\[
\int_0^t \langle f_i, v_i \rangle_i dt = \int_0^t \frac{d}{dt} \langle M_i v_i, v_i \rangle_i dt \\
= \frac{1}{2} \int_0^t \langle \dot{M}_i v_i, v_i \rangle_i dt + \frac{1}{2} \int_0^t (\langle M_i v_i, v_i \rangle_i + \langle M_i \dot{v}_i, v_i \rangle_i) dt \\
= \frac{1}{2} \langle M_i v_i, v_i \rangle_i \delta + \frac{1}{2} \int_0^t \langle M_i v_i, v_i \rangle_i dt \\
= \frac{1}{2} \langle M_i v_i, v_i \rangle_i \delta + \frac{1}{2} \int_0^t \langle \dot{M}_i v_i, v_i \rangle_i dt. \quad (2.5)
\]

Thus, the difference between the net energy entering the system and the stored energy is given by

\[
\int_0^t \langle f_i, v_i \rangle_i dt - \frac{1}{2} \langle M_i v_i, v_i \rangle_i \delta = \frac{1}{2} \int_0^t \langle M_i v_i, v_i \rangle_i dt, \quad (2.6)
\]

and for this to be zero it follows that \( \langle v_i, \dot{M}_i v_i \rangle_i = 0 \) a.e. for all \( v_i \in H_i \). \( \square \)

It is assumed that all inertial elements satisfy the conservation of energy principle of the previous theorem. Thus, the inertial element, \( \dot{M}_i \), can be decomposed into the sum of a null-space operator, \( N_i \), and its transpose, so that

\[ \dot{M}_i = N_i + N_i^*, \quad (2.7) \]

where \( N_i v_i = 0 \), for all \( v_i \in H_i \). In general, \( N_i^* v_i \neq 0 \), and thus \( \dot{M}_i \) is nonzero. As an example of the null-space operator, consider an inertia term in \( \mathbb{R}^3 \) given by

\[ M_i = R_{jii}^T M_j R_{jii}, \]

where \( R_{jii} = e^{S(v_i)} \) is a rotation matrix. Taking the derivative of \( M_i \) gives

\[
\dot{M}_i = \dot{R}_{jii}^T M_j R_{jii} + R_{jii}^T M_j \dot{R}_{jii} \\
= R_{jii}^T M_j \dot{R}_{jii} S(v_i) + S(v_i)^T \dot{R}_{jii}^T M_j R_{jii} \\
= N_i + N_i^*. 
\]
From Proposition 1.1, it follows that $S(v_i)$ is a skew-symmetric function of $v_i$, and thus,
\[ N_i v_i = R_{jii}^T M_j R_{jii} S(v_i) v_i = 0. \]

The stiffness one-port element, $K_i: \mathbf{H}_i \rightarrow \mathbf{H}_i$, is represented by a capacitor (Fig. 2.1c), and has the constitutive relationship:
\[ f_i = K_i x_i, \quad (2.8) \]
where $x_i \triangleq \int_0^t v_i \, d\tau$.

The stiffness element is positive semidefinite and symmetric. The energy stored in the capacitive element, i.e., potential energy, is given by
\[ P.E. = \int_{x(0)}^{x(t)} \langle f_i, dx_i \rangle. \quad (2.9) \]
This is equal to the net energy applied across the capacitive one-port,
\[ \int_0^t \langle f_i, v_i \rangle \, d\tau, \quad (2.10) \]
as long as the stiffness is conservative, i.e., the same amount of energy is consumed no matter which path is taken. If the system is nonconservative (e.g., exhibiting plastic behavior), then it must be remodeled using both conservative stiffness elements and nonconservative damping elements.

The inverse relationship of stiffness is compliance. The compliance operator, $C_i: \mathbf{H}_i \rightarrow \mathbf{H}_i$, is also represented by a capacitor (Fig. 2.1c), and has the constitutive relationship:
\[ v_i = \frac{d}{dt} (C_i f_i). \quad (2.11) \]
If $C_i$ is constant and invertible, then $K_i = C_i^{-1}$, and $K_i^{-1} = C_i$. Thus, both representations can be considered equivalent. Situations will arise, however, when neither the stiffness element nor the compliance element is invertible, and only one representation can be used, or when time variations in the element make one representation preferable.

The final passive one-port element is the memoryless damping element, $B_i: \mathbf{H}_i \rightarrow \mathbf{H}_i$, which is represented by a resistor (Fig. 2.1b), and has the constitutive relationship:
\[ f_i = B_i v_i. \quad (2.12) \]
The damping element, $B_i$, like the stiffness element, is positive semidefinite and symmetric. Considerably more flexibility is allowed, however, with the functional variation of $B_i$, since $B_i$ need not satisfy any conservation of energy principles.

The inverse relationship of damping is conductance. The conductance operator, $G_i : \mathbf{H}_i \rightarrow \mathbf{H}_i$, is also represented by a resistor (Fig. 2.1b), and has the constitutive relationship:

$$v_i = G_i f_i.$$  \hfill (2.13)

As might be surmised, both representations exist only if the operators, $G_i$ and $B_i$, are invertible.

Later in this chapter, an additional damping element, $D_i$, is defined which has the same constituent equation and representation as $B_i$, but is not necessarily either positive definite or symmetric. This element characterizes the "fictitious" forces that arise when different reference frames are used. The conductance element is also used in this context.

2.1.2 The passive two-port elements

Both of the standard passive two-port elements, the transformer, $J_{jii}$, and the gyrator, $G_{jii}$, are memoryless elements. They allow energy to flow from a network in one Hilbert space to a network in another Hilbert space.

The ideal transformer, $J_{jii} : \mathbf{H}_i \rightarrow \mathbf{H}_j$, is represented by a box (Fig. 2.1d), and has the constitutive relationships:

$$f_i = J_{jii}^* f_j, \quad v_j = J_{jii} v_i.$$  \hfill (2.14)

By convention, the ideal transformer is drawn so that the input effort and flow, $f_i$ and $v_i$, are either below or on the left and the output effort and flow, $f_j$ and $v_j$, are on top or on the right of the transformer element. By the definition of the adjoint, the input energy equals output energy, since

$$\langle f_i, v_i \rangle_i = \langle J_{jii}^* f_j, v_i \rangle_i = \langle f_j, J_{jii} v_i \rangle_j = \langle f_j, v_j \rangle_j.$$  \hfill (2.15)

The gyrator, $G_{jii} : \mathbf{H}_i \rightarrow \mathbf{H}_j$, is represented by a box with notches (Fig. 2.1e), and has the constitutive relationships

$$v_i = G_{jii}^* f_j, \quad v_j = G_{jii} f_i.$$  \hfill (2.16)

The gyrator is drawn using the same conventions as the transformer, and, assuming its invertibility, satisfies the same energy condition.
2.1.3 The active elements

Besides the passive elements, four active one-port elements and an active two-port element are included.

The four active one-port elements are as follows: an independent or "desired" effort (force or voltage) source, \( f_{d_i} \in \mathbf{H}_i \); an independent or "desired" flow (current or velocity) source, \( w_i \in \mathbf{H}_i \); a dependent, or "controlled" effort source, \( f_{c_i} \in \mathbf{H}_i \); and a dependent or "controlled" flow source, \( v_i \in \mathbf{H}_i \). As the names suggest, the independent sources cannot functionally depend on other vectors in the network, whereas the dependent sources are directly dependent on other vectors. Circuit representations for these elements are given in Fig. 2.2.

![Diagram of active circuit elements](image)

Fig. 2.2. Active circuit elements: a) independent effort source; b) independent flow; c) dependent effort source; d) dependent flow source.

For robots the controlled sources, \( v_i \) and \( f_{c_i} \), represent the controlled actuators. The choice of the control laws for \( v_i \) and \( f_{c_i} \) represents the control problem for robotics. The dependent sources, \( w_i \) and \( f_{d_i} \), on the other hand, represent desired trajectories or disturbances. A typical control problem for robotics is to determine, \( f_{c_i} \), so that the system’s flow, \( v_i \), tracks the desired flow, \( w_i \).

The two-port amplifier, \( A_i: \mathbf{H}_i \rightarrow \mathbf{H}_i \), is shown in Fig. 2.3, and is defined by

\[
\begin{align*}
    f_{2_i} &= A_i f_{1_i}, \\
    v_{2_i} &= A_i v_{1_i}.
\end{align*}
\]

(2.17)

The two-port amplifier could be represented with two one-port dependent sources, but it is interesting in its own right. It is used for both teleoperators and computed torque algorithms. The amplifier element is an active element, but when used properly, will not disrupt the passivity of the network. Unfortunately when it is misused, the passivity is disrupted, and instabilities can result.

Using the power gain element in conjunction with two transformers, any arbitrary positive scaling of forces and velocities is achieved. To show this, consider two symmetric, positive
definite operators, $B_i : H_i \rightarrow H_i$ and $C_i : H_i \rightarrow H_i$, and define the mappings, $v_{2i} = B_i v_{1i}$ and $f_{1i} = C_i f_{2i}$. Because $B_i$ and $C_i$ are positive definite operators, their positive definite square roots exist, and the mappings are rewritten as,

$$v_{2i} = B_i^{1/2} B_i^{-1/2} C_i^{-1/2} C_i^{1/2} v_{1i}$$  \hspace{1cm} (2.18)

$$f_{1i} = C_i^{1/2} C_i^{-1/2} B_i^{-1/2} B_i^{1/2} f_{2i}.$$  \hspace{1cm} (2.19)

Equations (2.18)-(2.19) can now be represented by the network shown in Fig. 2.4, where the gain element has the value, $A_i = B_i^{1/2} C_i^{-1/2}$, and the two transformers have the values, $J_{1i} = C_i^{1/2}$ and $J_{2i} = B_i^{1/2}$, respectively.

Note that by setting $B_i = I_i$ or $C_i = I_i$, the special cases of effort scaling ($f_{2i} = C_i f_{1i}$, $v_{2i} = v_{1i}$) or flow scaling ($f_{2i} = f_{1i}$, $v_{2i} = B_i v_{1i}$) are achieved.
2.1.4 Generic ports, impedances, admittances, and hybrid forms

In many cases, a group of passive one-port elements in a space, \( \mathbf{H}_i \), are lumped together into a generic one-port element with either a well-defined impedance operator, \( Z_i : \mathbf{H}_i \rightarrow \mathbf{H}_i \), given by \( f_i = Z_i v_i \), or a well-defined admittance operator, \( Y_i : \mathbf{H}_i \rightarrow \mathbf{H}_i \), given by \( v_i = Y_i f_i \). In either case, the resistor diagram represents the element. If, on the other hand, a well-defined impedance or admittance operator is not known, or cannot be found, then the generic one-port network, \( N_{i,..} \) of Fig. 2.5a, represents the one-port network.

![Fig. 2.5. Generic networks: a) one-port; b) two-port.](image)

For a two-port network with left side signals in the space, \( \mathbf{H}_i \), and right side signals in the space, \( \mathbf{H}_j \), the analysis is a bit more complicated. Although considering either both flows or both efforts as inputs defines an impedance operator, \( Z_{i\times j} \), or an admittance operator, \( Y_{i\times j} \), it is more natural to consider the left side flow, \( v_i \), and the right side effort, \( f_j \), to be the inputs for the system. In this case a mixed form, or hybrid operator, \( H_{i\times j} : \mathbf{H}_i \times \mathbf{H}_j \rightarrow \mathbf{H}_j \times \mathbf{H}_i \), is needed. The hybrid operator for the two-port is defined as

\[
\begin{bmatrix}
  f_i \\
  -v_j
\end{bmatrix} =
\begin{bmatrix}
  H_i & H_{iij} \\
  H_{jii} & H_j
\end{bmatrix}
\begin{bmatrix}
  v_i \\
  f_j
\end{bmatrix}.
\]

\[
= H_{i\times j}
\begin{bmatrix}
  v_i \\
  f_j
\end{bmatrix}.
\]

(2.20)

The hybrid representation of the transformer two-port, for example, is given by

\[
H_{i\times j} =
\begin{bmatrix}
  H_i & H_{iij} \\
  H_{jii} & H_j
\end{bmatrix} =
\begin{bmatrix}
  0_i & J_{ji}^* \\
  -J_{ij} & 0_j
\end{bmatrix}.
\]

As in the one-port case, there are many times when either a well-defined operator representation for the two-port is not known or does not exist in either the impedance, admittance, or hybrid forms. In this case, the generic representation, \( N_{j\times i} \), shown in Fig. 2.5b, is used.
Ports with more than two inputs, or n-ports, are no more difficult to represent than the two-port, since only two input signals, either effort or flow, and only two outputs, flow or effort, are possible. By defining the spaces, $\mathbf{H}_i$ and $\mathbf{H}_j$, as the Cartesian products of all the flow and effort inputs respectively, the n-port can always be reduced to the two-port form.

The admittance, impedance, and hybrid operators are especially useful when the inputs and outputs are Laplace transformable, and the basic elements are linear and time-invariant. In this case, the operators themselves are represented in the frequency domain, and the operations consist of simple matrix multiplications. The Laplace domain equivalents of the operators, $Y_i$, $Z_i$, and $H_i$, are given respectively as $Y_i(s)$, $Z_i(s)$, and $H_i(s)$.

2.2 Network Principles

In this section a number of useful theorems used for standard networks are introduced and extended to the Hilbert network.

2.2.1 Tellegen's Theorem and passivity

Tellegen's Theorem is well known in standard circuit theory, being directly derivable from Kirchoff's laws. It states that the sum over each of the $b$ branches in a network and the product of effort and flow for each branch is equal to zero, i.e.,

$$\sum_{i=1}^{b} f_i v_i = 0.$$ 

This result is directly extendable to the Hilbert network. The approach is similar to the proof in (Anderson 1973).

Theorem 2.2: (Tellegen's Theorem for Hilbert networks)

Suppose that $N$ is a Hilbert network with $b$ branches and $n$ nodes. For the $i$th branch of the graph, suppose that $f_i \in \mathbf{H}_i$ is the branch effort under one set of operating conditions at any one instant of time, and $v_i \in \mathbf{H}_i$ is the branch flow under any other set of operating conditions at any other instant of time, with the standard sign conventions for $f_i$ and $v_i$. Then,

$$\sum_{i=1}^{b} <f_i, v_i>_i = 0.$$ 

Proof The network $N$ is partitioned, as is shown in Fig. 2.6, into $n_{bi}$ partitions, so that each partition, $N_i$, contains $b_i$ elements from the same Hilbert space, $\mathbf{H}_i$. The partitions are connected together by two-port transformers and gyrators. Replacing the two-port
transformers and gyrators by dependent sources results in a network containing \( n_{bi} \) disconnected subnetworks. Applying the standard form of Tellegen's theorem to each of the subnetworks gives

\[
\sum_{i=1}^{n_{bi}} \langle f_i, v_i \rangle_i = \sum_{i=1}^{n_{bi}} \sum_{j=1}^{n_{bj}} \langle f_j, v_j \rangle_j = \sum_{i=1}^{n_{bi}} 0 = 0. \quad \square
\]

Passivity is one of the most important characteristics of systems which a network can represent. A passive system cannot create energy and thus, from a control point of view, cannot go unstable. Consider an n-port with \( n \) entering flows, \( (v_{i1}, \ldots, v_{in}) \), and \( n \) port efforts, \( (f_{i1}, \ldots, f_{in}) \), where each of the individual flows, \( v_{ij} \), and efforts, \( f_{ij} \), belong to the space, \( H_{ij}(t) \). Define the composite space, \( H_i(t) \triangleq \bigoplus_{j=1}^{n_{bi}} H_{ji}(t) \), and set

\[
v_i(t) = (v_{i1}(t), \ldots, v_{in}(t)) \in H_i(t),
\]

and,

\[
f_i(t) = (f_{i1}(t), \ldots, f_{in}(t)) \in H_i(t).
\]

Now passive and lossless n-ports can be defined.

**Definition 2.1:** An n-port is said to be passive if and only if for any independent set of \( n \) port flows, \( v_i = (v_{i1}(t), \ldots, v_{in}(t)) \), injected into the system, and efforts, \( f_i = (f_{i1}(t), \ldots, f_{in}(t)) \), applied across the system, there exists \( \gamma_i \in \mathbb{R} \), such that

\[
\int_0^t \langle f_i, v_i \rangle_i \, dt = \int_0^t \sum_{j=1}^{n_{bj}} \langle f_{ji}, v_{ji} \rangle_{ji} \, dt \geq -\gamma_i^2 \quad \forall t \in [0, \infty).
\]
Definition 2.2: An n-port is said to be lossless if and only if for any independent set of n port flows, \( v_i = (v_{i1}(t), \cdots, v_{in}(t)) \), injected into the system, and efforts, \( f_i = (f_{i1}(t), \cdots, f_{in}(t)) \), applied across the system,
\[
\int_0^t \langle f_i, v_i \rangle_i \, dt = \int_0^t \sum_{j=1}^n \langle f_{ji}, v_{ji} \rangle_{ji} \, dt = 0.
\]

From these definitions, all of the so-called passive elements are indeed passive, and, in addition, the transformer and gyrator elements are lossless.

Theorem 2.3: A Hilbert network port, \( N_i \), consisting solely of passive elements is passive.

Proof Let the input flow for the network be given by \( v_i \), and let the applied effort be given by \( f_i \). Because each element in the network is passive, it follows that for each branch, \( j \), there exists \( \gamma_j \), such that,
\[
\int_0^t \langle f_j, v_j \rangle_j \, dt \geq -\gamma_j^2.
\]
From Tellegen's theorem,
\[
\int_0^t \langle f_i, v_i \rangle_i \, dt = \int_0^t \sum_{j=1}^n \langle f_{ji}, v_{ji} \rangle_{ji} \, dt \geq \sum_{j=0}^b -\gamma_j^2 \Delta - \gamma_j^2. \quad \Box
\]

Although circuits made up of passive elements are themselves passive and linear, as shown in Section 2.2.3, time-invariant causal systems which exhibit a passive input/output behavior are realized solely by a circuit made up of passive elements. This does not imply that the circuit must be constructed using passive elements. In fact in robot control, dependent sources, namely dc-motors, imitate the input/output behavior of a passive system. This leads to the following definition:

Definition 2.3: A controlled source, \( f_c \) or \( w_c \), is controlled passively if it can be replaced with passive elements which maintain the same input/output behavior.

From this definition and Theorem 2.3, it follows that a port consisting solely of passive elements and passively controlled sources is passive.

In addition to passive elements and controlled sources, many systems contain independent sources. Intuitively, if the system is passive, assuming that the independent sources provide only a finite amount of energy, then their inclusion cannot cause instability. To show this, suppose that a one-port with dependent sources, \( N_i \), is converted to a three-port, \( N_i \times j(x_k) \), by pulling out the independent sources, \( f_d j \) and \( w_k \), as is shown in Fig. 2.7.
Let the energy provided by independent flow and effort sources be given by
\[ \gamma_i^2 \triangleq \int_0^t <v_d, f_j>, \]
and,
\[ \gamma_k^2 \triangleq \int_0^t <v_k, f_d k>, \]
respectively. Checking the passivity condition for the port, \( N_i \), gives
\[
\int_0^t <f_i, v_i>_i = \int_0^t <f_i, v_i>_i + <f_j, v_d>_j + <f_k, v_k>_k - \int_0^t [<f_j, w_d>_j + <f_k, v_k>_k] dt \\
\geq -\gamma_{jk}^2 - \gamma_j^2 - \gamma_k^2 \\
\triangleq -\gamma_i^2,
\]
where \( \gamma_{jk} \) exists because \( N_{i \times j \times k} \) is passive. If \( \gamma_j \) and \( \gamma_k \) are bounded, then the system is passive. Because the amount of energy provided by the dependent sources is not bounded in general, no lower bound, \( \gamma_i \), exists for \( N_i \), and the system is not passive. Thus, the following definition is required:

**Definition 2.4:** A port with independent sources is **passive** if the port obtained from pulling all independent sources from the system is passive.

When describing arbitrary networks, it is useful to combine individual signals and elements having similar characteristics into larger **composite** signals and elements. Let \( N \) represent an arbitrary passive Hilbert network. Assume that the network contains \( n_k \) stiffness elements, \( n_m \) inertial elements, and \( n_b \) damping elements, and let the spaces for these elements be given by \( H_{ki} \) \((i = 1, \cdots, n_k)\), \( H_{m_i} \) \((i = 1, \cdots, n_m)\), and \( H_{bi} \) \((i = 1, \cdots, n_b)\), respectively. The
composite stiffness, damping, and inertial vectors can then be defined for the system by first defining the composite spaces, \( H_k \triangleq H_{k1} \times \cdots \times H_{kn} \), \( H_m \triangleq H_{m1} \times \cdots \times H_{mn} \), and \( H_b \triangleq H_{b1} \times \cdots \times H_{bn} \), then setting the respective composite efforts, \( f_k \), \( f_m \), and \( f_b \), and the composite flows, \( v_k \), \( v_m \), and \( v_b \), to the Cartesian product of the individual signals, e.g., \( v_k \triangleq (v_{k1}, \ldots, v_{kn}) \).

The following lemma is used in proofs regarding the stability of both the force control system and the teleoperator system.

**Lemma 2.1:** Let, \( N_i \) be a passive Hilbert network port with input flow, \( v_i \in H_i \), and output effort, \( f_i \in H_i \), and let \( v_k \), \( v_b \), and \( v_m \) be the composite stiffness, damping and inertial velocity vectors, and let \( f_k \), \( f_b \), and \( f_m \) be the composite stiffness, damping, and inertial effort vectors for \( N_i \). Then

\[
<v_i, f_i>_i \geq <v_k, f_k>_k + <v_m, f_m>_m.
\]

**Proof** Assume that the network contains \( n_g \) gyrators and \( n_j \) transformers, and that each of these two-port elements are replaced by two dependent sources. The total number of elements in \( N_i \) is then given by

\[
b \triangleq n_k + n_m + n_b + 2n_j + 2n_g.
\]

and thus by Tellegen's Theorem, it follows that:

\[
<v_i, f_i>_i = \sum_{j=1}^{b} <v_j, f_j>_j
\]

\[
= \sum_{i=1}^{n_k} <v_{ki}, f_{ki}>_{ki} + \sum_{i=1}^{n_m} <v_{mi}, f_{mi}>_{mi} + \sum_{i=1}^{n_b} <v_{bi}, f_{bi}>_{bi}
\]

\[
+ \sum_{i=1}^{n_g} <v_{g1i}, f_{g1i}>_{g1i} + \sum_{i=1}^{n_g} <v_{g2i}, f_{g2i}>_{g2i}
\]

\[
+ \sum_{i=1}^{n_j} <v_{j1i}, f_{j1i}>_{j1i} + \sum_{i=1}^{n_j} <v_{j2i}, f_{j2i}>_{j2i}.
\]

Because transformers and gyrators are lossless, the second line is identically zero. Thus, by using the definitions for composite elements, it follows that:

\[
<v_i, f_i>_i = <v_k, f_k>_k + <v_m, f_m>_m + <v_b, f_b>_b.
\]

Since all damping/conductance elements are positive semidefinite by assumption, \( <v_b, f_b>_b \geq 0 \) \( \forall t \in [0, \infty) \). Therefore,

\[
<v_i, f_i>_i \geq <v_k, f_k>_k + <v_m, f_m>_m.
\]
2.2.2 Scattering theory

Another useful tool for networks, especially for infinite dimensional systems such as transmission lines and delay systems, is scattering theory (Wohlers 1969). The scattering operator, \( S_i : H_i \rightarrow H_i \) is defined by

\[
f_i - Z_i v_i = S_i (f_i + Z_i v_i),
\]

and maps effort plus scaled flow into effort minus scaled flow. The impedance, \( Z_i \), is a constant scaling factor and is called the characteristic impedance.

For LTI systems, the scattering operator, \( S_i \), is expressed in the frequency domain as \( S_i(s) \), where

\[
f_i(s) - Z_i v_i(s) = S_i(s)(f_i(s) + Z_i v_i(s)).
\]

In the case of a two-port, this scattering matrix relates to the hybrid matrix, \( H_{i \times j}(s) \), as follows:

\[
\begin{bmatrix}
  f_i(s) - Z_i v_i(s) \\
  f_j(s) + Z_j v_j(s)
\end{bmatrix} = \begin{bmatrix}
  I_i & 0_{i j} \\
  0_{j i} & -Z_j
\end{bmatrix} \begin{bmatrix}
  f_i(s) \\
  v_i(s)
\end{bmatrix} + \begin{bmatrix}
  -Z_i & 0_{i j} \\
  0_{j i} & I_j
\end{bmatrix} \begin{bmatrix}
  f_j(s) \\
  v_i(s)
\end{bmatrix}.
\]

Likewise,

\[
\begin{bmatrix}
  f_i(s) + Z_i v_i(s) \\
  f_j(s) - Z_j v_j(s)
\end{bmatrix} = \begin{bmatrix}
  I_i & 0_{i j} \\
  0_{j i} & -Z_j
\end{bmatrix} H_{i \times j}(s) + \begin{bmatrix}
  Z_i & 0_{i j} \\
  0_{j i} & I_j
\end{bmatrix} \begin{bmatrix}
  f_i(s) \\
  f_j(s)
\end{bmatrix}.
\]

Therefore,

\[
S_{i \times j}(s) = \begin{bmatrix}
  I_i & 0_{i j} \\
  0_{j i} & -Z_j
\end{bmatrix} H_{i \times j} - \begin{bmatrix}
  Z_i & 0_{i j} \\
  0_{j i} & Z_j^{-1}
\end{bmatrix} \begin{bmatrix}
  H_{i \times j} + \begin{bmatrix}
  Z_i & 0_{i j} \\
  0_{j i} & Z_j^{-1}
\end{bmatrix}^{-1}
\end{bmatrix}.
\]

**Theorem 2.4**: An n-port is passive if and only if the time norm of its scattering operator is less than or equal to one.

**Proof**: (sufficiency) If \( || S_i || \leq 1 \), then \( || f_i - Z_i v_i || \leq || f_i + Z_i v_i || \), for all \( f_i, v_i \in H_i(t) \). This implies that \( || f_i + Z_i v_i ||^2 - || f_i - Z_i v_i ||^2 \geq 0 \). Writing out the norms in terms of inner products gives

\[
\int_0^\infty < f_i + Z_i v_i, f_i + Z_i v_i > dt = < f_i - Z_i v_i, f_i - Z_i v_i > dt \geq 0,
\]

but this is equivalent to
\[ 2 \int_0^\infty \langle f_i, Z_i v_i \rangle_i + \langle Z_i v_i, f_i \rangle_i \, dt = 4 \int_0^\infty \langle f_i, Z_i v_i \rangle_i \, dt \geq 0. \]

Reversing the argument will show necessity. □

**Corollary 2.1** A system is passive if and only if \( \sup_{\omega} \lambda^{\frac{1}{2}}(S^T(j\omega)S(j\omega)) \leq 1. \)

**Proof** Follows directly from (3.4) and Thm. 3.1.

For a transmission line, the scattering operator relates the reflected wave \((f_i - Z_i v_i)\) to the incident wave \((f_i + Z_i v_i)\). Its norm is interpreted as the square root of the maximum power gain for the element. Thus, it follows that for a passive element \( ||S_i||_1 \leq 1. \)

### 2.2.3 Realization for passive LTI systems

In this subsection, it is shown how positive real transfer functions are realized in Hilbert network form. Consider a system with \( m \) inputs, \( m \) outputs, and \( n \) states. Define the input/output space, \( H_m \triangleq \mathbb{R}^m \), and the state space, \( H_n \triangleq \mathbb{R}^n \), and let \( Z_m(s) \) represent a transfer function which relates the output vector, \( f_m \in H_m \), to the input vector, \( v_m \in H_m \). If \( Z_m(\infty) < \infty \), then it is possible to represent the transfer function matrix in the following state space form,

\[
Z_m(s) = D_m + C_{m/n} (sI_n - A_n)^{-1} B_{n/m},
\]

where \( A_n \) is the plant matrix, \( B_{n/m} \) is the input matrix, \( C_{m/n} \) is the output matrix, and \( D_m \) is the feedforward matrix.

**Definition 2.5:** A transfer function, \( Z_m(s) \), is **positive real**, if the following conditions are satisfied:

1. All elements of \( Z_m(\cdot) \) are analytic in \( \text{Re}[s] > 0 \).
2. \( Z_m(s) \) is real for real positive \( s \).
3. \( Z_m^*(s) + Z_m(s) \geq 0 \) for \( \text{Re}[s] > 0 \).

**Theorem 2.5:** An LTI system is passive if and only if both its impedance and admittance operators are positive real.

**Proof** (see Anderson 1973)

An LTI impedance function is realized in terms of a passive network. Consider a finite dimensional, linear, time-invariant impedance, represented by the transfer function, \( Z_m(s) \).

**Definition 2.6:** A realization \( Z_m(s) = D_m + C_{m/n}^T (sI_n - A_n)^{-1} B_{n/m} \) is a **passive realization** if the matrix,
\[
\begin{bmatrix}
D_m & -C_{m/n} \\
B_{n/m} & -A_n
\end{bmatrix},
\]
is nonnegative definite.

**Lemma 2.2:** (Positive Real Lemma)

Let \( Z_m(s) \) be an \( m \times m \) matrix of real rational functions of a complex variable, \( s \), with \( Z(\infty) < \infty \). Let \( (A_n, B_{n/m}, C_{m/n}, D_m) \) be a minimal realization of \( Z_m(s) \). Then \( Z_m(s) \) is positive real if and only if there exist real matrices, \( P_n, L_{nlp}, \) and \( W_{p/im} \), with \( P_n \) positive definite symmetric, such that

\[
P_n A_n + A_n^T P_n = -L_{nlp} L_{nlp}^T \\
P_n B_{n/m} = C_{m/n}^T - L_{nlp} W_{p/im} \\
W_{p/im}^T W_{p/im} = D_m + D_m^T.
\]

**Proof** (see Anderson 1973).

**Theorem 2.6:** Let \( Z_m \) be an \( m \times m \) positive real matrix of real rational functions of a complex variable, \( s \), with \( Z_m(\infty) < \infty \). Then there exists a passive realization of \( Z_m(s) \).

**Proof** Let \( (\hat{A}_n, \hat{B}_{n/m}, \hat{C}_{m/n}, \hat{D}_m) \) be a minimal realization of \( Z_m(s) \). Let \( P_n, L_{nlp} \), and \( W_{p/im} \) be matrices which satisfy the positive real lemma. Then

\[
(A_n, B_{n/m}, C_{m/n}, D_m) \Delta (P_n^{\frac{1}{2}} \hat{A}_n P_n^{-\frac{1}{2}}, P_n^{\frac{1}{2}} \hat{B}_{n/m}, \hat{C}_{m/n} P_n^{-\frac{1}{2}}, \hat{D}_m)
\]
is a passive realization, since

\[
\begin{bmatrix}
x_m^T \\
x_n^T
\end{bmatrix}
\begin{bmatrix}
D_m & -C_{m/n} \\
B_{n/m} & -A_n
\end{bmatrix}
\begin{bmatrix}
x_m \\
x_n
\end{bmatrix}
= \frac{1}{2}
\begin{bmatrix}
x_m^T \\
x_n^T
\end{bmatrix}
\begin{bmatrix}
D_m + D_m^T & -C_{m/n} + B_{n/m}^T \\
B_{n/m} - C_{m/n}^T & -A_n - A_n^T
\end{bmatrix}
\begin{bmatrix}
x_m \\
x_n
\end{bmatrix}
= \begin{bmatrix}
x_m^T \\
x_n^T
\end{bmatrix}
\begin{bmatrix}
W_{p/im} W_{p/im} & W_{p/im} L_{nlp} L_{nlp}^T P_n^{-\frac{1}{2}} \\
-P_n^{-\frac{1}{2}} L_{nlp} W_{p/im} & P_n^{-\frac{1}{2}} L_{nlp} L_{nlp}^T P_n^{-\frac{1}{2}}
\end{bmatrix}
\begin{bmatrix}
x_m \\
x_n
\end{bmatrix},
\]

which is nonnegative for all \( x_m \in H_m, x_n \in H_n \).

Given the previous theorems, it is now possible to create a network realization of \( Z_m(s) \). Consider the network shown in Fig. 2.8, in which the port, \( N_{m \times n} \), contains only the nondynamic elements, namely transformers, resistors, and gyrators, and is connected to an identity valued stiffness element in \( H_n \), so that \( v_n(s) = s f_n(s) \). By setting the hybrid matrix representation of \( N_{m \times n} \) to the passive realization matrix, so that
\[
\begin{bmatrix}
  f_m \\
  -v_n
\end{bmatrix} = \begin{bmatrix}
  D_m & -C_{m/n} \\
  B_{n/m} & -A_n
\end{bmatrix} \begin{bmatrix}
  v_m \\
  f_n
\end{bmatrix} \triangleq H_{m \times n} \begin{bmatrix}
  v_m \\
  f_n
\end{bmatrix},
\]

\(Z_m(s)\) is realized. The system transfer function is derived by solving for the state vector, \(x_n \in \mathbb{H}_n\), as is shown below. Since the stiffness element is identity valued, it follows that

\[v_n(s) = sx_n(s) = sf_n(s)\].

From the hybrid representation, however,

\[v_n(s) = -B_{n/m} v_m + A_n f_n\],

and thus,

\[f_n(s) = x_n(s) = -(sI_n - A_n)^{-1} B_{n/m} v_m\].

Substituting this in the hybrid representation again gives

\[f_m = D_m v_m + C_{m/n} (sI_n - A_n)^{-1} B_{n/m} v_m = Z_m(s) v_m\].

The hybrid matrix, \(H_{m \times n}\), is positive semidefinite by construction. It can be decomposed into a symmetric and skew-symmetric parts. As is shown in (Anderson 1973), the symmetric portion is constructed using a single transformer connected to a unity valued resistor, while the skew-symmetric portion is constructed using only gyrators and transformers.

### 2.3 Network Reduction Rules

Network reduction rules for scalar LTI circuits are well known, and consist of combining frequency domain impedances of parallel and series passive elements into an equivalent impedance. When elements belong to many different vector spaces, however, additional network reduction operations exist. Care must be taken so that nonlinear and time-varying terms are transformed correctly.
In this section a number of operations involving the two-port elements are introduced. The most basic of operations is moving a one-port element from one side of a two-port element to the other, or "pulling" the element through the two-port. For time-varying, nonlinear two-ports this operation requires the introduction of "fictitious" elements. Next, a number of operations involving cascade, series and parallel network structures are shown. Finally, a ladder network with $2n$ elements is reduced to a network with only two elements.

2.3.1 Pulling impedances through transformers and gyrators

First, consider the effect of an impedance as seen through a transformer. Fig. 2.9 shows a one-port element, $Z_j: H_j \rightarrow H_j$, viewed through a constant valued two-port transformer, $J_{ji}: H_i \rightarrow H_j$. The equivalent impedance is calculated from the constituent equation for the transformer,

$$f_j = J_{ji}^* f_j = J_{ji}^* Z_j v_j = J_{ji}^* Z_j J_{ji} v_i.$$  \hspace{1cm} (2.25)

![Diagram of transformer and equivalent impedance](image)

Fig. 2.9. Equivalent impedance as seen through a transformer: a) network with transformer; b) network with equivalent impedance.

The equivalent impedance, $Z_i: H_i \rightarrow H_i$ is given by $Z_i \triangleq J_{ji}^* Z_j J_{ji}$. In general, any one-port element can be moved from the right side of a constant valued transformer to the left side by scaling the appropriate operator in this fashion. This operation preserves the positive semidefiniteness of the operator, but in cases where $\text{rank}(J_{ji}) < i$, it does not preserve positive definiteness. If $\text{dim}(H_i) = \text{dim}(H_j)$ and $J_{ji}$ is of full rank, then its inverse, $J_{ji}^{-1}$, exists, and an element with impedance, $Z_i$, on the left of the transformer can be moved to the right of the transformer with the equivalent impedance $Z_i \triangleq J_{ji}^* Z_i J_{ji}^{-1}$.

If the inverse of $J_{ji}$ does not exist, but an admittance representation, $Z_i^{-1} \triangleq Y_i: H_i \rightarrow H_i$, does exist, then the element may be pulled to the right with the equivalent admittance, $Y_j \triangleq J_{ji} Y_i J_{ji}^*$, since
\( v_j = J_{ji} v_i = J_{ji} Y_i J_{ji} f_i = J_{ji} Y_i J_{ji} f_j. \) \hspace{1cm} (2.26)

A similar result occurs when a gyrorator is cascaded with an impedance. Consider the network shown in Fig. 2.10. The constituent equation for the gyrorator gives

\[ v_i = G_{ji} f_j = G_{ji} Z_j v_j = G_{ji} Z_j G_{ji} f_i \triangleq Y_i f_i, \] \hspace{1cm} (2.27)

where the equivalent admittance is given by \( Y_i \triangleq G_{ji} Z_j G_{ji} \). If the gyrorator is identity valued, then \( Y_i = Z_j \). As this suggests, the gyrorator "dualizes" a network. As viewed from one side of the gyrorator, impedances appear as admittances, parallel circuits appear as series circuits, effort signals become flow signals, and vice versa. Thus, it makes little difference if one relates either voltage to velocity or voltage to force in an electrical/mechanical analog.1

![Fig. 2.10. Equivalent admittance as seen through a gyrorator: a) network with gyrorator; b) network with equivalent admittance.](image)

If elements on the input port of a gyrorator are connected to elements on the output port of the gyrorator, then nonreciprocal impedances are possible, i.e., impedances which are nonsymmetric. Although this is an important characteristic for realization theory, it does not often occur when modeling mechanical systems, and thus the gyrorator plays a minor role in this work.

Returning to the transformer network, consider a transformer, \( J_{ji} \), which is null-valued. In this case the constituent equations give \( v_j = 0_{ji} v_i = 0 \) and \( f_i = 0_{ij} f_j = 0 \). This implies that the zero-valued transformer appears as a short circuit to the input port and as an open circuit to the output port. This is shown in Fig. 2.11.

---

1 This is true in the case of planar networks. Nonplanar networks cannot be dualized, and thus in these cases the analog used is important. It has been pointed out (Breedveld 1985) that voltage should be related to current in these cases. Unfortunately, this does not coincide with the accepted effort-flow convention used here. Nevertheless, because nonplanar networks are not encountered in this work, the point is moot.
Fig. 2.11. Zero-valued Jacobian: a) network with transformer; b) network with equivalent open/short circuits.

When elements are pulled through a Jacobian which is either directly time-varying or is a function of a variable which evolves over time, then great care must be taken to ensure that the resulting system maintains the same defining equations as the original system. In the case of inertial and capacitive elements, additional "fictitious" damping elements need to be added to the new network.

Fig. 2.12. Pulling an inertia through a time-varying transformer: a) before; b) after.

Consider the inertial term shown in Fig. 2.12a. The two-port equations for the network are given by

\[ v_j = J_{jii}(t)v_i; \quad f_i = J_{ji*}(t)f_j + J_{ji*}(t) \frac{d}{dt}(M_j(t)v_j). \]

This is equivalent to the network shown in Fig. 2.12b, where the transformed inertia, \( M_i : \mathbb{H}_i \rightarrow \mathbb{H}_i \), is given by

\[ M_i = J_{ji*}^* M_j J_{ji*}, \tag{2.28} \]

and the additional corrective damping term, \( D_i : \mathbb{H}_i \rightarrow \mathbb{H}_i \), defined by

\[ D_i = -J_{ji*}^* M_j J_{ji*}, \tag{2.29} \]
has been added in series with \( M_i \). Computing the force equation for the transformed network verifies this:

\[
f_i = J_{ji} f_j + J_{ji} \left( \frac{d}{dt} (M_i(t) v_i) + D_i(t) v_i \right)
\]

\[
= J_{ji} f_j + \frac{d}{dt} \left( J_{ji} M_j J_{ji} v_i \right) - J_{ji} M_j J_{ji} v_i
\]

\[
= J_{ji} f_j + J_{ji} \frac{d}{dt} (M_j(t) v_j).
\]

In general, three different terms are associated with a given nonlinear inertia term: the inertia term, \( M_i \); the associated corrective damping term, \( D_i \); and the null space operator, \( N_i \), Eq. (2.7) which ensures that conservation of energy is maintained for the element. From these three terms, a fourth term, the coriolis and centripetal force operator, \( C_i \), is defined as

\[
C_i = N_i^* - D_i^*
= \dot{M}_i + D_i - N_i. \tag{2.30}
\]

It should be clear from context, when the compliance operator, which is also represented by a capital \( C \), and the coriolis and centripetal force operator are being referenced.

The coriolis and centripetal force term is used in the standard form of the robot equations, which is derived below:

\[
f_{ci} = \frac{d}{dt} (M_i v_i) + D_i v_i
\]

\[
= \dot{M}_i v_i + M_i \dot{v}_i + D_i v_i
\]

\[
= M_i \dot{v}_i + (C_i + N_i - D_i) v_i + D_i v_i
\]

\[
= M_i \dot{v}_i + C_i v_i. \tag{2.31}
\]

As defined, the coriolis and centripetal force operator, \( C_i \), gives rise to the following identity:

\[
\dot{M_i} = C_i + C_i^*. \tag{2.32}
\]

In other words, \( \frac{1}{2} \dot{M_i} - C_i \) is skew-symmetric.

When an inertia term, \( M_j \), whose derivative satisfies the identity,

\[
\dot{M}_j = N_j - D_j + N_j^* - D_j^*
= C_j^* + C_j,
\]

and its associated fictitious damping term, \( D_j \), are pulled through a transformer with Jacobian, \( J_{ji} \), then the resulting terms in the space, \( H_i \), are
\[ M_i = J_{ji}^* M_j J_{ji} \]  
\[ N_i = J_{ji}^* N_j J_{ji} \]  
\[ D_i = J_{ji}^* D_j J_{ji} - \dot{J}_{ji}^* M_j J_{ji} \]  
\[ C_i = N_i^* - D_i^* = J_{ji}^* C_j J_{ji} + J_{ji}^* M_j \dot{J}_{ji} \]  

Thus,
\[
\dot{M}_i = \frac{d}{dt}(J_{ji}^* M_j J_{ji})
\]
\[ = \dot{J}_{ji}^* M_j J_{ji} + J_{ji}^* (N_j - D_j + N_j^* - D_j^*) J_{ji} + J_{ji}^* M_j \dot{J}_{ji}\]
\[ = N_i + N_i^* - D_i - D_i^* \]
\[ = C_i + C_i^*, \]

as expected.

A similar approach is applied when a compliance term, \( C_i \), is pulled through a two-port, except that a corrective conductance term, \( G_j \), appears in parallel, rather than in series with the transformed compliance element. This is shown in Fig. 2.13., where the corrective conductance term is given by \( G_j = -\dot{J}_{ji}^* C_i J_{ji}^* \).

\[ v_j = J_{ji}^* v_i - \frac{d}{dt}(C_i J_{ji}^* f_j) \]
\[ = J_{ji}^* v_i - \frac{d}{dt}(J_{ji}^* C_i J_{ji}^* f_j) + \dot{J}_{ji}^* C_i J_{ji}^* f_j \]
\[ = J_{ji}^* v_i - \frac{d}{dt}(C_j f_j) - G_j f_j. \]
2.3.2 Cascade, series, and parallel network reductions

The reductions in this section may be done on any network, whether the elements are time-varying or not. First, consider the cascade combination of two transformers, as is shown in Fig. 2.14a. The two transformers, \( J_{ji} : H_i \to H_j \) and \( J_{kli} : H_j \to H_k \), can be combined to get the equivalent transformer,

\[
J_{kli} \triangleq J_{kij} J_{ji} ,
\]

shown in Fig. 2.14b, since

\[
v_k = J_{kij} v_j = J_{kij} J_{ji} v_i = J_{kii} v_i ,
\]

and

\[
f_i = J^*_{ji} f_j = J^*_{ji} J^*_{kli} f_k = J^*_{kli} f_k .
\]

![Diagram](image)

Fig. 2.14. Transformers in cascade: a) two transformers in cascade; b) equivalent single transformer representation.

Additional simplifications exist when new spaces are introduced. Consider the network shown in Fig. 2.15a, where \( J_{ji} : H_{i1} \to H_j \) and \( J_{ji2} : H_{i2} \to H_j \) are in parallel. By defining a new space, \( H_i \triangleq H_{i1} \times H_{i2} \), the simplification of Fig. 2.15b is achieved, where \( J_{ji} \triangleq \begin{bmatrix} J_{ji1} & J_{ji2} \end{bmatrix} \). This is verified by applying Kirchoff's laws, as is shown below:

\[
v_j = J_{ji1} v_{i1} + J_{ji2} v_{i2} = \begin{bmatrix} J_{ji1} & J_{ji2} \end{bmatrix} \begin{bmatrix} v_{i1} \\ v_{i2} \end{bmatrix} = J_{ji} v_i ,
\]

and

\[
f_i = \begin{bmatrix} f_{i1} \\ f_{i2} \end{bmatrix} = \begin{bmatrix} J^*_{ji1} \\ J^*_{ji2} \end{bmatrix} f_j = J^*_{ji} f_j .
\]

The same approach can be applied to the series combination of transformers shown in Fig. 2.16. By introducing the space, \( H_j \triangleq H_{j1} \times H_{j2} \), the two transformers, \( J_{ji} : H_i \to H_{j1} \) and
Fig. 2.15. Transformers in parallel: a) two transformers in parallel; b) equivalent single transformer representation.

\[ J_{12ii} : \mathbf{H}_i \rightarrow \mathbf{H}_{j2} \], \ are combined to obtain \( J_{jii} : \mathbf{H}_i \rightarrow \mathbf{H}_j \), which is defined by

\[ J_{jii} \triangleq \begin{bmatrix} J_{1j}^{1ii} \\ J_{2j}^{1ii} \end{bmatrix} \]  \hspace{1cm} (2.39)

Applying Kirchhoff’s laws verifies the equivalence:

\[ v_j = \begin{bmatrix} v_{j1} \\ v_{j2} \end{bmatrix} = \begin{bmatrix} J_{1j}^{1ii} \\ J_{2j}^{1ii} \end{bmatrix} v_i = J_{jii} v_i, \]

and

\[ f_i = J_{1j}^{1ii} f_{j1} + J_{2j}^{1ii} f_{j2} = [J_{1j}^{1ii} J_{2j}^{1ii}] f_j = J_{jii}^* f_j. \]

In both the parallel and series combination cases, care must be taken that all the impedances are redefined on the new space. For instance in the series case, if the impedances, \( Z_{j1} : \mathbf{H}_{j1} \rightarrow \mathbf{H}_{j1} \) and \( Z_{j2} : \mathbf{H}_{j2} \rightarrow \mathbf{H}_{j2} \), are defined by \( f_{j1} = Z_{j1} v_{j1} \) and \( f_{j2} = Z_{j2} v_{j2} \) respectively, then the equivalent impedance, \( Z_j : \mathbf{H}_j \rightarrow \mathbf{H}_j \), should be defined as

\[ Z_j = \begin{bmatrix} Z_{j1} & 0_{j1/j2} \\ 0_{j2/j1} & Z_{j2} \end{bmatrix}. \]

Sometimes it is beneficial to decompose, rather than to combine, vector spaces. For instance, in the case that a Jacobian, \( J_{jii} \), is not of full rank, it is useful to divide the input space, \( \mathbf{H}_i \), into the null space of \( J_{jii} \) and its complement, and to divide the output space, \( \mathbf{H}_j \), into the span of \( J_{jii} \) and its complement. Define \( \mathbf{H}_{i2} = null (J_{jii}) \), \( \mathbf{H}_{i1} = \mathbf{H}_{i2} \), \( \mathbf{H}_{j1} = span (J_{jii}) \) and \( \mathbf{H}_{j2} = \mathbf{H}_{f1} \). The operator \( J_{jii} \) can then be decomposed to get
Fig. 2.16. Transformers in series: a) two transformers in series; b) equivalent single transformer representation.

\[ J_{ji} = \begin{bmatrix} J_{j1i1} & J_{j1i2} \\ J_{j2i1} & J_{j2i2} \end{bmatrix} = \begin{bmatrix} J_{j1i1} \\ J_{j2i1} \end{bmatrix} \begin{bmatrix} J_{i1i1} & J_{i1i2} \end{bmatrix}. \]  

(2.40)

These equations are represented by the decomposed network of Fig. 2.17b, in which both the series and parallel network simplifications have been reversed. By the definitions of the spaces, \( \mathbf{H}_{i1}, \mathbf{H}_{i2}, \mathbf{H}_{j1}, \) and \( \mathbf{H}_{j2}, \) it is clear that \( J_{j2i1} = 0_{j2i1}, J_{i1i2} = 0_{i1i2}, \) and \( J_{i1i1} = I_{i1}. \) Therefore, by replacing \( J_{j1i1} \) and \( J_{i1i2}, \) and \( J_{i1i1} \) with their short/open circuit equivalents, the network of Fig. 2.17c is obtained.

### 2.3.3 Partitioning of the identity operator

Although singular Jacobians cannot be inverted directly, there are cases where a series or parallel combination of transformers, which together span an entire space, are inverted together. An important example of this is the partition of the identity operator.

Consider a network operating in the space, \( \mathbf{H}_i. \) The identity operator in this space, \( I_i: \mathbf{H}_i \rightarrow \mathbf{H}_i, \) is split either with respect to inputs, so that \( I_i = [P_{i1i1} \ P_{i1i2}], \) or so that

\[ I_i = \begin{bmatrix} P_{i1i1} \\ P_{i2i1} \end{bmatrix}. \]

These two partitions correspond, respectively, to the parallel and series networks shown in Fig. 2.18. Computing Kirchoff’s laws for the two circuits proves their equivalence. For the parallel circuit,

\[ v_i = P_{i1i1}v_{i1} + P_{i1i2}v_{i2} = [P_{i1i1} \ P_{i1i2}] \begin{bmatrix} v_{i1} \\ v_{i2} \end{bmatrix} = I_i \begin{bmatrix} v_{i1} \\ v_{i2} \end{bmatrix}. \]
Fig. 2.17. Decomposing a singular Jacobian: a) original network; b) decomposition into series and parallel components; c) network with equivalent open/short circuits.

\[
\begin{bmatrix}
  f_{i1} \\
  f_{i2}
\end{bmatrix} = 
\begin{bmatrix}
P_{i11}^* \\
P_{i12}^*
\end{bmatrix} f_i = I_i f_i,
\]

and for the series circuit

\[
\begin{bmatrix}
v_{i1} \\
v_{i2}
\end{bmatrix} = 
\begin{bmatrix}
P_{i11} \\
P_{i21}
\end{bmatrix} v_i = I_i v_i
\]

\[
f_i = P_{i11}^* f_{i1} + P_{i21}^* f_{i2} = \begin{bmatrix} P_{i11}^* & P_{i21}^* \end{bmatrix} \begin{bmatrix} f_{i1} \\ f_{i2} \end{bmatrix} = I_i \begin{bmatrix} f_{i1} \\ f_{i2} \end{bmatrix}.
\]

From these equations, it is clear that \( P_{i11} = P_{i11}^* \), and \( P_{i21} = P_{i21}^* \). Although the individual partitions are not invertible directly, the series combination of input partitions is seen as the inversion of the parallel combination of output partitions, and vice versa. This result proves useful when hybrid control is introduced.

### 2.3.4 Reduction of a ladder network

One of the most frequently occurring structures in mechanical systems is the ladder network. This occurs not only when modeling a serial rigid link manipulator, but also when
modeling distributed behavior, such as flexible beams or transmission lines. The ladder network is substantially reduced by using Hilbert network reduction techniques.

Consider the ladder network shown in Fig. 2.19. The effort signals, $f_1, \cdots, f_{2n+1}$, and the flow signals, $v_0, \cdots, v_{2n}$, all belong to the same Hilbert space, $H_1$, and the impedances, $Z_1, \cdots, Z_{2n}$, all map $H_1$ into $H_1$. Here, the additional subscript, (e.g., $v_{0,i}$, $Z_{1,i}$), denoting the underlying Hilbert space is dropped, since all elements are defined on the same space. For physical systems such as a flexible beam or a transmission line, the even numbered impedances, $Z_2, Z_4, \cdots, Z_{2n}$, correspond to inertial elements, and the odd numbered impedances, $Z_1, Z_3, \cdots, Z_{2(n-1)}$, correspond to capacitive elements.

![Diagram of ladder network in $H_1$.](image)

The ladder network with $2n$ one-port elements is reduced to a network with two one-port elements. Define the even and odd spaces, $H_e = H_o \triangle H_1 \times H_1 \times \cdots \times H_1 = H_{e1}$, and let the even and odd flow vectors, $v_e \in H_n$ and $v_o \in H_n$, be given by $v_e = (v_2, v_4, \cdots, v_{2n})$ and $v_o = (v_1, v_3, \cdots, v_{2n-1})$, and assume the even and odd effort vectors, $f_e \in H_e$ and $f_o \in H_o$, are
defined similarly. Consider the ladder network shown in Fig. 2.20, where \( J_{1/o/1: H_1 \rightarrow H_o} \) and \( J_{2/o: H_o \rightarrow H_1} \) are defined by

\[
J_{1/o/1} = \begin{bmatrix}
I \\
0 \\
\vdots \\
0
\end{bmatrix}, \quad J_{2/o} = \begin{bmatrix}
I & I & \cdots & I \\
\end{bmatrix},
\] (2.41)

\( J_{e/o}: H_o \rightarrow H_e \) is defined by the lower triangular operator

\[
J_{e/o} = \begin{bmatrix}
I & 0 & \cdots & 0 \\
I & I & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
I & I & \cdots & I
\end{bmatrix},
\] (2.42)

and the odd and even impedance elements, \( Z_o: H_o \rightarrow H_o \) and \( Z_e: H_e \rightarrow H_e \), are defined by

\[
Z_o \triangleq \text{diag} (Z_1, Z_3, Z_5, \ldots, Z_{2n-1}),
\] (2.43)

\[
Z_e \triangleq \text{diag} (Z_2, Z_4, Z_6, \ldots, Z_{2n}),
\] (2.44)

respectively.

![Fig. 2.20. Reduced ladder network in H_n.](image)

Applying Kirchoff’s laws to the network gives for the output flow

\[
v_{2n} = J_{2/o} J_{e/o} v_e = \begin{bmatrix}
I & 0 & \cdots & 0 \\
-I & I & 0 & \cdots \\
\vdots & \vdots & \ddots & \vdots \\
0 & -I & I & \cdots \\
\vdots & \vdots & \vdots & \ddots \\
0 & 0 & 0 & \cdots & I
\end{bmatrix} \begin{bmatrix}
v_2 \\
v_4 \\
\vdots \\
v_6 \\
v_{2n}
\end{bmatrix}
\]

\[= J_{2/o} (J_{1/o/1} v_0 - v_o)\]
\[ 
= [I \ I \ \cdots \ I] \begin{bmatrix} I \\ 0 \\ \vdots \\ 0 \\ v_0 - \begin{bmatrix} v_1 \\ v_3 \\ \vdots \\ v_{2n-1} \\ \end{bmatrix} \\
\end{bmatrix} 
= v_0 - (v_1 + v_3 + \cdots + v_{2n-1}).
\]

The input force, \( f_1 \), is computed similarly:

\[ 
f_1 = J_{1_{0/1}^{*}Z_0} v_o \\
= [I \ 0 \ \cdots \ 0] Z_0 \cdot v_o = Z_1 v_1 \\
= J_{1_{0/1}^{*}(J_{e/0}^{*}Z_e \cdot v_e + J_{2_{1/0}f_{2n+1}})} \\
= [I \ 0 \ \cdots \ 0] \begin{bmatrix} I \ I \ \cdots \ I \\ 0 \ I \ \cdots \ I \\ \vdots \ \cdots \ \vdots \\ 0 \ 0 \ \cdots \ I \\
\end{bmatrix} \begin{bmatrix} Z_2 \ 0 \ \cdots \ 0 \\ 0 \ Z_4 \ \cdots \ 0 \\ \vdots \ \cdots \ \vdots \\ 0 \ 0 \ \cdots \ Z_{2n} \\
\end{bmatrix} \begin{bmatrix} v_2 \\ v_4 \\ \vdots \\ v_{2n} \\
\end{bmatrix} + [I \ 0 \ \cdots \ 0] f_{2n+1} \\
= Z_2 v_2 + Z_4 v_4 + \cdots + Z_{2n} v_{2n} + f_{2n+1}. 
\]

Thus, the circuit in Fig. 2.19 is identical to the circuit in Fig. 2.20.

As shown in Subsection 2.3.1, the transformer, \( J_{e/0} \), and the impedance, \( Z_e \), can be replaced by the equivalent impedance,

\[ 
Z_{1_{0}} = J_{e/0}^{*} Z_{e} J_{e/0} \\
= \begin{bmatrix} \Sigma_{i=1}^{n} Z_{2i} & \Sigma_{i=2}^{n} Z_{2i} & \cdots & Z_{2n} \\ \Sigma_{i=2}^{n} Z_{2i} & \Sigma_{i=3}^{n} Z_{2i} & \cdots & Z_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{2n} & Z_{2n} & \cdots & Z_{2n} \\
\end{bmatrix} 
\] (2.45)

An equivalent form of the ladder network is achieved by applying a useful trick. Consider the network shown in Fig. 2.21a in which a the unity valued transformer, \( I \), is added in cascade with the operator, \( Z_0 \). This does not affect the network equations in any way, but it does give a means to cancel the Jacobian, \( J_{e/0} \). By factoring the identity transformer into \( J_{e/0} \) and its inverse, \( J_{e/0}^{-1} \), as shown in Fig. 2.21b, and then propagating the inverse Jacobian, \( J_{e/0}^{-1} \), through the network to the right, and \( J_{e/0} \) to the left, the equivalent network of Fig. 2.21c is obtained, where \( J_{3_{e/1}:H_1\rightarrow H_e} \) is given by
\[
J_{3e/1} \Delta J_{e/0} J_{0/1} = \begin{bmatrix}
I & 0 & \cdots & 0 \\
I & I & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
I & I & \cdots & I
\end{bmatrix}
\begin{bmatrix}
I \\
0 \\
0 \\
\vdots \\
i
\end{bmatrix} = \begin{bmatrix}
i \\
0 \\
0 \\
\vdots \\
i
\end{bmatrix},
\]
(2.46)

and \(J_{4e/1}: \mathbf{H}_e \rightarrow \mathbf{H}_1\) is given by
\[
J_{4e/1} \Delta J_{2/0} = J_{2/0} J_{0/e}
\]
\[
= \begin{bmatrix}
I & 0 & \cdots & 0 \\
-I & I & \cdots & 0 \\
0 & -I & I & \cdots \\
0 & 0 & \cdots & I
\end{bmatrix}
= \begin{bmatrix}
0 & \cdots & 0 & 0 & I
\end{bmatrix}.
\]
(2.47)

The transformer, \(J_{e/0}^{-1}\), with the cascaded impedance, \(Z_0\), could be replaced by the equivalent impedance,
\[
Z_{1e} = (J_{e/0}^{-1})^* Z_0 J_{e/0}^{-1}
\]
\[
Z_{1e} = \begin{bmatrix}
Z_1 + Z_3 & -Z_3 & 0 & \cdots & 0 \\
-Z_3 & Z_3 + Z_5 & -Z_5 & \cdots & 0 \\
0 & -Z_5 & Z_5 + Z_7 & \cdots & \vdots \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & Z_{2n-1}
\end{bmatrix}.
\]
(2.48)

This trick can be applied to any circuit, using any decomposition of the identity transformer desired. When distributed parameter systems are investigated later in this work, the transformer/inverse transformer pair consists of a generalized Fourier transform and its inverse.

2.4 The PHIDE Network

In this section a useful subclass of Hilbert networks is introduced. The term, PHIDE network, describes a Passive Hilbert network which has Identity-valued Dynamic Elements. Thus, each inertia element, \(M_i: \mathbf{H}_i \rightarrow \mathbf{H}_i\), in a PHIDE network is identity valued, i.e.,
\[
f_i = \frac{d}{dt}(M_i v_i)
= \frac{d}{dt}(I_i v_i)
= \dot{v}_i.
\]
(2.49)

and each stiffness element, \(K_i: \mathbf{H}_i \rightarrow \mathbf{H}_i\), in a PHIDE network is identity valued, i.e.,
Fig. 2.21. Equivalent ladder network: a) unity-valued transformer included; b) transformer factored; c) transformers propagated through network.

\[ f_i = K_i x_i \]
\[ = I_i x_i \]
\[ = x_i. \]  \hspace{1cm} (2.50)

The kinetic energy at time, \( t \), associated with an inertia element in a PHIDE network is given by

\[ K.E.(t) = \frac{1}{2} <v_i, M_i v_i>_i \]
\[ = \frac{1}{2} <v_i, v_i>_i \]
\[ = \frac{1}{2} \| v_i \|^2, \]  \hspace{1cm} (2.51)

and the potential energy at time, \( t \), associated with a stiffness element in a PHIDE network is given by
\[ P.E. (t) = \int_0^t <K_i x_i, dx_i>_i + P.E. (0) \]
\[ = \int_0^t <x_i, dx_i>_i + P.E. (0) \]
\[ = \frac{1}{2} <x_i, x_i>_i |_{0}^{t} + P.E. (0) \]
\[ = \frac{1}{2} ||x_i||^2. \]  

(2.52)

For a PHIDE network, \( N \), the composite stiffness position vector, \( x_k \), and the composite inertial velocity vector, \( v_m \), make up a complete, though possibly redundant, set of states for the network. The total energy, \( T.E. \triangleq P.E. + K.E. \), makes a valid Lyapunov function, since

\[ V(x) = V(x_k, v_m) = T.E. = \frac{1}{2} ||x_k||^2 + \frac{1}{2} ||x_m||^2, \]

and by applying Lemma 2.1,

\[ \dot{V}(x) = <x_k, v_k>_k + <v_m, \dot{v}_m>_m \]
\[ = <f_k, v_k>_k + <v_m, f_m>_m \]
\[ = -<f_b, v_b>_b \leq 0 \quad \forall t \in [0, \infty). \]

From these properties the notational and conceptual advantages of using PHIDE network models of systems are clear. Fortunately, little generality is lost in this representation. Any network with constant, symmetric, positive definite dynamic elements, \( A_i \), can be represented in a PHIDE network by using transformers with associated Jacobians, \( J_i \triangleq A_i^{1/2} \), in cascade with the element. In Chapter 3, it is shown how many nonconstant, functionally dependent terms, such as manipulator inertias, nonlinear stiffness terms, and contact/noncontact of a workpiece’s surface, are also representable in a PHIDE network.

2.5 The Instability Problem for Nonscalar Gain Elements

The two-port gain element, \( A_i \), appears often in the design of control laws. It is used in the contemporary version of computed torque, for power scaling in teleoperation, and for applying force feedback. In general, the gain element is nonpassive, and can lead to instability in an otherwise passive system. Only in the case that \( A_i \) is scalar valued, do the passivity properties of the system remain intact. This is summarized in the theorem below.

**Theorem 2.7:** Consider the network shown in Fig. 2.22, where \( N1_i \) and \( N2_i \) represent two distinct, passive PHIDE subnetworks. If \( A_i = \alpha I_i \), where \( \alpha \in \mathbb{R} \) is constant and \( I_i \) is the identity operator on \( H_i \), then the complete network is stable in the sense of Lyapunov (i.s.L).
Fig. 2.22. Network subdivided by a power scaling element.

Proof Let \( x_{k1} \in H_{k1} \) and \( x_{k2} \in H_{k2} \) represent the composite stiffness position vectors for \( N_{1i} \) and \( N_{2i} \), and let \( v_{m1} \in H_{m1} \) and \( v_{m2} \in H_{m2} \) represent the composite inertial velocity vectors for \( N_{1i} \) and \( N_{2i} \). Let \( f_{k1}, f_{m1}, f_{k2}, \) and \( f_{m2} \) represent the corresponding composite efforts for these elements, so that: \( f_{k1} = x_{k1}, f_{m1} = \dot{v}_{m1}, f_{k2} = x_{k2} \) and \( f_{m2} = \dot{v}_{m2} \). Consider the Lyapunov function,

\[
V(x) = \frac{1}{2} \alpha^2 (||x_k||^2 + ||v_m||^2_1) + \frac{1}{2} (||x_{k2}||^2 + ||v_{m2}||^2_2).
\]

Taking the derivative and applying the definitions for the PHIDE dynamic elements gives

\[
\dot{V}(x) = \alpha^2 (\langle x_{k1},v_{k1}\rangle_{k1} + \langle v_{m1},\dot{v}_{m1}\rangle_{m1})
+ \langle x_{k2},v_{k2}\rangle_{k2} + \langle v_{m2},\dot{v}_{m2}\rangle_{m2}
= \alpha^2 (\langle f_{k1},v_{k1}\rangle_{k1} + \langle v_{m1},f_{m1}\rangle_{m1})
+ \langle f_{k2},v_{k2}\rangle_{k2} + \langle v_{m2},f_{m2}\rangle_{m2}
= -\alpha^2 (\langle f_{b1},v_{b1}\rangle_{b1} + \langle f_{l1},v_{l1}\rangle_{l1})
- (\langle f_{b2},v_{b2}\rangle_{b2} + \langle f_{l2},v_{l2}\rangle_{l2}),
\]

where \( f_{b1}, v_{b1}, f_{b2}, \) and \( v_{b2} \) are the composite damping vectors. Applying the defining equation for the amplifier gives

\[
\dot{V}(x) = -\alpha^2 \langle f_{b1},v_{b1}\rangle_{b1} - \langle f_{b2},v_{b2}\rangle_{b2} \leq 0,
\]

and thus the complete network is stable, i.s.L. \( \square \)

Although a scalar valued gain element maintains the passivity of a network, this is not true for all positive definite gain elements. Consider the system shown in Fig. 2.23., where all the signals are operating in \( \mathbb{R}^2 \). Assume that the parameters for the circuit have the following
values:

\[ K_k = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad B_m = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad M_m = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad A_{km} = \begin{bmatrix} 4 & 0 \\ 0 & \frac{1}{4} \end{bmatrix}. \]

![Diagram of a destabilizing gain element](image)

Fig. 2.23. Example of a destabilizing gain element.

Define the state vector, \( x = (x_k, v_m) = (x_{k1}, x_{k2}, v_{m1}, v_{m2}) \). The state equation for the system shown in Fig. 2.23 is then

\[
\dot{x}(t) = \begin{bmatrix}
0_k & A_{km} \\
-M^{-1}_m A_{km} K_k & -M^{-1}_m B_m
\end{bmatrix} x(t) \\
= \begin{bmatrix}
0 & 0 & 4 & 0 \\
0 & 0 & 0 & \frac{1}{4} \\
-\frac{1}{2} & -\frac{1}{4} & -2 & -1 \\
-4 & -8 & -1 & -2
\end{bmatrix} x(t),
\]

which has the characteristic equation,

\[ s^4 + 4s^3 + 7s^2 - 8.0625s + 4 = 0. \]

This equation has two roots in the right half plane, and thus the system is unstable.

This example shows that caution is necessary if all degrees of freedom are not scaled equally. For a teleoperator, this occurs when different power gains are used for angular and linear motions, and for a manipulator this occurs when the standard computed torque algorithm is used.
3. MODELING THE MANIPULATOR AND THE ENVIRONMENT

In this chapter the network models developed in the previous section are used for obtaining efficient and elegant representations of actual systems. First, networks are used to completely describe the kinematics and dynamics of the manipulator. The common, open-chain manipulator configuration appears as a ladder network. Even complicated manipulator structures, such as "closed-link" manipulators, are representable in a network form. Once the network model for the manipulator is constructed, the dynamics are derived by applying Kirchoff's laws.

Next, networks are used to describe environments which the manipulator might contact. Extending the usual notion of the environment, which assumes that the environment can be modeled by a 6-DOF impedance, to a model which allows for edge and surface contact, leads to a more intuitive description.

Finally in the third section, a brief look at distributed systems is taken. Distributed system modeling has applications in both the modeling of flexible joint manipulators and of flexible structures appearing in the environment, and is also useful for understanding distributed parameter behavior in general. Unfortunately, because of the limited scope of this work, only a cursory glance at distributed systems can be taken.

3.1 The Manipulator Model

In this section the Hilbert network model of the previous chapter is used to characterize an n-link robot manipulator. This model represents the complete, coupled, nonlinear system, including coriolis and centripetal forces, gravity, and actuator dynamics. Manipulator characteristics such as the manipulator Jacobian, the equivalent joint stiffness, and the dynamic equations, are derived by applying Kirchoff's laws to the network. Often, considerable simplification is achieved by applying network reduction techniques without requiring that equations be written explicitly, as is shown with a parallel linkage mechanism.

3.1.1 Robot spaces and frames

A standard pedestal-type robot consists of n rigid links and n joints. The origin of link 1 corresponds to joint 1, and represents the base of the manipulator. The tip of link n is called the gripper. The motion of a robot is controlled by actuators located at the joints. The actuators themselves consist of dc motors and gear trains, and typically operate with only one DOF. Each of the joints are either revolute or prismatic, depending on whether the joint actuator changes the relative angle or the relative distance between the previous link and the next.
A number of spaces are needed to describe robot motion. The space used to describe the motor and gear variables for link $i$ is called the actuator space for link $i$ and is denoted by $H_{ai}$. The space consisting of all the individual actuator spaces is denoted by $H_a$, and is called the composite actuator space, or simply the actuator space. Three different spaces are needed to describe positions and forces of each of the robot’s $n$ rigid links. The link space, $H_i$, is used for describing forces and velocities at the origin of link $i$. The center of mass space, $H_{ci}$, is used for describing forces and velocities of the center of mass of link $i$. Both $H_i$ and $H_{ci}$ are **fixed inertial frames**, since all positions and angles are measured with respect to a fixed position and orientation, corresponding to a Cartesian reference frame. The gripper space, $H_g$, is identical to any of the link spaces, $H_i$, except that it refers to the Cartesian motion of the end of link $i$. Because a rigid beam’s inertia tensor in a fixed space changes with the beam’s orientation, it is useful to define a third reference frame, $H_{mi}$, in which the beam’s inertia tensor is constant.

Transformations between two Cartesian reference frames, $i$ and $j$, are characterized by a rotation matrix, $R_{ji}$, which determines the change in orientation, and a distance vector, $d_{ji} \in \mathbb{R}^3$, which determines the distance between the two frame origins. When the rotation matrix relates a space, $H_i$, to a fixed Cartesian space, $H_j$, such as the link spaces, or the center of mass spaces, then the subscript, $j$, is dropped.

The distance from the base of a robot link $i$ to the tip of link $i$, is related to the length of a revolute link, $l_i \in \mathbb{R}^3$, by

$$d_{i+1/i} = R_i l_i.$$  \hfill (3.1)

Likewise, the distance from the base to the center of mass for the link is given by

$$d_{ci/i} = R_i l_{ci},$$ \hfill (3.2)

where $l_{ci}$ is the fixed distance between the base and the center of mass for the link.

By convention (Denavit and Hartenberg 1955), the rotation axis, or prismatic motion axis, for joint $i$ is given by

$$z_{i-1} = R_i \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \hfill (3.3)$$

### 3.1.2 Transformations between frames

Consider the link shown in Fig. 3.1, with Cartesian coordinate systems attached to the base and tip. The prismatic velocity, $v_{p(i+1)} \in \mathbb{R}^3$, at the tip is given in terms of the base
velocities, \( v_{pi} \in \mathbb{R}^3 \) and \( v_{\theta i} \in \mathbb{R}^3 \), by

\[
v_{lp(i+1)} = v_{pi} + v_{\theta i} \times d_{i+1/i}
\]

\[
= v_{pi} - S(d_{i+1/i})v_{\theta i},
\]

and the angular velocity at the tip, \( v_{l\theta(i+1)} \in \mathbb{R}^3 \), is simply,

\[
v_{l\theta(i+1)} = v_{\theta i}.
\]

In matrix form these equations are given by,

\[
v_{l+1} = \begin{bmatrix} v_{lp(i+1)} \\ v_{l\theta(i+1)} \end{bmatrix} = \begin{bmatrix} I_3 & -S(d_{i+1/i}) \\ 0_3 & I_3 \end{bmatrix} \begin{bmatrix} v_{pi} \\ v_{\theta i} \end{bmatrix} = \begin{bmatrix} I_3 & -S(d_{i+1/i}) \\ 0_3 & I_3 \end{bmatrix} v_i.
\] (3.4)

Thus, the Jacobian, \( J_{i+1/i} \), relating \( v_{l+1} \) to \( v_i \) is

\[
J_{i+1/i} = \begin{bmatrix} I_3 & -S(d_{i+1/i}) \\ 0_3 & I_3 \end{bmatrix}.
\] (3.5)

As expected, the force, \( f_i \), is related to the force, \( f_{i+1} \), by

\[
f_i = \begin{bmatrix} f_{pi} \\ f_{\theta i} \end{bmatrix} = \begin{bmatrix} I_3 & 0_3 \\ S(d_{i+1/i}) & I_3 \end{bmatrix} \begin{bmatrix} f_{p(i+1)} \\ f_{\theta(i+1)} \end{bmatrix} = \begin{bmatrix} f_{p(i+1)} \\ f_{\theta(i+1)} + d_{i+1/i} \times f_{p(i+1)} \end{bmatrix} = J_{i+1/i}^T f_{i+1}.
\] (3.6)

A similar analysis can be applied to determine the Jacobian which relates actuator velocity to link velocity. The contribution of the actuator velocity, \( v_{ai} \in \mathbb{R} \), to the link prismatic and angular velocities is given by

\[
v_{2pi} = \begin{cases} 
0 & \text{if joint } i \text{ is revolute} \\
z_{i-1}v_{ai} & \text{if joint } i \text{ is prismatic}
\end{cases}
\]
\[ v_{2i} = \begin{cases} 
  z_{i-1} v_{ai} & \text{if joint } i \text{ is revolute} \\
  0 & \text{if joint } i \text{ is prismatic.}
\end{cases} \]

In matrix form, this is given by
\[
v_{2i} = \begin{bmatrix} v_{2pi} \\ v_{2\theta i} \end{bmatrix} = \begin{bmatrix} 
  0 \\ z_{i-1} \end{bmatrix} v_{ai} \quad \text{if joint } i \text{ is revolute}
\begin{bmatrix} 
  z_{i-1} \\ 0 \end{bmatrix} v_{ai} \quad \text{if joint } i \text{ is prismatic.}
\] (3.7)

Likewise, the actuator force (i.e., motor torque) is given by
\[
f_{ai} = \begin{bmatrix} 0 & z_{i-1}^T \end{bmatrix} f_i \quad \text{if joint } i \text{ is revolute}
\begin{bmatrix} z_{i-1}^T \\ 0 \end{bmatrix} f_i \quad \text{if joint } i \text{ is prismatic.}
\] (3.8)

Therefore, the Jacobian relating the n-DOF link motion, \( v_{2i} \), to the one DOF actuator motion, \( v_{ai} \), is given by
\[
J_{i/ai} = \begin{cases} 
  0 & \text{if joint } i \text{ is revolute} \\
  z_{i-1} \end{cases}
\begin{bmatrix} z_{i-1} \\ 0 \end{bmatrix} \quad \text{if joint } i \text{ is prismatic.} \] (3.9)

3.1.3 The nondynamic network model

![Diagram of a nondynamic circuit model for a single link.](image)

Fig. 3.2. Nondynamic circuit model for a single link.

Equations (3.4)-(3.9) for the single link can be represented in the form of a circuit as is shown in Fig. 3.2. The actuator is represented by a one-dimensional effort source, \( f_{cai} \), and is
connected to the rest of the circuit via the transformer, \( J_{ilai} \). The link is connected to the next link via the transformer, \( J_{i+1/i} \).

The network model for the open-chain n-link manipulator is created by connecting \( n \) of these single link models in series. Figure 3.3 shows the result. From this model, characteristics such as the manipulator Jacobian, equivalent stiffness, etc., are derived.

![Diagram of serial manipulator model](image)

Fig. 3.3. Nondynamic model for serial manipulator.

3.1.3.1 The manipulator Jacobian

The manipulator Jacobian, \( J_{g/a}: \mathbb{H}_a \rightarrow \mathbb{H}_g \), is the particular Jacobian which relates the composite actuator space, \( \mathbb{H}_a = \mathbb{R}^n \), to the gripper space, \( \mathbb{H}_g = \mathbb{R}^6 \) (\( \mathbb{R}^3 \) for planar systems). It is represented in the decomposed form shown below.

\[
\mathbf{v}_g = J_{g/a} \mathbf{v}_a = \begin{bmatrix} J_{g/a1} & J_{g/a2} & \cdots & J_{g/an} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{a1} \\ \vdots \\ \mathbf{v}_{an} \end{bmatrix}
\]  

(3.10)

To determine \( J_{g/a} \), first iteratively calculate the velocity of each link by applying Kirchoff’s laws to the circuit shown in Fig. 3.3. This gives for links \( i = 1 \cdots n \)

\[
v_{i+1} = J_{i+1/a(i+1)} \mathbf{v}_a(i+1) + J_{i+1/i} v_i
\]

\[
= J_{i+1/a(i+1)} \mathbf{v}_a(i+1) + J_{i+1/i} J_{i/i} v_{ai} + \cdots,
\]

and for the last link

\[
v_g = J_{g/n} \mathbf{v}_n.
\]

Multiplying the Jacobians for two connected links gives

\[
J_{i+1/i-1} = J_{i+1/i} J_{i/i-1}
\]

\[
= \begin{bmatrix} I_3 & -S(d_{i+1/i}) \\ 0_3 & I_3 \end{bmatrix} \begin{bmatrix} I_3 & -S(d_{ii-1}) \\ 0_3 & I_3 \end{bmatrix}
\]
\[
\begin{bmatrix}
I_3 & -S(d_{i+1/i}) - S(d_{i/i-1}) \\
0_3 & I_3
\end{bmatrix}
= \begin{bmatrix}
I_3 & -S(d_{i+1/i-1}) \\
0_3 & I_3
\end{bmatrix},
\]

and therefore,
\[
v_g = J_{g/ai} v_{a_1} + J_{g/ai} v_{a_2} + \cdots + J_{g/ai} v_{a_n},
\]

where \( J_{g/ai} \) is given by
\[
J_{g/ai} = J_{g/in} J_{n/n-1} J_{n-1/n-2} \cdots J_{i+1/i} J_{i/ai}
\]
\[
= \begin{bmatrix}
I_3 & -S(d_{g/ii}) \\
0_3 & I_3
\end{bmatrix} J_{i/ai}.
\]

Thus, the columns of the manipulator Jacobian are given by
\[
J_{g/ai} = \begin{cases}
\begin{bmatrix}
z_{i-1} \\
0
\end{bmatrix} & \text{if joint } i \text{ is prismatic} \\
-S(d_{g/ii}) z_{i-1} \\
z_{i-1}
\end{cases}
\]
(3.11)

3.1.3.2 Finding the joint stiffness

The joint stiffness for a manipulator refers to the apparent stiffness of the manipulator as seen from the environment's gripper. To find the joint stiffness for a serial manipulator, replace each actuator with a stiffness element, \( K_{ai}: \mathbb{R} \rightarrow \mathbb{R} \), as shown in Fig. 3.4. This corresponds to a position feedback with gain, \( K_{ai} \), around a zero set-point. Applying a probing force to the end effector, \( f_g \), and assuming the motion is small (so that \( \delta x = \Delta x \) ) gives

![Fig. 3.4. Joint stiffness model.](image)
\[
\Delta x_{ai} = K_{ai}^{-1} f_{ai} = K_{ai}^{-1} J_{l(ai)}^T f_i = K_{ai}^{-1} J_{g(ai)}^T f_g ,
\]

and the total deflection at the end effector, \( \Delta x_g \), is given by

\[
\Delta x_g = \sum_{i=1}^r J_{g(ai)} \Delta x_{ai} = \sum_{i=1}^r J_{g(ai)} K_{ai}^{-1} J_{l(ai)}^T f_g ,
\]

which can be written more efficiently in matrix form as

\[
\Delta x_g = [ J_{g/a1} \cdots J_{g/ln} ] \begin{bmatrix} K_{a1}^{-1} & 0 & \cdots & 0 \\ 0 & K_{a2}^{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & K_{an}^{-1} \end{bmatrix} \begin{bmatrix} J_{g(a1)}^T \\ \vdots \\ J_{g(an)}^T \end{bmatrix} f_g
\]

\[
= J_{g/a} K_a^{-1} J_{g/a}^T f_g .
\]

Thus the equivalent stiffness seen at the gripper is defined by

\[
K_g \Delta (J_{g/a} K_a^{-1} J_{g/a})^{-1} .
\]

(3.12)

### 3.1.4 Manipulator dynamics

The dynamic circuit model extends the previous circuit model to include the dynamic effects of link inertias. To represent these effects new reference frames for the centers of mass of each link need to be added to the nondynamic network model. Fig. 3.5 shows the dynamic circuit model for the manipulator link shown previously in Fig. 3.1.

![Dynamic circuit model for one link.](image)

The velocity, \( v_c \), of the center of mass frame is related to the link origin frames by the Jacobians, \( J_{ci/i} \) and \( J_{i+1/ci} \), in the usual fashion:

\[
v_{ci} = J_{ci/i} v_i = \begin{bmatrix} I & -S(d_{ci/i}) \\ 0 & I \end{bmatrix} v_i ,
\]

(3.13)

\[
v_{i+1} = J_{i+1/ci} v_{ci} = \begin{bmatrix} I & -S(d_{i+1/ci}) \\ 0 & I \end{bmatrix} v_{ci} ,
\]

(3.14)
where \( d_{ci} \) is the vector from joint \( i \) to the center of mass (COM) of joint \( i \), and \( d_{i+1/ci} \) is the vector from the COM of joint \( i \) to joint \( i+1 \) as shown in Fig. 3.1. The center of mass angular and prismatic velocities, \( v_{\theta ci} \in \mathbb{R}^3 \) and \( v_{pci} \in \mathbb{R}^3 \), respectively, are uncoupled. The prismatic force is obtained from the center of mass linear velocity, \( v_{pci} \)

\[
f_{pci} = m_i v_{pci} = \frac{d}{dt}(m_i v_{pci}),
\]

where \( m_i \) is the mass of the link. Similarly, the angular force, or moment, \( f_{\theta ci} \) is obtained from the center of mass angular velocity, \( v_{\theta ci} = v_{\theta i} \), as is shown below,

\[
f_{\theta ci} = M_{c\theta i}(x_{\theta i}) \dot{v}_{\theta ci} + v_{\theta c(i+1)} \times M_{c\theta i}(x_{\theta i}) v_{\theta c(i+1)}.
\]

The moment of inertia, \( M_{c\theta i}(x_{\theta i}) \) can be expressed in terms of a constant matrix, \( M_{m\theta i} \) and the rotation matrix, \( R(x_{\theta i}) \).

\[
M_{c\theta i}(x_{\theta i}) = R_i(x_{\theta i}) M_{m\theta i} R_i^T(x_{\theta i}).
\]

Taking the derivative of \( M_{c\theta i} \) gives

\[
\dot{M}_{c\theta i} = S(v_{\theta i}) M_{c\theta i} + M_{c\theta i} S^T(v_{\theta i}),
\]

and thus the moment can be expressed as,

\[
f_{\theta ci} = M_{c\theta i} \dot{v}_{\theta i} + S(v_{\theta i}) M_{c\theta i} v_{\theta i} + M_{c\theta i} v_{\theta i} \times M_{c\theta i} S(v_{\theta i}) v_{\theta i}
\]

\[
= \frac{d}{dt}(M_{c\theta i} v_{\theta i}).
\]

In matrix form, this reduces to

\[
f_{ci} = \begin{bmatrix} f_{pci} \\ f_{\theta ci} \end{bmatrix} = \frac{d}{dt} \begin{bmatrix} m_i I_3 & 0_3 \\ 0_3 & M_{c\theta i} \end{bmatrix} v_{ci}.
\]

Since this is nothing more than the constituent equation for an inertia term, the inertia operator for the link in the space, \( H_{ci}, M_{ci}(x_{\theta i}) \), is defined by

\[
M_{ci}(x_{\theta i}) = \begin{bmatrix} m_i I_3 & 0_3 \\ 0_3 & M_{c\theta i}(x_{\theta i}) \end{bmatrix}.
\]

3.1.4.1 Obtaining the dynamics from the model

Applying Kirchhoff’s laws to the circuit shown in Fig. 3.5 gives the dynamic equations for the manipulator. As before, the velocity for link \( i+1 \) in terms of link \( i \) is given by

\[
v_{i+1} = J_{i+1/ci} v_{ci} + J_{i+1/a(i+1)} v_{a(i+1)}
\]
\[ = J_{i+1/i} v_i + J_{i+1/a(i+1)} v_{a(i+1)}. \]  

With the addition of the mass element, the force at link \( i \) is now expressed by

\[
f_i = J_{ei/i}^T \left( \frac{d}{dt} (M_{ci} (x_{bi}) J_{ei/i} v_i) + J_{T_{i+1/ci}}^T f_{i+1} \right)
= J_{ei/i}^T \left( \frac{d}{dt} (M_{ci} (x_{bi}) J_{ei/i} v_i) + J_{T_{i+1/ci}}^T f_{i+1} \right),
\]

and the joint torque is given by

\[
f_{ai} = J_{T_{i/ai}}^T f_i.
\]

Equations (3.17)-(3.19) are commonly called the iterative form of the Newton-Euler dynamics (Asada and Slotine 1986). By applying them to each link in the manipulator the dynamics for the entire manipulator can be obtained.

The joint torques can also be expressed in closed form as

\[
f_{ai} = J_{T_{i/ai}}^T \sum_{j=i}^{+} J_{T_{j/ai}}^T \frac{d}{dt} (M_{cj} \sum_{k=1}^{j} J_{eik} J_{k/ak} v_{ak}).
\]

To be of use, however, the forces and velocities need to be expressed in terms of link variables. Therefore closed-form expressions for the derivatives need to be derived.

### 3.1.4.2 Obtaining derivatives

Taking the derivative of Eq. (3.17) gives

\[
\dot{v}_{i+1} = \dot{J}_{i+1/i} v_i + \dot{J}_{i+1/a(i+1)} v_{ai} + J_{i+1/i} \ddot{v}_i + J_{i+1/a(i+1)} \ddot{v}_{ai}.
\]

By taking advantage of the behavior of the skew symmetric matrix, \( S (d_{i+1/i}) \), the derivatives of the Jacobians, \( \dot{J}_{i+1/i} \) and \( \dot{J}_{i+1/a(i+1)} \) are found. First notice that

\[
\frac{d}{dt} S (d_{i+1/i}) = S \left( \frac{d}{dt} d_{i+1/i} \right) = S \left( \frac{d}{dt} R_i l_i \right)
= S (S (\dot{v}_{bi}) R_i l_i) = S (S (\dot{v}_{bi}) d_{i+1/i}),
\]

and likewise,

\[
\frac{d}{dt} z_{i-1} = \frac{d}{dt} R_{i-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = S (\dot{v}_{\theta(i-1)}) z_{i-1}.
\]

Therefore,

\[
\dot{J}_{i+1/i} = \begin{bmatrix} 0 & -S (S (\dot{v}_{bi}) d_{i+1/i}) \\ 0 & 0 \end{bmatrix},
\]

(3.21)
and

$$\dot{j}_{i+1/a(i+1)} = \begin{cases} 
0 \\
S(v_{bi})z_i \\
S(v_{bi})z_i \\
0 
\end{cases} 
\begin{bmatrix} 
0 \\
\dot{v}_{bi} \\
\dot{v}_{bi} \\
0 
\end{bmatrix}$$

if joint $i$ is revolute

$$\dot{j}_{i+1/a(i+1)} = \begin{cases} 
0 \\
S(v_{bi})z_i \\
S(v_{bi})z_i \\
0 
\end{cases} 
\begin{bmatrix} 
0 \\
\dot{v}_{bi} \\
\dot{v}_{bi} \\
0 
\end{bmatrix}$$

if joint $i$ is prismatic.

Writing out the acceleration for the revolute case gives

$$\dot{v}_{i+1} = \begin{bmatrix} 0 & -S(S(v_{bi})d_{i+1/i}) & v_{pi} \\
0 & 0 & \dot{v}_{bi} \\
0 & S(v_{bi})z_i & 0 
\end{bmatrix} \begin{bmatrix} \dot{v}_i \\
0 \\
\dot{v}_{bi} 
\end{bmatrix} + \begin{bmatrix} I \\
0 \\
0 
\end{bmatrix} \begin{bmatrix} S(d_{i+1/i}) \\
0 \\
z_i 
\end{bmatrix} \dot{v}_{a(i+1)},$$

which, when taken component-wise and expanded in terms of cross products, results in the familiar dynamic equations for a revolute joint:

$$\dot{v}_{p(i+1)} = -S(S(v_{bi})d_{i+1/i})v_{bi} + \dot{v}_{pi} - S(d_{i+1/i})\dot{v}_{bi}$$

$$= v_{bi} \times (v_{bi} \times d_{i+1/i}) + \dot{v}_{pi} + v_{bi} \times d_{i+1/i}$$

$$\dot{v}_{bi} = \dot{v}_{bi} + S(v_{bi})z_i v_{ai} + z_i \dot{v}_{a(i+1)}$$

A similar result is reached for the prismatic case.

The acceleration, $\dot{v}_{ci}$, is obtained from $v_{ci}$ using the same simplifications.

$$\dot{v}_{ci} = \dot{j}_{ci/i} v_i + J_{ci/i} \dot{v}_i$$

$$= \begin{bmatrix} 0_3 & S(S(v_{bi})d_{ci/i}) \\
0_3 & 0_3 
\end{bmatrix} v_i + \begin{bmatrix} I_3 \\
0_3 
\end{bmatrix} \begin{bmatrix} S(d_{ci/i}) \\
0_3 
\end{bmatrix} \dot{v}_i.$$  

(3.24)

3.1.4.3 Including gravity in the model

The effect of gravity can be accounted for in two ways. First, a force source $f_{di} = m_i g$, is added in series with each mass in the model, as shown in Fig. 3.6a. Second, and more elegantly (Craig 1986), the base of the manipulator is given a constant acceleration of $\ddot{v}_1 = -g_1$, where $g_1 \triangleq (0, 0, 0, 0, 0, 0)$ is the gravity vector in the Cartesian space, $H_0$. In the first case, the resulting joint torques are given by

$$f_{ai} = J_{i/lai} \sum_{j=1}^k J_{ej/i} \left( \frac{d}{dt} (M_{cij} \sum_{k=1}^i J_{ek/ik} k_{iak} v_{ak}) + m_i g_1 \right).$$  

(3.25)
Fig. 3.6. Adding gravity to the model: a) adding an effort source to each mass; b) adding a flow source to the base.

and in the second case, the resulting joint torques are given by

$$f_{ai} = J_{iai} \sum_{j=1}^{n} T_{ji} \frac{d}{dt} (M_{cj}(\sum_{k=1}^{i} J_{cjik} J_{klk} v_{ak} + J_{cjik} \dot{q}_{k})).$$  \hspace{2cm} (3.26)

A quick inspection shows that Eq. (3.25) and Eq. (3.26) are identical.

3.1.5 Actuator dynamics

Until now the actuator has been represented as an independent source. In this section this model is extended to include the actuator dynamics. The actuator consists of: a motor, with resistance $R_{ai}$, inductance, $L_{ai}$, torque constant, $G_{ai}$, voltage, $V_{ai}$, and rotor inertia, $M_{1ai}$; and a gear train with gear ratio, $n_{ai}$, and gear inertia, $M_{2ai}$. Because motors and gears operate with only one DOF, the elements associated with them become scalar quantities, and a standard circuit model is used. Fig. 3.7 shows the network model for a dc-motor with gear reduction. A gyrator, $G_{ai}$, couples the electrical portion of the circuit to the mechanical portion.

3.1.6 Reduction of the manipulator ladder network

Applying the network reduction techniques of Chapter 2 to the network model for the manipulator results in an even simpler representation. Assume that the manipulator is an open-chain mechanism. In this case the manipulator network for the system is a ladder network of the form shown in Fig. 2.7. By defining the composite center of mass space, $H_{c} \triangleq H_{c1} \times \cdots \times H_{cn}$, and the composite actuator space, $H_{a} \triangleq H_{a1} \times \cdots \times H_{an}$, the network shown in Fig. 3.8 is created.
Fig. 3.7. Network diagram showing actuator dynamics.

Fig. 3.8. Reduced manipulator ladder network.

The effort source in the composite actuator space, $\mathbf{H}_a$, is given by

$$f_{ca} \triangleq (f_{ca1}, \ldots, f_{can}).$$

The composite center of mass inertia operator, $M_c : \mathbf{H}_c \rightarrow \mathbf{H}_c$, is a $6n \times 6n$ matrix given by

$$M_c \triangleq \text{diag} (M_{c1}, \ldots, M_{cn}).$$

The element, $M_c$, is connected to the actuator space network via the transformer, $J_{c/a} : \mathbf{H}_a \rightarrow \mathbf{H}_c$. This transformer is defined by

$$J_{c/a} \triangleq \begin{bmatrix}
J_{c1/a1} & 0_{6/1} & \cdots & 0_{6/1} \\
J_{c2/a1} & J_{c2/a2} & \cdots & 0_{6/1} \\
\vdots & \vdots & \ddots & \vdots \\
J_{cn/a1} & J_{cn/a2} & \cdots & J_{cn/an}
\end{bmatrix},$$

where each of the Jacobians, $J_{ci/aj}$, are defined by

$$J_{ci/aj} \triangleq J_{ci/i-1}J_{i-1/i-2} \cdots J_{j+1/j}J_{j/aj}.$$

The actuator space is coupled to the gripper space by means of the manipulator Jacobian, $J_{g/a}$. 
The manipulator model shown in Fig. 3.8. is reduced to the standard form shown in Fig. 3.9 by pulling the manipulator inertia through the transformer, $J_{c/a}$. The manipulator inertia in actuator space is thus given by

$$M_a \triangleq J_{c/a} C J_{c/a}$$

and because $J_{c/a}$ is time-varying, the additional fictitious damping term, $D_a$ defined by

$$D_a \triangleq -J_{c/a} C J_{c/a},$$

is created.

From the standard form of the manipulator network, the standard form of the manipulator dynamic equations is obtained.

$$f_{c/a} = \frac{d}{dt}(M_a \dot{v}_a) + D_a v_a + J_{g/a}^T f_g$$

$$= M_a \ddot{v}_a + C_a v_a + J_{g/a}^T f_g.$$  \hspace{1cm} (3.27)

### 3.1.7 Closed chain manipulators

The analysis presented so far has been based on an open chain model for the manipulator, i.e., manipulator configurations in which the removal of any link from the manipulator would lead to two distinct "submanipulators." This is not, however, the only desirable configuration for a robot. Closed chain manipulator configurations provide many benefits, such as greater freedom in the placement of actuators, the ability to uncouple the manipulator dynamics, and increased stiffness.

For the open chain manipulator, the dynamic equations are readily computed by iteratively calculating all of the velocities and accelerations from link 1 to link $n$, and then iteratively computing the force back from link $n$ to link 1. This represents the Newton-Euler approach for obtaining the manipulator's dynamic equations. For a closed-chain manipulator,
however, it is not possible to apply this "kinematics forward/dynamics backward" approach due to the constraining nature of the closed link. For this reason, the closed chain dynamic formulation has been ignored in most robotics textbooks. Recently, however, a new approach for the solution of closed chain robot dynamics has been presented by Murray and Lovell (1987) based on the principle of virtual work. Because the principle of virtual work is directly related to the constituent equations for the transformer, this approach is readily interpreted in terms of networks.

Suppose that the manipulator's closed chain was opened at some nonactuated joint, resulting in a manipulator with two open-chains. Assume that pseudo-actuators exist at each joint of the two open-chains. The approach of the previous sections gives the joint torque for the open chain manipulators. Now, suppose that these pseudo-actuators cause the two opened-chain manipulator segments to track the same trajectory as the original closed chain manipulator. In this case, the pseudo-actuators use the same amount of energy as the original system, and by applying the principle of virtual work it is possible to obtain the actuator torque from the pseudo-actuator torque, as is done below.

Let \( H_a \) represent the space of all of the existing actuators in the closed chain robot, and let \( H_d \) represent the "psuedo-actuator" space, consisting of all the joints in the two open-chains. If the opened system follows the same trajectory as the closed system, then the position of each of the nonopened joints, \( x_d \), is related to the actuated joint positions, \( x_a \), by the constraint equation,

\[
x_d \overset{\Delta}{=} \Omega(x_a).
\] (3.28)

Taking the derivative gives

\[
\frac{d}{dt}x_d = v_d = \frac{d}{dt}\Omega(x_a) = J_{\dot{d}/a}v_a,
\] (3.29)

and from the principle of virtual work the actuator torque, \( f_{c_a} \), is related to the pseudo-actuator torque, \( f_{c_d} \), by the equation,

\[
f_{c_a} = J_{\dot{a}/a}^Tf_{c_d}.
\] (3.30)

Thus, if the dynamics for the serial "psuedo" system are given by

\[
f_{c_d} = M_d\dot{v}_d + C_d\dot{v}_d + J_{\dot{g}/d}^Tf_g,
\] (3.31)

then the dynamics for the original closed-link system are given by

\[
f_{c_a} = J_{\dot{a}/a}^TM_dJ_{\dot{d}/a}v_a + (J_{\dot{a}/a}^TM_dJ_{\dot{d}/a}J_{\dot{d}/a}M_dJ_{\dot{d}/a}^T + J_{\dot{g}/d}^TC_dJ_{\dot{d}/a}^T)f_{c_d} + J_{\dot{g}/d}^Tf_g = M_a\dot{v}_a + C_a\dot{v}_a + J_{\dot{g}/d}^Tf_g.
\] (3.32)
This approach is shown in Fig. 3.10, and illustrated by example in Subsection 3.1.8.2.

![Diagrams](image)

Fig. 3.10. Closed-link manipulator network: a) psuedo-network model; b) reduced model.

### 3.1.8 Examples

In this section the network approach for a two-link open-chain serial arm and a four-link closed-chain parallel arm is demonstrated. For simplicity, only planar systems are considered. For a system operating in a plane, only three-DOF are allowed, and thus the forces and velocities are represented by vectors of the form, \( f = (f_x, f_y, f_z) \), \( \nu = (\nu_x, \nu_y, \nu_z) \), and the inertias are represented by diagonal 3×3 matrices of the form, \( M_{ci} = \text{diag} (m_i, m_i, m_{\theta i}) \), where \( m_i \) is the mass of the link, and \( m_{\theta i} \) is the moment of inertia around the z-axis. The Jacobians relating two link spaces, \( H_i \) and \( H_j \), are given by

\[
J_{i/j} = \begin{bmatrix} 1 & 0 & -d_{jix} \\ 0 & 1 & d_{jiy} \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.33}
\]

where \( d_{jix} \) and \( d_{jiy} \) are the x and y components, respectively, of the distance between the frames.

#### 3.1.8.1 The two-link revolute manipulator

Figure 3.11 shows the two-link system and the corresponding network model. The actuator position, i.e., angle, of link 1, is given by \( x_{a1} \), and the actuator position of link 2, is given by \( x_{a2} \). The inertia matrices for the system are given by \( M_{c1} = \text{diag} (m_1, m_1, m_{\theta 1}) \) and \( M_{c2} = \text{diag} (m_2, m_2, m_{\theta 2}) \), and the Jacobians of interest are given by

\[
J_{c1/1} = \begin{bmatrix} 1 & 0 & -l_{c1}S_1 \\ 0 & 1 & l_{c1}C_1 \\ 0 & 0 & 1 \end{bmatrix}, \quad J_{2/1} = \begin{bmatrix} 1 & 0 & -l_1S_1 \\ 0 & 1 & l_1C_1 \\ 0 & 0 & 1 \end{bmatrix}
\]
Fig. 3.11. Two-link manipulator: a) diagram; b) network model.

\[
J_{c2/2} = \begin{bmatrix}
1 & 0 & -l_c S_{12} \\
0 & 1 & l_c C_{12} \\
0 & 0 & 1
\end{bmatrix}, \quad J_{a1/1} = J_{a2/2} = \begin{bmatrix}
0 \\
0 \\
1
\end{bmatrix},
\]

where the abbreviations, \( C_1 = \cos(x_{a1}) \), \( S_1 = \sin(x_{a1}) \), \( S_2 = \sin(x_{a2}) \), and \( C_{12} = \cos(x_{a1}+x_{a2}) \), have been used. By applying Kirchoff’s laws to the network, the dynamic equations can be determined. The force, \( f_2 \), at joint two is given by

\[
f_2 = J_{c2/2}^T \frac{d}{dt} (M_{c2} v_{c2}) = J_{c2/2}^T M_{c2} \frac{d}{dt} [v_{c2}(J_{2a2} v_{a2} + J_{2a1} v_{a1})],
\]

and the actuator torque, \( f_{a2} \in \mathbb{R} \), at joint two is given by

\[
f_{a2} = J_{a2/2}^T f_2 = \left[ -l_c S_{12} \quad -l_c S_{12} \quad 1 \right] M_{c2} \frac{d}{dt} \left( \begin{bmatrix}
-l_1 S_1 & -l_c S_{12} \\
0 & l_1 C_1 + l_c C_{12} \\
0 & 0
\end{bmatrix} v_{a1} + \begin{bmatrix}
-l_c S_{12} \\
0 \\
0
\end{bmatrix} v_{a2} \right)
\]

\[
= (m_2 l_c^2 + m_2 l_1 l_c C_2 + m_{a2}) \ddot{v}_{a1}
\]

\[
+ (m_2 l_c^2 + m_{a2}) \ddot{v}_{a2} + l_c C_{12} m_2 g + m_2 l_1 l_c C_2 \dddot{v}_{a1}.
\]

(3.34)
Similarly, the actuator torque, \( f_a \in \mathbb{R} \), at joint 1 is given by

\[
    f_{a1} = J_{i1/a1} J_{i1/1}(\frac{d}{dt}(M_{c1} \dot{v}_{c1}) + J_{c1} \dot{f}_2).
\]

\[
= J_{i1/a1} (M_{c1} \frac{d}{dt}(J_{c1/a1} \dot{v}_{a1}) + J_{c2/c1} M_{c2} \frac{d}{dt}[J_{c2/2}(J_{2/a2} \dot{v}_{a2} + J_{2/1} \dot{v}_{a1})])
\]

\[
= (m_1 l_{c1}^2 + m_{\theta1} + m_2 (l_1^2 + l_2^2 + 2l_1 l_2 C_2) + m_{\theta2}) \ddot{v}_{a1}
\]

\[
+ (m_2 (l_2^2 + l_1 l_2 C_2) + m_{\theta2}) \ddot{v}_{a2} - 2l_1 l_2 S_2 \dot{v}_{a1} \dot{v}_{a2}
\]

\[
- l_1 l_2 S_1 \dot{v}_{a2}^2 + g (m_1 l_{c1} C_1 + m_2 (l_1 C_1 + l_2 C_{12}))
\]

\[ (3.35) \]

### 3.1.8.2 The parallel-link manipulator

The parallel-link manipulator is a special case of a closed chain manipulator configuration, in which the closed chain consists of a parallelogram. Fig. 3.12 shows a parallel manipulator, and Fig. 3.13 shows the equivalent network model. With the proper mass distribution the dynamic equations for the parallel-link manipulator are uncoupled and linear (Asada and Youcef-Toumi 1984). In this section, circuit models are used to clarify this result.

![Parallel-link manipulator](image)

**Fig. 3.12.** Parallel-link manipulator.

Although the equations could be obtained from applying Kirchoff’s laws to Fig. 3.13 directly, this would involve solving a set of constrained equations. Instead the approach of Subsection 3.1.6 is applied. Assume that joint 4 of the manipulator has been opened, so that the pseudo-actuator velocity vector, \( \dot{v}_d \), consists of the velocities of the joints 1, 2, 3 and 5.
vector, $v_d$, is then related to the actuator velocity vector, $v_a$, by

$$v_d = \begin{bmatrix} v_{a1} \\ v_{a5} \\ v_{a2} \\ v_{a3} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} v_{a1} \\ v_{a2} \end{bmatrix} = J_{a/a} v_a.$$  

From the principle of virtual work, $f_a^T v_a = f_d^T v_d$, and thus,

$$f_a = \begin{bmatrix} f_{a1} \\ f_{a2} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 1 \\ 0 & 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} f_{a1} \\ f_{a5} \\ f_{a2} \\ f_{a3} \end{bmatrix} = J_{a/a} f_d.$$  

Using this result, the parallel manipulator can be represented by the network shown in Fig. 3.14.

Instead of deriving the equations directly from this model, circuit reduction techniques are applied. Fig. 3.15 shows the progression of the network model for the manipulator as circuit reduction techniques are implemented. The reduction from Fig. 3.14 to Fig. 3.15a is allowable, since

$$M_a \begin{bmatrix} \dot{v}_{a1} \\ \dot{v}_{a2} \end{bmatrix} = \begin{bmatrix} m_{a1} + m_1 l_c^2 & 0 \\ 0 & m_{a2} + m_2 l_c^2 \end{bmatrix} \begin{bmatrix} \dot{v}_{a1} \\ \dot{v}_{a2} \end{bmatrix}.$$  

Fig. 3.13. Network model for parallel-link manipulator.
Fig. 3.14. Modified network model for parallel-link manipulator.

Fig. 3.15. Circuit reduction of parallel-link manipulator network:
   a) move inertias $M_{c1}$ and $M_{c2}$ inside actuator loop;
   b) add parallel transformers; c) multiply series transformers.
\[J_{5/a} = J_{5/a1}[1 \ 0] + J_{5/a5}[1 \ 0] \]
\[= \begin{bmatrix} -l_1S_1 & 0 \\ l_1C_1 & 0 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} -l_1S_1 & 0 \\ l_1C_1 & 0 \\ 0 & 1 \end{bmatrix},\]
and likewise,
\[J_{3/a} = J_{3/a2}[0 \ 1] + J_{3/a3}[1 \ 1] = \begin{bmatrix} 0 & -l_2S_2 \\ 0 & l_2C_2 \\ 1 & 0 \end{bmatrix}.\]

The final reduction from Fig. 3.15b to Fig. 3.15c is obtained by multiplying the two series transformers:
\[J_{5c/a} = J_{5c1}J_{5/a} = \begin{bmatrix} 1 & l_5S_2 \\ 0 & 1 -l_5C_2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -l_1S_1 & 0 \\ l_1C_1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} -l_1S_1 & l_5S_2 \\ l_1C_1 & -l_5C_2 \\ 0 & 1 \end{bmatrix},\]
\[J_{3c/a} = J_{3c1}J_{3/a} = \begin{bmatrix} 1 & -l_3S_1 \\ 0 & l_3C_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -l_2S_2 \\ 0 & l_2C_2 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} l_3S_1 & -l_2S_2 \\ l_3C_1 & l_2C_2 \\ 1 & 0 \end{bmatrix}.\]

Now the dynamic equations for the parallel manipulator can be found by applying Kirchoff's laws to the reduced network.
\[
\begin{bmatrix} f_{a1} \\ f_{a2} \end{bmatrix} = M_a \begin{bmatrix} v_{a1} \\ v_{a2} \end{bmatrix} + J_{5c/a}^T C_{5c} \frac{d}{dt} (J_{5c/a} \begin{bmatrix} v_{a1} \\ v_{a2} \end{bmatrix}) + J_{3c/a}^T C_{3c} \frac{d}{dt} (M_{3c} J_{3c/a} \begin{bmatrix} v_{a1} \\ v_{a2} \end{bmatrix})
\]
\[= \begin{bmatrix} m_{th1} + m_1 l_1^2 + m_5 l_5^2 + m_3 l_3^2 + m_{th3} \\ (m_5 l_1 l_5 - m_5 l_2 l_3) C_3 \end{bmatrix} \begin{bmatrix} m_{th1} + m_1 l_1^2 + m_5 l_5^2 + m_3 l_3^2 + m_{th3} \\ (m_5 l_1 l_5 - m_5 l_2 l_3) C_3 \end{bmatrix} \begin{bmatrix} v_{a1} \\ v_{a2} \end{bmatrix}
+ \begin{bmatrix} (m_5 l_1 l_5 - m_5 l_2 l_3) S_3 v_{a2}^2 \\ (m_5 l_1 l_5 - m_5 l_2 l_3) S_3 v_{a1}^2 \end{bmatrix},
\]

(3.36)
where $S_3$ and $C_3$ are the sine and cosine of $x_{a3} = x_{a1} - x_{a2}$. From Eq. (3.36) if $m_3l_2c_3 = m_5l_1c_5$, then everything is uncoupled and linear.

3.2 Representing Environmental Contact

Very few dynamic models for representing environmental contact exist in the robot literature. Mason and Salisbury (1985), and McClamroch and Wang (1987) consider static contact, and its constraining effects on the manipulator. Dynamic models frequently assume that the forces and motions can be uncoupled along different degrees of freedom around some operating point. The environment is then represented as either a spring-damper system (Eppinger and Seering 1986) or a more general impedance operator, $Z_e$, (Anderson 1988) along each degree of freedom. It is assumed that there exists a point in the environment for which the impedance is uncoupled, i.e., there exists a "center of impedance", a generalization of the idea of centers of mass and centers of compliance. In the dynamic case the type of contact is typically ignored, since it is assumed that the manipulator has already entered into a particular regime. Distinctions between point contact, edge contact, and surface contact are ignored, as well as the inevitable coulomb friction effects of the environment's surface.

In this section it is shown how Hilbert networks are used to describe contact behavior. The effects of nonlinear friction and nonlinear stiffnesses, as well as point, edge and surface contact are included. In addition, the linear impedance model, $Z_e$, fits naturally into the representation.

3.2.1 The linearized impedance model

The concept of a linear impedance along six-DOF is nonintuitive. Because environments are usually stiff, the slightest motion of the manipulator will cause loss of contact with the surface, a nonlinear phenomena which is not amenable to a linearizing impedance representation approach. Furthermore, it is difficult to assimilate angular impedances. Consider the manipulator shown in Fig. 3.16.

It is reasonable to represent the impedance in the $x$-direction as a stiffness, $Z_{ex}(s) = K_{ex}/s$, and the impedance in the tangential $y$-$z$ plane as a damper (assuming viscous friction), $Z_{ef}(s) = B_{ef}$. The impedance around any of the axes, however, depends on the surface features of the environment and the tool. Both the friction forces and the uneven application of pressure at the tool surface affect the moments around the tool center, and thus should play some part in the angular impedance. The position at which angular moments are uncoupled from linear forces depends on both the geometry and the dynamic characteristics of the robot and environment. This position will change relative to the manipulator as the robot
Fig. 3.16. Model of robot contact.

completes a task. In addition, the center of compliance does not, in general, correspond to the center of mass for an object. Therefore it seems unreasonable to rely on a model which uses this moving center of compliance for control.

Despite these reservations, the uncoupled impedance representation is useful conceptually and amenable to the network approach taken in this paper. Let $v_g \in H_g = \mathbb{R}^6$ represent the velocity of the manipulator’s gripper, and suppose a center of impedance exists at a distance, $d_{e/g} \in \mathbb{R}^3$, from the gripper position. Then the Jacobian,

$$J_{e/g} = \begin{bmatrix} I_3 & S(d_{e/g}) \\ 0_3 & I_3 \end{bmatrix},$$

relates the velocity of the gripper to the velocity of the environment’s impedance center, and the Jacobians,

$$J_{ex/e} = [ 1 0 0 0 0 0 ], \quad J_{ey/e} = [ 0 1 0 0 0 0 ], \quad J_{ez/e} = [ 0 0 1 0 0 0 ],$$

$$J_{e\tilde{\xi}/e} = [ 0 0 0 1 0 0 ], \quad J_{e\psi/e} = [ 0 0 0 0 1 0 ], \quad J_{e\zeta/e} = [ 0 0 0 0 0 1 ],$$

relate the motion of the center of compliance to the individual degrees of freedom. Let the impedances along the six DOF be given by $Z_{ex}$, $Z_{ey}$, $Z_{ez}$, $Z_{e\tilde{\xi}}$, $Z_{e\psi}$, and $Z_{e\zeta}$, respectively. The network diagram for a system with a center of impedance is represented by the network in Fig. 3.17. Because each of the impedances are assumed to be LTI and passive, they are realizable in terms of a network of passive elements. Furthermore, because each stiffness and inertial element can be scaled to unity by the use of a transformer, it is representable as a PHIDE network.
3.2.2 Representing point contact

Instead of assuming a center of impedance and the existence of an uncoupled representation for the environment, a more intuitive approach is taken in this section. In the previous case the tool was represented as a rigid body with 6-DOF, which has forces and moments lumped together at a single point. Now, it is assumed that the tool contacts the environment at many discrete points, and that at each point of contact only 3-DOF forces can exist. In addition, a switching mechanism is used to represent the effects of initiation and loss of contact, and a more realistic model for damping is derived.

For notational simplicity boolean expressions such as \((x > y)\), give the numeric scalar values of one for true, and zero for false. Thus the expression \(J = (x > y) \cdot I_3\) implies that

\[
J = \begin{cases} 
1 \cdot I_3 = I_3 & \text{if } x > y \\
0 \cdot I_3 = 0 & \text{if } x \leq y.
\end{cases}
\]  
(3.39)

To illustrate this approach, the case of point contact with no friction is first considered. Fig. 3.18 shows a manipulator contacting a planar surface at the point, \(x = x_e\). The tip of the tool is located a distance, \(d_{t/g} = R_{i/g}(x_g) l_{i/g}\), from the gripper, where \(l_{i/g}\) is the length from the gripper to the tool tip. It is assumed that surface has a stiffness constant, \(K_{Ne}\). The tool tip is a discrete point with three prismatic degrees of freedom and no revolute degrees of freedom. Thus the tool space, \(H_t\), used for point contact is \(\mathbb{R}^3\). The Jacobian, \(J_{t/g} : H_g \rightarrow H_t\), relating
velocities, \( v_t \), of the tool tip to velocities, \( v_g \), of the gripper is given by

\[
J_{t/g} = \begin{bmatrix} I_3 & S(d_{t/g}) \end{bmatrix}.
\]  

(3.40)

Because no friction exists tangential to the surface, only one-DOF is needed to describe the dynamics in the environment, namely the direction along the surface normal. This space is denoted by \( H_N = \mathbb{R} \). The tool tip is related to the surface normal by the Jacobian,

\[
J_{N/tl} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}.
\]  

(3.41)

Not only are spaces needed for describing the manipulator and the manipulator tool, but spaces are also needed for describing the environment. Let \( H_{Ne} \triangleq \mathbb{R} \) be the space which measures deflections in the normal direction of the environment. \( H_N \) is related to \( H_{Ne} \) by the switching Jacobian,

\[
J_{Ne/In}: H_N \rightarrow H_{Ne} = (x_N > x_e).
\]  

(3.42)

With these definitions a network model for point contact with the environment is given in Fig. 3.19. From the network model, the dynamic equations for the system are derived. In particular, the gripper force, \( f_g \in \mathbb{R}^6 \), is given by

\[
f_g = J_{t/g}^* J_{N/t}^* J_{Ne/In}^* f_{Ne} = \begin{bmatrix} I_3 \\ ST(d_{t/g}) \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} (x_N \leq x_e) f_{Ne},
\]

where \( f_{Ne} = K_{Ne} \int_0^t v_{Ne} dt \), and the normal velocity in the environment is given by

\[
v_{Ne} = J_{Ne/In} J_{N/tl} J_{t/g} v_g = (x_N \leq x_e) \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} [ I_3 \ S(d_{t/g}) ] v_g.
\]
In order to include the effects of friction tangential to the surface of the environment, some additional spaces and elements need to be defined. The tangential space for the manipulator, $\mathbf{H}_T$, and the tangential space for the environment, $\mathbf{H}_{Te}$, are both equal to $\mathbb{R}^2$. Let the Jacobian relating the tool space to the manipulator tangent space, $J_{T/T}: \mathbf{H}_t \rightarrow \mathbf{H}_T$, be given by

$$J_{T/T} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

and the switching Jacobian, $J_{Te/T}: \mathbf{H}_T \rightarrow \mathbf{H}_{Te}$, be given by

$$J_{Te/T} = (x_N > x_c) I_2.$$  

(3.43)  

(3.44)

To include the effects of coulomb and viscous friction, let $\mu_c \in [0, \infty)$ represent the coefficient of coulomb friction, and let $\mu_v \in [0, \infty)$ represent the coefficient of viscous friction for the surface. The damping element $B_{Te}$ can now be defined by

$$B_{Te} \triangleq (\mu_c \frac{\|f_{Ne}\|}{\|v_{Te}\|} + \mu_v) I_2,$$

and thus the force, $f_{Te}$, tangential to the surface is given by

$$f_{Te} = B_{Te} v_{Te} = (\mu_c \frac{\|f_{Ne}\|}{\|v_{Te}\|} + \mu_v) v_{Te}.$$

With these definitions, the network for single point contact with friction is shown in Fig. 3.20. Although $B_{Te}$ faithfully represents the effects of coulomb friction, it is not a continuous operator. To ensure that the network is mathematically tractable, the approximate damping operator, $\tilde{B}_{Te}$, defined by

$$\tilde{B}_{Te} \triangleq \mu_c \frac{\|f_{Ne}\|}{\|v_{Te}\| + \varepsilon} + \mu_v,$$

(3.45)

where $\varepsilon \in [0, \infty)$ is an arbitrarily small number, is used in its place.
By adding \( n-1 \) additional networks, the preceding model for single point contact can model \( n \)-point contact. Let \( \mathbf{H}_{ii} = \mathbb{R}^3 \) represent the tool spaces for the \( i = 1, \cdots, n \) discrete points on the tool. The manipulator normal and tangential spaces, \( \mathbf{H}_{Ni} = \mathbb{R} \) and \( \mathbf{H}_{Ti} = \mathbb{R}^2 \) respectively, and the environmental normal and tangential spaces, \( \mathbf{H}_{Nei} = \mathbb{R} \) and \( \mathbf{H}_{Tei} \) respectively, are defined accordingly. The Jacobian relating gripper and tool velocities is given by
\[
J_{ii/g} = [I_3 \ S(d_{ii/g})],
\]
where \( d_{ii/g} \) is the distance between the gripper and the tool position, \( x_i \). The Jacobians which partition the normal and tangential subspaces for \( t_i \) are given by
\[
J_{Ni/i} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix},
\]
\[
J_{Ti/i} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},
\]
respectively, as long as the surface normal is in the \( x \)-direction. If the position of the environment varies, however, these Jacobians will change accordingly. The switching Jacobians for the normal and tangential subspace are defined by
\[
J_{Nei/Ni} = (x_{ii} > x_{ei}),
\]
\[
J_{Tei/Ti} = (x_{ii} > x_{ei})I_2.
\]
The circuit representing \( n \)-point contact is given in Fig. 3.21.

Although the \( n \)-point representation shown in Fig. 3.21 is useful when contact is initiated at a few positions, it is inappropriate and far too cumbersome for representing contact at a continuum of points. To alleviate this problem distributed contact models have been developed.
3.2.3 Network models for distributed contact

The simplest form of distributed contact is edge contact. Let the tool space, \( \mathbf{H}_t \), be defined by \( \mathbf{H}_t \triangleq L_2^3[0,1] \). The edge of the tool can be described parametrically by the length, \( l_{lg} \in L_2^3[0,1] \), from the gripper position to positions along the tool edge. The Cartesian distance, \( d_{lg} \in L_2^3[0,1] \), from the tool to the edge is defined by

\[
d_{lg}(r) = R_t l_{lg}(r), \tag{3.46}
\]

where \( r \in [0,1] \). The Jacobian, \( J_{lg} : \mathbf{H}_g \to \mathbf{H}_t \), relating gripper position to tool position is given by the constituent equation,

\[
v_t(r) = J_{lg}(r)v_g = [ I_3 \ S(d_{lg}(r)) ]v_g, \tag{3.47}
\]
where \( \nu_t \in \mathbf{H}_t \) is the velocity of the tool edge.

Extending the definitions of the other spaces to include the edge representations gives the following: \( \mathbf{H}_N = \mathbf{H}_{Ne} = L^2[0,1] \); \( \mathbf{H}_T = \mathbf{H}_{Te} = L^2[0,1] \). Surprisingly, the network diagram for edge contact is identical to the diagram for single-point contact (Fig. 3.20), except that all the Jacobians and variables are defined over different spaces. The Jacobians which divide the system into normal and tangent spaces, \( J_{N/t} \) and \( J_{T/t} \), respectively, are defined by

\[
\nu_N(r) = J_{N/t} \nu_t(r) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \nu_t(r), \]

\[
\nu_T(r) = J_{T/t} \nu_t(r) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \nu_t(r) \tag{3.48}
\]

where \( \nu_N \) and \( \nu_T \) are the normal and tangential components of the tool edge velocity. The switching Jacobians, \( J_{Ne/N} \) and \( J_{Te/T} \), are defined by

\[
\nu_{Ne}(r) = J_{Ne/N} \nu_N = (\chi_N(r) > x_e) \nu_N
\]

\[
\nu_{Te}(r) = J_{Te/T} \nu_T = (\chi_N(r) > x_e) \nu_T. \tag{3.49}
\]

The damping element, \( B_{Te} : \mathbf{H}_{Te} \rightarrow \mathbf{H}_{Te} \), is defined by

\[
f_{Te}(r) = B_{Te} \nu_{Te}(r) = (\mu_c \frac{\|f_{Ne}(r)\|}{\|\nu_{Te}(r)\|} + \mu_v) \nu_{Te}(r),
\]

and the stiffness element, \( K_{Ne} : \mathbf{H}_{Ne} \rightarrow \mathbf{H}_{Ne} \), is defined by

\[
f_{Ne}(r) = K_{Ne} \int_0^1 \nu_{Ne}(r) dt.
\]

The inner product in \( \mathbf{H}_t = L^2[0,1] \) is given by

\[
\langle x, y \rangle = \left( \int_0^1 x^T(r) y(r) dr \right)^{1/2},
\]

and thus,

\[
\langle f_t, J_{t/g} \nu_g \rangle_t = \left( \int_0^1 f_t(r) \left[ I_3 \ S(d_{t/g}(r)) \right] \nu_g \ dr \right)^{1/2}
\]

\[
= \left( \int_0^1 \begin{bmatrix} I_3 \\ ST(d_{t/g}(r)) \end{bmatrix} f_t(r) \ dr \right)^{1/2} \nu_g
\]

\[
= \langle J_{t/g}^* f_t, \nu_g \rangle_g.
\]

Therefore the adjoint, \( J_{t/g}^* : \mathbf{H}_g \rightarrow \mathbf{H}_t \), is defined by

\[
f_g = \int_0^1 \begin{bmatrix} I_3 \\ ST(d_{t/g}(r)) \end{bmatrix} f_t(r) \ dr.
\]
The network representation of edge contact can be extended to multiple edges in much the same way that the single point contact representation was extended to represent multiple point contact. This could be used, for instance, to represent contact of both teeth of a simple gripper on a surface. Instead of belaboring this point, the general case of surface contact is developed next.

Assume that the surface of interest, i.e., the surface of the tool which might contact the environment, is homeomorphic to the unit square in \( \mathbb{R}^2 \), and let \( l_{tg} : [0,1] \times [0,1] \to \mathbb{R}^3 \) represent this mapping. A few examples of this parameterization are given as follows: the surface of a unit hemisphere is represented by

\[
l_{tg}(r_1, r_2) = \begin{bmatrix} \sin(\pi r_1) \cos(\pi r_2) \\ \sin(\pi r_1) \sin(\pi r_2) \\ \cos(\pi r_1) \end{bmatrix};
\]

the side of a unit cylindrical is represented by

\[
l_{tg}(r_1, r_2) = \begin{bmatrix} \cos(\pi r_1) \\ \sin(\pi r_1) \\ r_2 \end{bmatrix};
\]

the end of a unit cylinder is represented by

\[
l_{tg}(r_1, r_2) = \begin{bmatrix} r_1 \cos(\pi r_2) \\ r_1 \sin(\pi r_2) \\ 1 \end{bmatrix};
\]

and the face of a rectangular prism, of height, \( h \), and width, \( w \), is represented by

\[
l_{tg}(r_1, r_2) = \begin{bmatrix} 1 \\ h \cdot r_1 \\ w \cdot r_2 \end{bmatrix}.
\]

When the manipulator is grasping the object, the distance of each point on the surface of the object from the gripper center is represented parametrically as

\[
d_{tg}(r_1, r_2) = R_t(x_{tg}) l_{tg}(r_1, r_2).
\] (3.50)

Thus, in order to represent surface contact, \( H_t \) is defined as

\[
H_t = L_t^2([0,1] \times [0,1]),
\]

and the Jacobian, \( J_{tg} \), which relates the gripper velocity to the tool surface velocity, is defined by

\[
v_t(r_1, r_2) = [I_3 \ S(d_{tg}(r_1, r_2))] v_g,
\] (3.51)
with adjoint, \( J_{i/g}^* \), defined by

\[
f_g = \int_0^1 \int_0^1 \begin{bmatrix} I_3 \\ S_T(d_{i/g}(r_1, r_2)) \end{bmatrix} f_1(r_1, r_2) dr_1 dr_2.
\] (3.52)

As before, the network used to describe point contact and edge contact (Fig. 3.20) is equally valid for describing surface contact once the element descriptions have been extended to the appropriate spaces. These extensions can be performed in the same way as they were performed in the edge contact case, except the normal and tangent spaces are \( L_2([-1,1] \times [0,1]) \) and \( L_2^2([-1,1] \times [0,1]) \) respectively, rather than \( L_2[0,1] \) and \( L^2[0,1] \). Additional surfaces can also be included in the environment model by following the approach of Fig. 3.21.

### 3.3 Distributed Parameter Systems

The Hilbert network notation is ideal for describing distributed parameter systems. In robotics, distributed parameter systems arise in a number of places. Distributed behavior arises in the environment, since the robot may carry flexible beams, contact flexible plates, or interact with fluid systems. Distributed behavior also arises in the robot, since the the links of a robot may not be sufficiently rigid to be considered lumped elements. Due to the complexity involved in describing a manipulator with distributed links, however, to date only the simplest cases of flexible link behavior in robotics have been analyzed. Since Hilbert network representations simplify complex systems while maintaining a highly intuitive model, it is expected that more difficult models can be attempted by using the Hilbert network representation.

Distributed parameter systems can be approximated by using a finite number of lumped parameter elements. Consider the ladder network (Fig. 3.22) shown previously in Fig. 2.18. As shown in Section 2.3, this is representable in two different forms (Fig. 3.23).

![Fig. 3.22. Ladder network.](image-url)
Fig. 3.23. Ladder networks: a) \( Z_o \) form; b) \( Z_e \) form.

As the number of ladder elements approaches infinity and the magnitude of each element approaches zero, it becomes natural to redefine the spaces, \( H_o \) and \( H_e \), to the function space, \( L_2[0,1] \). All of the operators shown in Fig. 3.23 can be redefined on the new space. Table 3.1 lists these operators and their adjoints as defined for \( H_o = H_e = \mathbb{R}^3 \) and for \( H_o = H_e = L_2[0,1] \).

**Table 3.1: Redefinition of Ladder Network Jacobians.**

<table>
<thead>
<tr>
<th>Operator</th>
<th>( J_{i/j} ) on ( \mathbb{R}^3 )</th>
<th>( J_{i/j} ) on ( L_2[0,1] )</th>
<th>( J_{i/j}^* ) on ( L_2[0,1] )</th>
</tr>
</thead>
</table>
| \( J_{oii} \) | \[
\begin{pmatrix}
1 \\
0 \\
0
\end{pmatrix}
\] | \( v_o(z) = v_i \delta(z) \) | \( f_i = f_o(z) \big|_{z=0} \) |
| \( J_{eio} \) | \[
\begin{pmatrix}
1 & 0 & 0 \\
1 & 1 & 0 \\
1 & 1 & 1
\end{pmatrix}
\] | \( v_e(z) = \int_0^z v_o(z)dz \) | \( f_o(z) = \int_0^1 f_e(z)dz \) |
| \( J_{jio} \) | \[
\begin{pmatrix}
1 & 1 & 1
\end{pmatrix}
\] | \( v_j = \int_0^1 v_o(z)dz \) | \( f_o(z) = f_j \) |
| \( J_{eii} \) | \[
\begin{pmatrix}
1 & 1 & 1
\end{pmatrix}
\] | \( v_o(z) = v_i \) | \( f_i = \int_0^1 f_e(z)dz \) |
| \( J_{ole} \) | \[
\begin{pmatrix}
1 & 0 & 0 \\
-1 & 1 & 0 \\
0 & -1 & 1
\end{pmatrix}
\] | \( v_o(z) = \frac{\partial f}{\partial z} v_e(z) \) | \( f_e(z) = \frac{\partial f}{\partial z} f_o(z)dz \) |
| \( J_{jie} \) | \[
\begin{pmatrix}
0 & 0 & 1
\end{pmatrix}
\] | \( v_j = v_e(z) \big|_{z=1} \) | \( f_e(z) = f_j \delta(z-1) \) |
3.3.1 The transmission line model

Modeling a transmission line using the standard network elements provides a number of benefits. First, it demonstrates how a distributed parameter system can be represented in a finite network. Second, it provides a model for simple mechanical waves, such as those appearing on a taut string. Finally, it serves as a passive mechanism for obtaining time delay, which is useful in dealing with the time-delay problem in teleoperation.

Consider the network shown in Fig. 3.24. The network contains signals in three spaces: $H_{c1} \triangleq \mathbb{R}^n$, $H_{c2} \triangleq \mathbb{R}^n$, and $H_{c} \triangleq L^2[0,1]$. The Jacobians, $J_{c1/c}$, $J_{c2/c}$ and $J_{c}$, and their adjoints are defined below:

$$v_{1c}(z,t) = J_{c1/c}v_{1c}(t) = v_{c1}(t)$$

$$f_{c1}(t) = J_{c1/c}^* f_{c1}(z,t) = \int_0^1 f_{1c}(z,t)dz$$

$$v_{c2}(t) = J_{c2/c}v_{c}(t) = v_{c1}(1,t)$$

$$f_{c2}(z,t) = J_{c2/c}^* f_{c2}(t) = f_{c2}(t)\delta(z-1)$$

$$v_{2c}(z,t) = J_{c} \Delta v_{c}(z,t) = \frac{\partial r}{\partial z}\Delta v_{c}(z,t)$$

$$f_{1c}(z,t) = J_{c}^* f_{c}(z,t) = -\frac{\partial f_{c}}{\partial z}f_{c}(z,t).$$

The inertia element, $M_{c} : H_{c} \rightarrow H_{c}$, and the compliance element, $C_{c} : H_{c} \rightarrow H_{c}$, are constant, diagonal, $n \times n$ matrices.

Applying Kirchhoff's laws to the dynamic elements in the network results in the two equations below:

$$\frac{\partial}{\partial t}(C_{c} f_{c}) = \frac{\partial r}{\partial z}(v_{1c} - v_{c})$$

$$= -\frac{\partial}{\partial z}v_{c} - \delta(z)(v_{c1}(t) - v_{c}(z,t))$$

(3.53)
\[ \frac{\partial}{\partial t} (M_c v_c) = f_1 c - f_2 c \]
\[ = J_c^* f_c - J_c^* 2c f_{c2} \]
\[ = -\frac{\partial}{\partial z} f_c - \delta(z-1)f_{c2} \]
\[ = -\frac{\partial}{\partial z} f_c + \delta(z-1)(f_c(1) - f_{c2}). \]

(3.54)

The input force, \( f_{c1} \), and output velocity, \( v_{c2} \), are given by
\[ v_{c2} = J_{c2c} v_c = v_c(1,t) \]
\[ f_{c1} = J_{c1c} f_c = J_{c1c} \int_0^1 (-\frac{\partial}{\partial z} f_c) dz \]
\[ = \int_0^1 (-\frac{\partial}{\partial z} f_c + \delta(z-1)f_c) = f_c(0,t). \]

Thus on the interval (0,1), the standard transmission line equations are obtained:
\[ \frac{\partial}{\partial z} v_c = -C_c \frac{\partial}{\partial t} f_c, \]  
(3.55)
\[ \frac{\partial}{\partial z} f_c = -M_c \frac{\partial}{\partial t} v_c. \]  
(3.56)

Furthermore by integrating Eq. (3.53) from 0 to 0\(^+\), and integrating Eq. (3.54) from 1\(^-\) to 1\(^+\), the boundary conditions
\[ v_c(0,t) = v_c(0,t); \quad f_c(1,t) = f_{c2}(t), \]
are derived as well. Thus, the network provides a complete description of the transmission line.

The input/output equations for the system are derived by first finding the forward and reverse waves for the lossless transmission line, and then substituting for the end-points. Let the forward effort wave, \( f_c^+ \in H_c \), for the system be defined by
\[ f_c^+ = Z_c \frac{\partial}{\partial z} v_c(z,t) + f_c(z,t), \]  
(3.57)
where \( Z_c \triangleq M_c\frac{\partial}{\partial z} C_c^{-1/2} \) is the characteristic impedance for the line. Define \( T \triangleq C_c^{-1/2} M_c^{-1/2} \) to be the wave period. By differentiating Eq. (3.57) with respect to \( z \) and substituting the telegrapher's equations Eqs. (3.55) and (3.56), it follows that
\[ \frac{\partial}{\partial z} f_c^+ = Z_c \frac{\partial}{\partial z} v_c + \frac{\partial}{\partial z} f_c = -Z_c C_c \frac{\partial}{\partial t} f_c - M_c \frac{\partial}{\partial t} v_c \]
\[ -\frac{\partial}{\partial t} (T^{-1} f_c + T^{-1} Z_c v_c) = -T^{-1} \frac{\partial}{\partial t} f_c^+. \]  
(3.58)

Similarly, differentiating the negative effort wave, \( f_c^- \in H_c \), defined by
\[ f_c = Z_c f_c(z,t) - v_c(z,t), \]

leads to the equation,

\[ \frac{\partial}{\partial z} f^+_c = \frac{\partial}{\partial z} f^+_c - Z_c \frac{\partial}{\partial z} v_c \]

\[ = -M_c \frac{\partial}{\partial t} v_c + Z_c C_c \frac{\partial}{\partial t} f_c = T^{-1} \frac{\partial}{\partial t} f^- c. \]  

(3.59)

From Eqs. (3.58) and (3.59) it follows that

\[ f^+_c(z,t) = f^+_c(0, t - Tz), \]  

(3.60)

and,

\[ f^- c(z,t) = f^- c(0, t - T(1-z)). \]  

(3.61)

By using the definitions for the forward and reverse effort waves, and substituting \( z=1 \) in Eq. (3.60) and \( z=0 \) in Eq. (3.61), the input-output form of the transmission line equations is derived:

\[ v_c(1,t) = v_c(0,t-T) + Z_e^{-1} (f_c(1,t) - f_c(0,t-T)) \]

\[ f_c(0,t) = f_c(1,t-T) + Z_e (v_c(1,t) - v_c(0,t-T)) . \]

Finally, writing these equations in terms of the input and output port variables in \( \mathbf{H}_c \) and \( \mathbf{H}_2 \), respectively, gives

\[ v_c(1,t) = v_c(0,t-T) + Z_e^{-1} (f c(2,t) - f c(1,t-T)) \]  

(3.62)

\[ f c(1,t) = f c(2,t-T) + Z_e (v c(2,t) - v c(1,t-T)) . \]  

(3.63)

The total stored energy for the transmission line, \( T.E. \), is given by

\[ T.E. = \frac{1}{2} < f_c, C_c f_c >_c + \frac{1}{2} < v_c, M_c v_c >_c . \]

The total energy also equals the integral over time of the input energy minus the output energy. This is stated in the following lemma:

**Lemma 3.1:**

\[ T.E. = \int_0^t < v c1, f c1 >_c dt - \int_0^t < v c2, f c2 >_c dt \]

**Proof**

\[ T.E. = P.E. + K.E. \]

\[ = \frac{1}{2} < f c, C_c f c >_c + \frac{1}{2} < v c, M_c v c >_c \]
\[ = \int_{t_0}^{t} \langle f_c, C_c \dot{f}_c \rangle_c dt + \int_{t_0}^{t} \langle M_c \dot{v}_c \rangle_c dt \]
\[ = \int_{t_0}^{t} \langle f_c, v_{2c} \rangle_c dt + \int_{t_0}^{t} \langle v_{c1} - f_{2c} \rangle_c dt \]
\[ = \int_{t_0}^{t} \langle f_{1c}, \Delta v_c \rangle_c dt + \int_{t_0}^{t} \langle v_{1c} - \Delta v_c, f_{1c} \rangle_c dt \]
\[ - \int_{t_0}^{t} \langle v_c, f_{2c} \rangle_c dt \]
\[ = \int_{t_0}^{t} \langle v_{1c}, f_{1c} \rangle_c dt - \int_{t_0}^{t} \langle v_c, f_{2c} \rangle_c dt \]
\[ = \int_{t_0}^{t} \langle v_{c1}, f_{c1} \rangle_c dt - \int_{t_0}^{t} \langle v_{c2}, f_{c2} \rangle_c dt. \]

3.3.2 Orthonormal decompositions

When dealing with distributed parameter systems it is convenient to represent the vectors in the system with respect to an orthonormal expansion. Such an expansion redefines the underlying space of the system from a function space, such as \( L_2[0,1] \), to a series space, such as \( l_2 \). Operators defined with respect to the new space are often simpler to implement, and by truncating the series expansion, natural approximations to the system response are derived. In this subsection transformations from function spaces to series spaces will be achieved by network operations.

Let \( H_i \) be a function space defined over a bounded region, \( V \), with inner product,

\[ \langle v_i, f_i \rangle_i = \int_V v_i(z) f_i(z) W_i(z) dz, \quad (3.64) \]

where \( W_i \) is a positive definite symmetric weighting function, and suppose an orthogonal series, \( \Phi_{i;j}(n,z) \), has been found such that,

\[ \int_V \Phi_{i;j}^T(n,z) W_i(z) \Phi_{i;j}(m,z) dz = \begin{cases} I_j & \text{if } m = n \\ 0_j & \text{otherwise.} \end{cases} \quad (3.65) \]

Then consider the summation,

\[ \sum_{n=0}^{\infty} \Phi_{i;j}(n,z) v_j(n), \quad (3.66) \]

where \( v_j \in H_j \), and \( H_j \) is a series space with inner product,

\[ \langle f_j, v_j \rangle_j = \sum_{n=0}^{\infty} f_j^T(n,z) v_j(n,z). \quad (3.67) \]

If Eq. (3.66) converges uniformly to \( v_i(z) \), then \( v_j(n) \) is given by

\[ v_j(n) \Delta \int_V \Phi_{i;j}^T(n,z) W_i(z) v_i(z) dz. \quad (3.68) \]

Assuming that Eq. (3.66) is absolutely summable and Eq. (3.64) is absolutely integrable, then it
follows that,
\[ v_j(n) \triangleq \int_V \Phi_{i j}^T(n, z) W_i(z) \sum_{m=0}^\infty \Phi_{i ij}(m, z) v_j(m) \, dz \]
\[ = \sum_{m=0}^\infty \left( \int_V \Phi_{i j}^T(n, z) W_i(z) \Phi_{i ij}(m, z) \right) v_j(m) \]
\[ = v_j(n). \]  
(3.69)

The transformation from the function space, \( H_i \), to the series space, \( H_j \), given by Eq. (3.68) is represented by the transformer, \( J_{iji} \). Likewise, the inverse transformation from \( H_j \) to \( H_i \), given by Eq. (3.66) is represented by the transformer, \( J_{ij} \triangleq J_{ij}^{-1} \). From the definitions of inner products for the two spaces it follows that
\[ <f_j, v_j>_{ij} = <f_j, J_{iji} v_i>_{ij} \]
\[ = \sum_{n=0}^\sum f_j^T(n) \int_V \Phi_{i j}^T(n, z) W_i(z) v_i(z) \, dz \]
\[ = \int_V \sum_{n=0}^\sum f_j^T(n) \Phi_{i j}^T(n, z) W_i(z) v_i(z) \, dz \]
\[ = \int_V v_i(z)^T W_i(z) \left( \sum_{n=0}^\sum \Phi_{i j}^T(n, z) f_j(n) \right) \, dz \]
\[ = <v_i, J_{iji}^* f_j>_{ij}, \]  
(3.70)
and thus, \( J_{iji}^* = J_{ij}^{-1} = J_{iji} \). Similarly,
\[ <f_i, v_i>_{ii} = <f_i, J_{ij} v_j>_{ii} \]
\[ = \int_V f_i^T(z) W_i(z) \sum_{n=0}^\sum \Phi_{i ij}(n, z) v_j(n) \, dz \]
\[ = \sum_{n=0}^\sum v_j(n)^T \int_V \Phi_{i ij}(n, z) W_i(z) f_i(z) \, dz \]
\[ = <v_j, J_{ij}^* f_i>_{ij} \]  
(3.71)
and thus, \( J_{ijj}^* = J_{ij}^{-1} = J_{ijj} \). Parseval’s identity follows directly from Eqs. (3.70) and (3.71).
\[ <v_i, v_i>_{ii} = <v_i, J_{ij} v_j>_{ii} \]
\[ = <J_{ij}^* v_i, v_j>_{ij} \]
\[ = <J_{ij} v_i, v_j>_{ij} \]
\[ = <v_j, v_j>_{ij}. \]
As an example, consider the function space, \( H_i \triangleq L_2[-1,1] \), with inner product,

\[
<v_i, f_i> = \int_1^1 v_i(z)f_i(z)dz.
\]

An orthogonal series for this space is given by

\[
\Phi_{ij}(n, z) = [\sin(n \pi z) \cos(n \pi z)],
\]

and the corresponding series space, \( H_j \), has the inner product,

\[
<v_j, f_j> = 2v_j^T(0)f_j(0) + \sum_{n=1}^\infty v_j^T(n)f_j(n).
\]

The transformations, \( J_{jij} \) and \( J_{ijj} \), are defined by

\[
v_j(n) = \int_1^1 \begin{bmatrix} \sin(n \pi z) \\ \cos(n \pi z) \end{bmatrix} v_i(z)dz
\]

\[
v_i(z) = \sum_{n=0}^\infty \begin{bmatrix} \sin(n \pi z) \\ \cos(n \pi z) \end{bmatrix} v_j(n)
\]

respectively, and thus it follows that,

\[
<v_i, v_i> = \int_1^1 v_i(z)v_i(z)dz
\]

\[
= \int_1^1 \sum_{n=0}^\infty v_j^T(n) \begin{bmatrix} \sin(n \pi z) \\ \cos(n \pi z) \end{bmatrix} \sum_{m=0}^\infty \begin{bmatrix} \sin(m \pi z) \\ \cos(m \pi z) \end{bmatrix} v_j(m)dz
\]

\[
= \sum_{n=0}^\infty \sum_{m=0}^\infty v_j^T(n) \int_1^1 \begin{bmatrix} \sin(n \pi z) \\ \cos(n \pi z) \end{bmatrix} \begin{bmatrix} \sin(m \pi z) \\ \cos(m \pi z) \end{bmatrix} dz v_j(m)
\]

\[
= 2v_j^T(0)v_j(0) + \sum_{n=1}^\infty v_j^T(n)v_j(n)
\]

\[
= <v_j, v_j>.
\]

To illustrate the use of the series space representation in a network, consider a transmission line defined on the space, \( H_i = L_2[-1,1] \), with infinite input impedance and zero output impedance. The system is described by the equation,

\[
\frac{d}{dt}(M_i v_i) - \frac{\partial t}{\partial z} f^i K_i \frac{\partial r}{\partial z} v_i dt = 0
\]

where \( J_i = \frac{\partial r}{\partial z} \), \( M_i = M \), and \( K_i = K \). The system is represented by the network shown in Fig. 3.25a. The network is transformed to the space, \( H_j \), by inserting transformers, \( J_{jij} \) and \( J_{ijj} \), in cascaded pairs, as is shown in Fig. 3.25b. Since \( J_{ijj}J_{jij} = I_i \), the transformers have no net effect on the network. By pulling each of the impedances through one of the transformers and combining the remaining cascaded transformers, the network shown in Fig. 3.25c is realized.
Fig. 3.25. Function to series transformation: a) original network in $H_i$; b) network with added transformers; c) transformed network in $H_j$.

with the following definitions:

$$K_j = J_{ii}^* K_i J_{ij}$$

$$= \int V \Phi_{iij}^*(n,z)W_i(z)K_i \sum_{m=0}^{\infty} \Phi_{iij}(m,z)dz$$

$$= \int_1^1 \begin{bmatrix} \sin(n \pi z) \\ \cos(n \pi z) \end{bmatrix} K \sum_{m=0}^{\infty} \begin{bmatrix} \sin(m \pi z) \\ \cos(m \pi z) \end{bmatrix}dz$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$ if $n > 0$

$$= \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$ if $n = 0$
\[ M_j = J_{ij}^* M_j J_{ij} \]

\[
M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{if } n > 0
\]

\[
M = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{if } n = 0
\]

\[ J_j v_j = J_{ij} J_i J_{ij} v_j \]

\[
= \int_v \Phi^T_{ij}(n, z) W_i(z) J_j \sum_{n=0}^{\infty} \Phi(n, z) v_j(n) \, dz
\]

\[
= \int_1^1 \begin{bmatrix} \sin(n \pi z) \\ \cos(n \pi z) \end{bmatrix} \left( \frac{\partial c}{\partial z} + \delta(z+1) \right)
\]

\[
\cdot \sum_{m=0}^{\infty} \begin{bmatrix} \sin(m \pi z) \\ \cos(m \pi z) \end{bmatrix} \, dz \, v_j(n)
\]

\[ = \sum_{m=1}^{\infty} \int_1^1 \begin{bmatrix} \sin(m \pi z) \\ \cos(m \pi z) \end{bmatrix} (m \pi) \begin{bmatrix} \sin(m \pi z) \\ \cos(m \pi z) \end{bmatrix} \, dz \, v_j(n)
\]

\[ + \int_1^1 \begin{bmatrix} \sin(n \pi z) \\ \cos(n \pi z) \end{bmatrix} \delta(z+1) v_i(z) \, dz .
\]

Since the input impedance is infinite, \( v_i(-1) = 0 \), and the second integral can be dropped. Thus,

\[ J_j = \begin{bmatrix} n \pi & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \quad \text{if } n > 0
\]

\[ 0 = M s^2 - (n \pi)^2 \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} K \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = M s^2 + (n \pi)^2 K.
\]

From these equations it is clear that the series space representation of the constituent equations is simpler than the original functional space representation. Furthermore, a good approximation to the behavior of the system is obtained by including only the first \( n \) components of the expanded series. If only the first \( n \) modes are excited, then nothing is lost in using this truncated representation.

### 3.3.3 Beams, membranes, and Maxwell’s equations

In this subsection a few examples of network models for distributed parameter systems are developed. The intent is not to give an exhaustive survey of models for distributed
systems, but to demonstrate the wide applicability of the Hilbert network approach. For simplicity, boundary conditions are ignored. In order to include boundary conditions, impulse operators, such as were used to describe the transmission line, would be needed. This approach is thoroughly documented in (Butkovskiy 1982).

The first two systems to be investigated, a flexible membrane and a cantilever beam, represent possible environments with which a robot might interact. The third example, the realization of Maxwell’s equations in network form, has, admittedly, little to do with controlling robots. Its inclusion serves to illustrate the range of Hilbert network representations.

First consider the cantilever beam shown in Fig. 3.26a. Three different spaces are needed to describe the behavior of the beam. First, the displacement frame, \( H_d = L_2[0,l] \), is used to describe the deviation of points along the center of the beam from their rest positions. Next, the shear frame, \( H_s = L_2[0,l] \), is used to describe the shear force and angular displacements of points along the center of the beam. Finally, the bending frame, \( H_b = L_2[0,l] \), is used to describe the amount of bending in the beam as a function of position. Subnetworks in the three frames are connected by the Jacobians,

\[
J_{s/d} = J_{b/s} = -\frac{\partial}{\partial z}.
\]

Assuming that the ends of the beam are either fixed or free, i.e., no energy is transferred from the boundaries, then the adjoints for \( J_{s/d} \) and \( J_{b/s} \) are given by

\[
J_{s/d}^* = J_{b/s}^* = \frac{\partial}{\partial z},
\]

since

\[
<f_s, v_s> = <f_s, J_{s/d}v_d> = -\int_0^l f_s \frac{\partial}{\partial z} v_d dz
\]

\[
= -(f_s v_d|_0^l - \int_0^l v_d \frac{\partial}{\partial z} f_s dz)
\]

\[
= \int_0^l v_d \frac{\partial}{\partial z} f_s dz
\]

\[
= <J_{s/d}^* f_s, v_d>.
\]

The remaining parameters and variables for the cantilever beam are listed in Table 3.2. Because the notation used is nonstandard, the equivalent expressions as used in (Thomson 1981) are listed as well.
Table 3.2: Parameters and Variables for a Cantilever Beam.

<table>
<thead>
<tr>
<th>Term</th>
<th>Name</th>
<th>Notation in (Thomson 1981).</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_s$</td>
<td>rotary inertia</td>
<td>$J$</td>
</tr>
<tr>
<td>$M_d$</td>
<td>mass per length</td>
<td>$m$</td>
</tr>
<tr>
<td>$K_b$</td>
<td>bending stiffness</td>
<td>$E \cdot I$</td>
</tr>
<tr>
<td>$K_s$</td>
<td>shear stiffness</td>
<td>$k \cdot A \cdot G$</td>
</tr>
<tr>
<td>$x_d$</td>
<td>deflection of center line</td>
<td>$y$</td>
</tr>
<tr>
<td>$\Delta x_s$</td>
<td>shear deformation</td>
<td></td>
</tr>
<tr>
<td>$x_b$</td>
<td>curvature</td>
<td></td>
</tr>
<tr>
<td>$f_s$</td>
<td>shear</td>
<td>$V$</td>
</tr>
<tr>
<td>$f_b$</td>
<td>bending moment</td>
<td>$M$</td>
</tr>
<tr>
<td>$x_s$</td>
<td>slope due to bending</td>
<td>$-\psi$</td>
</tr>
<tr>
<td>$\nu_s$</td>
<td>rate of change of slope</td>
<td>$-\dot{\psi}$</td>
</tr>
<tr>
<td>$f_d$</td>
<td>applied pressure</td>
<td>$p$</td>
</tr>
</tbody>
</table>

Fig. 3.26. Cantilever beam: a) beam diagram; b) network representation.

The network representation for the beam is shown in Fig. 3.26b. The equations of motion are obtained by applying Kirchoff’s laws.

\[
J_{s/l} v_d - v_s = \frac{\partial}{\partial t} (K_s^{-1} f_s) \\
J_{b/l} v_s = \frac{\partial}{\partial t} (K_b^{-1} f_b) \\
\frac{\partial}{\partial t} (M_s v_s) = f_s - J_{b/l}^* f_b \\
f_d = \frac{\partial}{\partial t} (M_d v_d) + J_{s/l} f_s.\]
Combining these equations gives,

\[ J_{b/s}^* \left( \int_0^t K_b J_{b/s} \nu_s \right) + M_s \dot{\nu}_s - \int_0^t K_s (J_{s/id} \nu_d - \nu_s) \, dt = 0 \]

\[ f_{d/d} = M_d \ddot{\nu}_d + J_{s/id}^* \left( \int_0^t K_s (J_{s/id} \nu_d - \nu_s) \, dt \right) \]

\[ = M_d \ddot{\nu}_d + J_{s/id}^* K_s J_{s/id} \nu_d - J_{s/id}^* K_s \nu_s. \]

Finally,

\[ K_b \frac{\partial^4 x_d}{\partial z^4} + M_d \frac{\partial^2 x_d}{\partial t^2} - \left( M_s + \frac{K_b M_d}{K_s} \right) \frac{\partial^4 x_d}{\partial z^2 \partial t^2} + \frac{M_s M_d}{K_s} \frac{\partial^4 x_d}{\partial t^4} \]

\[ = f_{d/d} + \frac{M_d}{K_s} \frac{\partial^2 f_{d/d}}{\partial t^2} - \frac{K_b}{K_s} f_{d/d}. \]  \hspace{1cm} (3.72)

Equation (3.72) is known as the Timoshenko beam equation. By neglecting the shear stiffness, \((K_s^{-1} = 0)\) and the rotary inertia, \((M_s = 0)\), the simpler Euler beam equation can be obtained:

\[ f_{d/d} = M_d \ddot{\nu}_d + J_{s/id}^* \int_0^t K_b J_{b/s} J_{s/id} \nu_s \, dt \]

\[ = M_d \ddot{x}_d + \frac{\partial^2 }{\partial z^2} \left( K_b \frac{\partial^2 }{\partial z^2} x_d \right). \] \hspace{1cm} (3.73)

The second system to be described is a flexible membrane. Consider the membrane shown in Fig. 3.27a, where \(A\) is a bounded region in the \(z_1-z_2\) plane. Describing the response of the membrane requires the definition of two spaces: the membrane space, \(H_m = L_2[A]\), with inner product

\[ <v_m, f_m>_m = \int_A v_m f_m \, dz_1 dz_2; \]

and the gradient space, \(H_g = L_2^2[A]\), with inner product

\[ <v_g, f_g>_g = \int_A v_g f_g \, dz_1 dz_2. \]

The spaces are connected by the transformer, \(J_{g/im}\), which is defined by

\[ v_g = \text{grad}(v_m), \]

where \(\text{grad}\) (or \(\nabla\)) is the standard gradient operator. It is shown in Wyatt and Papadopoulos (1984), that if the signals in the two spaces are suitably differentiable, then the adjoint of the gradient operator is the negative of the divergence operator (\(\text{div}\) or \(\nabla\)), i.e.,

\[ f_m = J_{g/im}^* f_g = -\text{div}(f_g). \]

The network diagram for the system is shown in Fig. 3.27b, and the variables and parameters are listed in Table 3.3.
Fig. 3.27. Flexible membrane: a) membrane diagram; b) network representation.

Table 3.3: Parameters and Variables for a Flexible Membrane.

<table>
<thead>
<tr>
<th>Term</th>
<th>Name</th>
<th>Notation used in (Thomson 1981).</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_m$</td>
<td>membrane displacement</td>
<td>$\omega(x, y)$</td>
</tr>
<tr>
<td>$fd_m$</td>
<td>applied pressure</td>
<td>$p(x, y)$</td>
</tr>
<tr>
<td>$v_g$</td>
<td>membrane gradient</td>
<td>$(\Theta, \Phi)$</td>
</tr>
<tr>
<td>$M_m$</td>
<td>mass per unit area</td>
<td>$\rho$</td>
</tr>
<tr>
<td>$K_g$</td>
<td>tension</td>
<td>$T$</td>
</tr>
</tbody>
</table>

The equations of motion for this system are given by

$$fd_m = \frac{\partial}{\partial t}(M_m v_m) + \int_{t_0}^{t} J_{g/m} J_{g/m} v_m dt$$

$$= M_m \dot{v}_m - \text{div}(K_g \text{grad}(x_m))$$

$$= M_m \ddot{x}_m - K_g \nabla^2 x_m.$$  \hspace{1cm} (3.74)

The final system to be represented in this subsection is an electromagnetic field in a closed volume, $V$. The field dynamics are described by Maxwell’s equations. The network realization (Fig. 3.28) requires the definition of four spaces: the magnetic field space, $H_B = L^2_2[V]$; the electric field space, $H_E = L^2_2[V]$; the magnetic field divergence space, $H_b = L^2_2[V]$; and the electric field divergence space, $H_e = L^2_2[V]$. These spaces are connected by the three transformers, $J_{B/b}, J_{E/E}$, and $J_{E/B}$, which are defined below:

$$J_{B/b} = \text{grad}(\cdot), \quad J_{B/b}^* = -\text{div}(\cdot),$$
\[ J_{elE} = -\text{div}(\cdot), \quad J_{elE}^* = \text{grad}(\cdot), \]
\[ J_{E/B} = \text{curl}(\cdot), \quad J_{E/B}^* = \text{curl}(\cdot). \]

The fact that the curl operator is self-adjoint is shown in (Wyatt and Papadopoulos 1984).

![Diagram](image)

Fig. 3.28. Realization of Maxwell’s equations.

The parameters and variables used in the network and their more common representation as obtained from (Dearholt and McSpadden 1973), are listed in Table 3.4. Maxwell’s equations, and the other important laws which govern the behavior of the system are found by direct application of Kirchoff’s laws to the network shown in Fig. 3.28. These are summarized in Table 3.5.

Table 3.4: Parameters and Variables for the Electromagnetic Field Equations.

<table>
<thead>
<tr>
<th>Term</th>
<th>Name</th>
<th>Notation used in (Dearholt ’73).</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_B )</td>
<td>permeability</td>
<td>( \mu )</td>
</tr>
<tr>
<td>( C_E )</td>
<td>permittivity</td>
<td>( \varepsilon )</td>
</tr>
<tr>
<td>( G_E )</td>
<td>conductivity</td>
<td>( \sigma )</td>
</tr>
<tr>
<td>( v_B )</td>
<td>magnetic field intensity</td>
<td>( H )</td>
</tr>
<tr>
<td>( v_E )</td>
<td>current density</td>
<td>( J )</td>
</tr>
<tr>
<td>( \Delta x_E )</td>
<td>electric flux density</td>
<td>( D )</td>
</tr>
<tr>
<td>( f_E )</td>
<td>electric field intensity</td>
<td>( E )</td>
</tr>
<tr>
<td>( x_e )</td>
<td>free charge density</td>
<td>( \rho )</td>
</tr>
<tr>
<td>( \int f_B dt )</td>
<td>magnetic flux density</td>
<td>( B )</td>
</tr>
</tbody>
</table>
### Table 3.5: Maxwell’s Equations.

<table>
<thead>
<tr>
<th>Law</th>
<th>Name</th>
<th>Law in (Dearholt, ’73 notation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( v_E = G_E f_E )</td>
<td>ohm</td>
<td>( J = \sigma E )</td>
</tr>
<tr>
<td>( \Delta x_E = C_E f_E )</td>
<td>permittivity</td>
<td>( D = \varepsilon E )</td>
</tr>
<tr>
<td>( \int f_B dt = M_B v_B )</td>
<td>permeability</td>
<td>( B = \mu H )</td>
</tr>
<tr>
<td>( J_{e/E} \Delta x_E = x_e )</td>
<td>Gauss</td>
<td>( \nabla \cdot D = \rho )</td>
</tr>
<tr>
<td>( J_B^{*} f_B = 0 )</td>
<td>Gauss</td>
<td>( \nabla \cdot B = 0 )</td>
</tr>
<tr>
<td>( f_B = -J_{e/E}^{*} f_E )</td>
<td>Faraday</td>
<td>( \frac{\partial B}{\partial t} = -\nabla \times E )</td>
</tr>
<tr>
<td>( J_{e/B} v_B = v_E + \Delta v_E )</td>
<td>Ampere</td>
<td>( \nabla \times H = J + \frac{\partial D}{\partial t} )</td>
</tr>
<tr>
<td>( -J_{e/E} v_E = v_e )</td>
<td>continuity</td>
<td>( \nabla \cdot J = -\frac{\partial \rho}{\partial t} )</td>
</tr>
</tbody>
</table>

The examples in this section have served to illustrate how distributed parameter systems are described by Hilbert networks. By using function spaces it is possible to represent distributed systems with a finite number of basic elements. Furthermore, many of the concepts and techniques used for lumped parameter systems can be extended to the distributed systems. For example, potential and kinetic energy are defined in equivalent fashions for lumped and distributed systems, Lyapunov functions based on total energy can be applied for both lumped and distributed systems, and transformers can be used to transform elements into new spaces and to represent coupling interactions between other systems.

#### 3.4 Representing Nonlinear Terms in PHIDE Form

In order to represent a system in a PHIDE network it is necessary to find spaces in which all the inertia terms and all of the stiffness terms appear constant and linear. In the case of a positive definite inertia term, \( M_j \), if the conservation of energy condition is satisfied, then either \( \langle v_j, M_j v_j \rangle_j = 0 \), or there exists a fictitious damping term, \( D_j \), in series with the inertia term such that,

\[
\langle v_j, (\frac{1}{2} M_j + D_j) v_j \rangle_j = 0.
\]  

(3.75)

In this case, it is always possible to represent the inertia term in terms of an identity valued inertia in series with a skew-symmetric damping term. Consider the transformer, \( J_{jii} \Delta M_j^{-\frac{1}{2}} \). Using the transformation laws in Subsection 2.3.1, the equivalent inertia and fictitious damping elements in the space, \( H_i \), are given by

\[
M_i = J_{jii}^* M_j J_{jii} = M_j^{-\frac{1}{2}} M_j M_j^{-\frac{1}{2}} = I_j = I_i
\]
\[ D_i = J_{ji}^* D_i J_{ji}^* - \frac{d}{dt} (J_{ji}^* M_j J_{ji}) = M_j^{-\frac{1}{2}} D_j M_j^{-\frac{1}{2}} - \frac{d}{dt} (M_j^{-\frac{1}{2}}) M_j^{\frac{1}{2}} \]

The fact that \( D_i \) is skew-symmetric is proven below.

**Theorem 3.1** If \( \langle v_j, (\frac{1}{2} \dot{M}_j + D_j) v_j \rangle > j = 0 \), where \( M_j \) is positive definite and symmetric, then the damping term,

\[ D_i \triangleq M_j^{-\frac{1}{2}} D_j M_j^{-\frac{1}{2}} - \frac{d}{dt} (M_j^{-\frac{1}{2}}) M_j^{\frac{1}{2}}, \]

obtained from pulling \( M_j \) and \( D_j \) through the transformer, \( J_{ji} \triangleq M_j^{-\frac{1}{2}} \), is skew-symmetric.

**Proof**

\[ \langle v_i, D_i v_i \rangle > i = \langle v_i, M_j^{-\frac{1}{2}} D_j M_j^{-\frac{1}{2}} v_i \rangle > i - \langle v_i, \frac{d}{dt} (M_j^{-\frac{1}{2}}) M_j^{\frac{1}{2}} v_i \rangle > i \]

Since, \( v_j = M_j^{-\frac{1}{2}} v_i \), \( (M_j^{-\frac{1}{2}})^* = M_j^{-\frac{1}{2}} \), and

\[ \frac{d}{dt} (M_j^{-\frac{1}{2}}) = -M_j^{-\frac{1}{2}} d/dt (M_j^{\frac{1}{2}}) M_j^{-\frac{1}{2}}, \]

this simplifies to

\[ \langle v_i, D_i v_i \rangle > i = \langle v_j, D_j v_j \rangle > j + \langle v_i, M_j^{-\frac{1}{2}} \frac{d}{dt} (M_j^{\frac{1}{2}}) M_j^{-\frac{1}{2}} M_j^{\frac{1}{2}} v_i \rangle > i \]

\[ = \langle v_j, D_j v_j \rangle > j + \langle v_j, \frac{d}{dt} (M_j^{\frac{1}{2}}) M_j^{\frac{1}{2}} v_j \rangle > j . \]

But,

\[ \frac{1}{2} \langle v_j, \dot{M}_j v_j \rangle > j = \frac{1}{2} \langle v_j, \frac{d}{dt} (M_j^{\frac{1}{2}}) M_j^{\frac{1}{2}} v_j \rangle > j + \langle v_j, M_j^{\frac{1}{2}} \frac{d}{dt} (M_j^{\frac{1}{2}}) v_j \rangle > j \]

\[ = \langle v_j, \frac{d}{dt} (M_j^{\frac{1}{2}}) M_j^{\frac{1}{2}} v_j \rangle > j \]

and thus by assumption,

\[ \langle v_i, D_i v_i \rangle > i = \langle v_j, (\frac{1}{2} \dot{M}_j + D_j) v_j \rangle > j = 0, \]

and by the definition of skew-symmetry, \( D_i \) is skew-symmetric. □

From this theorem it is clear that inertia terms which satisfy Eq. (3.75) can always be put into PHIDE form.

The compliance term, \( C_i \), has a dual relationship to the inertia term, \( M_j \). Instead of requiring that the compliance is conservative, suppose it satisfies the stricter condition,

\[ \int_\infty^t \langle f_i, \left( \frac{d}{dt} (C_i f_i) + G_i f_i \right) \rangle > i \, dt = \frac{1}{2} \langle f_i, C_i f_i \rangle > i . \]  \hspace{1cm} (3.76)
Then, the dual of Theorem 3.1, is given by the theorem below:

**Theorem 3.2** The equality,

\[ \frac{1}{2} \langle f_i, C_i f_i \rangle_i = \int_0^1 \langle f_i, (\frac{d}{dt} (C_i f_i) + G_i f_i) \rangle_i dt \]

implies that \( \langle f_i, (\frac{d}{dt} \dot{C}_i + G_i) f_i \rangle = 0 \).

**Proof** Identical to proof for Thm. 3.1. \( \square \)

Applying a transformer with Jacobian, \( J_{ji} \triangleq C_i^{-1/2} \) results in a PHIDE network with compliance,

\[ C_j = J_{ji} C_i J_{ji}^* = C_i^{-1/2} C_i C_i^{-1/2} = I_i = I_j, \]

and skew-symmetric conductance,

\[ G_j = J_{ji} G_i J_{ji}^* - J_{ji} G_i J_{ji}^* = C_i^{-1/2} G_i C_i^{-1/2} - \frac{d}{dt} (C_i^{-1/2}) C_i^{1/2}. \]

The skew-symmetry of \( G_j \) can be proved by the same approach used in Theorem 3.1.

Unfortunately, although the condition Eq. (3.75) is reasonable to expect from an inertial term, it is not reasonable to require that Eq. (3.76) holds true for a stiffness term. To illustrate this, consider the simple nonlinearity in \( H_i = \mathbb{R} \), given by \( f_i = x_i^3 \). Here, \( C_i = 1/x_i^2 \), and the energy stored,

\[ \int_0^1 \langle f_i, \frac{d}{dt} (C_i f_i) \rangle_i dt = \int_0^1 \langle x_i^3, x_i^4 \rangle_i = \frac{1}{2} x_i^4, \]

is not equal to the quadratic form,

\[ \frac{1}{2} \langle f_i, C_i f_i \rangle = \frac{1}{2} \langle x_i^3, \frac{1}{x_i^2} x_i^3 \rangle = \frac{1}{2} x_i^4. \]

In general the condition Eq. (3.76) will be true only on the space, \( H_i = \mathbb{R} \), if the system is linear and \( C_i \) is constant.

Despite this apparent setback, it is still possible to represent nonlinear stiffness terms in PHIDE form, although it will require using higher-dimensional spaces. One example of a nonlinear representation for a stiffness element has already been given. Let \( H_i = \mathbb{R} \), and define \( H_j = \mathbb{R} \). Consider the contact nonlinearity response shown in Fig. 3.29a. This can be realized (Fig. 3.29d) by using the Boolean valued Jacobian function, \( J_{ji} \), defined by

\[ v_j = (x_i > a_i) v_i \quad f_i = (x_i > a_i) f_j \]

in cascade with a constant, linear stiffness, \( K_j \). To achieve more sophisticated nonlinearities, the dimension of the space, \( H_j \), needs to be increased. Consider the piecewise linear response shown in Fig. 3.29b. By defining \( H_j \triangleq \mathbb{R}^2 \), and setting
Fig. 3.29. Increasing stiffness responses: a) contact nonlinearity; b) piece-wise linear nonlinearity; c) continuously increasing nonlinearity; d) realization in network form.

\[ J_{jli} = \begin{bmatrix} (x_i > a_{i1}) \\ (x_i > a_{i2}) \end{bmatrix} \]

and \( K_j = \text{diag}(K_{j1}, K_{j2}) \), the nonlinearity is realized in the network shown in Fig. 3.29d. In general, any nonlinearity with \( n \) piecewise linear segments and a monotonically nondecreasing slope can be realized by the network shown in Fig. 3.29d, by setting \( H_j = \mathbb{R}^n \), and defining \( J_{jli} \) and \( K_j \) appropriately.

In order to realize a continuously varying function, \( f_i = \alpha(x_i) \), with a positive monotonically nondecreasing slope, such as shown in Fig. 3.29c, it is necessary to use an infinite dimensional space for \( H_j \). Suppose that the function continuously varies only over a bounded region, after which the slope is constant, and assume for simplicity that that bound is equal to one. Set \( H_j = L_2[0,1] \), define \( J_{jli} \) by

\[ v_j(z) = (x_i > z)v_i \]

\[ f_i = \int_0^1 (x_i > z)f_j(z)dz = \int_0^x f_j(z)dz \]

and define \( K_j \) by

\[ K_j(z) = \frac{\partial^2}{\partial z^2} \alpha(z). \]

Then, for \( x_i < 1 \),

\[ f_i = \int_0^{x_i} K(z) \int_0^x (x_i > z)v_i dtdz \]

\[ = \int_0^{x_i} K(z)(x_i - z)dz \]
\[ \begin{align*}
\frac{\partial}{\partial z} \alpha(z) x_i & \big|_{\delta_i} - \frac{\partial}{\partial z} (z) \delta_i \big|_{\delta_i} + \int_{0}^{x_i} \frac{\partial}{\partial z} (z) dz \\
& = \alpha(x_i),
\end{align*} \]

and for \( x_i \geq 1, \)

\[ f_i = \int_{0}^{1} K(z) v_i dt dz \\
= \frac{\partial}{\partial z} \alpha(1) x_i - \frac{\partial}{\partial z} \alpha(1) + \alpha(1) \]

and Fig. 3.29d serves as a realization for one type of a continuously varying nonlinear stiffness. Since it was assumed that the function \( \alpha(x_i) \) has a positive monotonically nondecreasing slope, it follows that \( K_j(z) \) is nonnegative for all \( z \in [0,1] \), and thus by using a transformer with the value, \( J_j(z) = K(z)^{1/2} \), it is possible to represent the stiffness in a PHIDE network.

\[ \begin{align*}
a) & \quad x_i \quad C_j + C_i \quad f_i \quad C_i \quad b_1 \\
b) & \quad x_i \quad C_j + C_i \quad f_i \quad C_i \quad b_1 \quad b_2 \\
c) & \quad x_i \quad \beta_{ij} \quad C_i \quad f_i \\
d) & \quad v_i \quad f_j \quad C_j \quad \vdots \quad \vdots \quad J_{ij} \quad C_i
\end{align*} \]

Fig. 3.30. Increasing compliance responses: a) single switching; b) piece-wise linear nonlinearity; c) continuously increasing nonlinearity; d) realization in network form.

The network realization shown in Fig. 3.29d can only represent systems with positive, monotonically nondecreasing slopes. In order to represent systems with positive, monotonically nonincreasing slopes, another representation (based on compliances rather than stiffnesses) is needed. Consider the nonlinearity shown in Fig. 3.30a, where the deflection, \( x_i \), is seen as a function of the applied force, \( f_i \). This can be realized by the network shown in Fig. 3.30d, where the Jacobian, \( J_{i/j} \), is defined by

\[ v_i = (f_i > b_i) v_j, \quad f_j = (f_i > b_i) f_i, \]

and the compliance, \( C_j \), is constant. Adding additional piecewise branches requires setting \( H_j \) to a larger space. For instance, to obtain the response shown in Fig. 3.30b, it is necessary to set \( H_j = \mathbb{R}^2 \), define \( J_{i/j} \) by
\[ v_i = \left[ (f_i > b_{11}) \right] v_{i} f_j = \left[ \begin{array}{c} (f_i > b_{11}) \\ (f_i > b_{12}) \end{array} \right], \]

and set \( C_j = \text{diag} (C_{j_1}, C_{j_2}) \). The extension to a continuously varying nonlinearity (Fig. 3.30c), \( x_i = \beta(f_i) \), requires setting \( H_j = L_2[0,1] \) (assuming the region of continuous variation is \([0,1]\)), defining \( J_{j_{\ell i}} \) by

\[ v_i = \int_0^1 (f_i > z) v_j(z) dz = \int_0^{f_i} v_j(z) dz, \quad f_j = (f_i > z) f_i \]

and defining \( C_j \) by

\[ C_j(z) = \frac{\partial^2}{\partial z^2} \beta(z). \]

Intuitively, stiffnesses added in series increase the slope, and stiffnesses added in parallel decrease the slope. Therefore, to obtain a nonlinear response which has a positive slope but is neither monotonically nondecreasing nor nonincreasing, both parallel and series stiffness elements need to be used. Consider the piecewise-linear, positive sloped response shown in Fig. 3.31a, which can be realized using the ladder network shown in Fig. 3.31c. By applying the reduction techniques of Section 2.3.4, the network shown in Fig. 3.31d can be obtained. For the nonlinearity shown in Fig. 3.31a, the spaces \( H_\ell \) and \( H_e \) are set to \( \mathbb{R}^2 \), and the operators in Fig. 3.31d are defined below:

\[
J_{e\ell i} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad J_e = \text{diag} \left( f_i > b_2, f_i > b_4 \right), \]
\[
J_{o\ell e} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad J_o = \text{diag} \left( x_i > a_1, x_i > a_2 \right), \]
\[
C_e = \text{diag} \left( C_2, C_4 \right), \quad K_o = \text{diag} \left( K_1, K_3 \right). \]

The slope as a function of position is obtained by computing

\[ K_i = J^*_{e\ell i} \left( C_e^\dagger + (J^*_{o\ell e} K_o^\dagger J_{o\ell e})^{-1} \right)^{-1} J_{e\ell i} \]

where

\[ C_e^\dagger = \text{diag} \left( C_2^\dagger, C_4^\dagger \right) \triangleq J_e C_e J_e^* = \text{diag} \left( (f_i > b_2) C_2, (f_i > b_4) C_4 \right) \]

\[ K_o^\dagger = \text{diag} \left( K_1^\dagger, K_3^\dagger \right) \triangleq J_o^* K_o J_o = \text{diag} \left( (x_i > a_1) K_1, (x_i > a_3) K_3 \right). \]

Since,

\[ (J^*_{o\ell e} K_o^\dagger J_{o\ell e})^{-1} = \begin{bmatrix} K_1^\dagger & K_1^\dagger \\ K_1^\dagger & K_1^\dagger + K_3^\dagger \end{bmatrix}^{-1} \begin{bmatrix} C_1^\dagger + C_3^\dagger & -C_3^\dagger \\ -C_3^\dagger & C_3^\dagger \end{bmatrix} \]

it follows that
\[
K_i = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} C_1 & +C_2 & +C_3 & -C_4 \\ -C_2 & C_3 & +C_4 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
= \frac{K_1 + K_2 + K_3 C_2}{(1 + K_1 C_2)(1 + K_3 C_4) + K_1 C_4}.
\]

Fig. 3.31. Positive slope nonlinearities: a) piecewise linear nonlinearity; b) continuously varying nonlinearity; c) ladder network realization; d) reduced ladder network realization.

In order to model a nonlinear stiffness which has a slope that varies continuously over a bounded region, such as is shown in Fig. 3.31b, infinite dimensional spaces are needed. Assume, for simplicity, that the region of continuous variation is the unit square. Set \( H = H_o \mathbb{A} L_2[0,1] \), and using the network shown in Fig. 3.31d, define the Jacobians,

\[
J_{e/\ell} = \delta(z-1), \quad J_e = (f_i > z), \\
J_{o/\ell} = \int_0^1 (\cdot) \, dz, \quad J_o = (x_i > z).
\]

Because the stiffness nonlinearity is positively sloped, there exists a functional relationship between \( f_i \) and \( x_i \) which is invertible, i.e., one can write \( f_i = \alpha(x_i) \), or \( x_i = \beta(f_i) \). Define the stiffness, \( K_o \), by
\[
K_o = \begin{cases} \\
\frac{\partial^2}{\partial z^2} \alpha(z) & \text{if } \frac{\partial^2}{\partial z^2} \alpha(z) > 0 \\
0 & \text{otherwise,} 
\end{cases}
\]

and define the compliance, \( C_e \), by
\[
C_e = \begin{cases} \\
\frac{\partial^2}{\partial z^2} \beta(z) & \text{if } \frac{\partial^2}{\partial z^2} \beta(z) > 0 \\
0 & \text{otherwise.} 
\end{cases}
\]

With these definitions the network shown in Fig. 3.31d represents the nonlinearity of Fig. 3.31b. Furthermore, by cascading transformers with the values, \( K_o^{\frac{1}{2}} \) and \( C_e^{\frac{1}{2}} \), it is simple to convert this network to PHIDE form.

A transformer with the \( \text{sgn}(x_i) \) function as a Jacobian can be used to reflect first quadrant nonlinearities into the third quadrant. The \( \text{sgn} \) function is defined by
\[
\text{sgn}(x_i) = \begin{cases} \\
1 & \text{if } x_i > 0 \\
0 & \text{if } x_i = 0 \\
-1 & \text{if } x_i < 0 
\end{cases}
\]

and thus if \( v_j = \text{sgn}(x_i)v_i \), it follows that \( x_j = |x_i| \). Two examples of the use of the \( \text{sgn} \) function are shown in the figures below. Fig. 3.32 shows the realization of a deadzone nonlinearity, and Fig. 3.33 shows the realization of a saturation nonlinearity.

![Diagram](image)

Fig. 3.32. Deadzone nonlinearity: a) plot; b) realization.

In Chapter 4, it is assumed that the PHIDE network representation can represent all passive environment behavior that a manipulator is likely to encounter. The examples in this chapter give credence to this assertion.
Fig. 3.33. Saturation nonlinearity: a) plot; b) realization.
4. FORCE CONTROL PROBLEMS AND STABILITY

In the previous chapter, it was shown that the manipulator's dynamics are given by

\[ fc_a = \frac{d}{dt}(M_a v_a) + D_a v_a + J_{g/a}^T f_g \]

\[ = M_a \dot{v}_a + C_a v_a + J_{g/a}^T f_g, \]  

(4.1)

where \( M_a \) is the manipulator inertia in joint space, \( D_a \) is the fictitious damping, \( C_a \) is the coriolis and centripetal force matrix, \( J_{g/a} \) is the manipulator Jacobian relating gripper and actuator spaces, \( f_g \) is the gripper force vector, and \( v_a \) is the joint velocity vector. These equations can be represented in standard network form, as shown in Fig. 4.1.

![Network representation of the manipulator.](image)

The field of force control is concerned with determining robot motor commands for tasks which involve environmental contact. In particular, the force control problem determines the choice of the actuator force signal, \( fc_a \), so that control objectives, such as position tracking, force tracking, robustness, and stability, are achieved. It is assumed that noiseless measurements of the manipulator joint velocity, \( v_a \), the manipulator joint position, \( x_a \), and the gripper force, \( f_g \), are obtainable. The gripper position, \( x_g \), is found by solving the forward kinematics. The difficulty arises from the nonlinear nature of the operators, \( M_a, C_a, J_{g/a} \), and the fact that the gripper force, \( f_g \), is in general an unknown function of the manipulator velocity and position, which consists of switching phenomena (e.g., contact/noncontact at a surface), distributed behavior, and nonlinearities (e.g., coulomb friction).

The network models and operator notation introduced so far are useful for understanding robot force control. In this chapter, a number of existing robot control algorithms, including a new passive implementation of hybrid control are introduced. Next, computed torque approaches are investigated in detail. It is shown how existing computed torque algorithms are nonpassive and potentially unstable upon contact, and how a passive computed torque algorithm is implemented. Then, an entirely new approach to computed torque, which uses a
variable bandwidth but maintains a constant damping ratio, is introduced. Finally, a stability theorem for passively controlled manipulators in contact with arbitrary environments is given. The proof uses the composite stiffness and inertia vectors to obtain a Lyapunov function, which is then shown to be negative semidefinite using Tellegen’s theorem. Additional arguments are applied to prove the asymptotic stability of manipulator velocities.

4.1 Network Representations of Robot Control Algorithms

A number of approaches have been applied to robot force control, with such names as "stiffness control," "damping control," "impedance control," and "hybrid position/force control." In this section many of these force control architectures are introduced and realized in network form.

4.1.1 Joint space, gripper space, and task space

Before the network representations can be given, a thorough understanding of the three manipulator spaces, $H_a$, $H_g$, and $H_t$ is necessary. These spaces were introduced in Chapter 3, and are summarized here for completeness.

The space, $H_a$, is called either actuator space or joint space. As its name would imply, it refers to the vectors of actuator torques and joint velocities. Thus, the controlled source, $f_c$, is the vector of torques commanded by the manipulator’s motors, and the flow signal, $v_a$, is the vector of joint velocities that can be either measured directly by joint tachometers or measured indirectly by differentiating the joint position, $x_a$. Optical encoder disks attached either to the motor rotor shaft, or directly to the robot link, are used to measure $x_a$. The dimension of the space, $H_a$, is equal to the total number of actuator degrees of freedom. A typical manipulator consists of a series connection of $n$ rigid links, each actuated by a single DOF dc-motor, and therefore, $H_a \cong \mathbb{R}^n$. Conceptually however, $H_a$ could include multi-DOF actuators or distributed actuators, and could be defined on a much larger space.

Control laws computed in joint space are the easiest to implement. They require no computation of either manipulator Jacobians or inertias. Velocities are measured directly, and forces are commanded directly. The determination of a position trajectory, $\mathbf{u}_a$, is obtained either by the use of a teach pendant, or by solving the inverse kinematics problem off-line.

The space, $H_g$, is called either gripper space or world space. It is a fixed Cartesian space with respect to an inertial reference frame. Typically, $H_g \cong \mathbb{R}^6$, corresponding to the 6-DOF, a rigid body. When motion is constrained to lie in a plane, $H_g \cong \mathbb{R}^3$, since a rigid body can only move linearly in two orthogonal directions in the plane or rotate around an axis orthogonal to
the plane.

Trajectories are most easily understood in gripper space, since no knowledge of manipulator kinematics is required. A user typically has no concern as to the position of a manipulator, as long as the gripper is positioned and oriented correctly. Sophisticated teach pendants allow commands to be input in gripper space, since as anyone who has operated a teach pendant in joint space can attest, commanding a robot to move joint by joint is a slow and painful process.

The final space, \( H_t \), is known as task space, and will be used here in the strict sense (i.e., the distributed contact models introduced in Section 3.2 will not be used). The task space is a 6-DOF Cartesian space like \( H_g \), but instead of being fixed, it is oriented with respect to the manipulator task. For instance, if the task involves motion across the surface of a workpiece, then the task space is oriented so that one axis is normal to the workpiece surface, and two axes are located tangential to the surface. If the surface is curved, then as the manipulator progresses across the surface, the space, \( H_t \), will rotate with respect to the fixed \( H_g \).

Computing control laws in task space offers special problems that do not occur in either joint space or gripper space. For instance, no correspondence exists between gripper position and task space position. This occurs since a workpiece surface provides a nonholonomic constraint to the manipulator. Velocities are obtained through the gripper to task space Jacobian, \( J_{tg} \), by the relationship

\[
v_t = J_{tg} v_g. \tag{4.2}
\]

Positions, however, are obtained only by integrating the velocity signal,

\[
x_t = \int_{-\infty}^{t} J_{tg} v_g dt. \tag{4.3}
\]

By integrating by parts, the equation above can be rewritten in terms of the position, \( x_g \), and an addition integral term. If \( J_{tg} \) consists solely of rotation matrices, as is often the case, then the integral term can be further simplified, since \( \dot{J}_{tg} = S_t J_{tg} \), where \( S_t \) is skew-symmetric. This development is shown below.

\[
x_t = J_{tg} x_g - \int_{-\infty}^{t} J_{tg} x_g dt = J_{tg} x_g - \int_{-\infty}^{t} S_t J_{tg} x_g dt. \tag{4.4}
\]

In spite of the difficulty in obtaining position errors, task space descriptions are useful for describing contact tasks, and are used to develop the hybrid control approach.

### 4.1.2 Position control

The simplest form of robot control is proportional-derivative (PD) control in joint space. In this case the commanded torque signal is given as
\[ f_{c_a} = K_a (x_d - x_a) - B_a v_a, \]  

where \( K_a \) and \( B_a \) are constant, diagonal matrices, and \( x_d = \int_{t_0}^{t} v_d \, dt \) is a commanded position trajectory. A network representation of this control is shown in Fig. 4.2.

![Fig. 4.2 PD control network.](image)

A slight variation which gives greater tracking (i.e., zero-error to a ramp input when no bias forces exist and inertia is constant) is the PD control with velocity feedforward. This requires the commanded force torque signal to be given as

\[ f_{c_a} = K_a (x_d - x_a) + B_a (\dot{w}_a - v_a). \]

The network representation for this control structure is shown in Fig. 4.3.

![Fig. 4.3 PD control with velocity feedforward network.](image)

Proportional-Integral-Derivative (PID) control is commonly used in position controls to overcome steady-state bias errors, which for the manipulator are due primarily to gravity loading. This approach will not be used here for two reasons. First, if integral action is used on a manipulator which contacts a hard surface prematurely, the applied torques will ramp up in an attempt to close the position gap, possibly damaging the work surface or the manipulator. For this reason, PID controls are rarely used when significant external forces are expected. Second, the PID control does not maintain the passivity of the network, thus making it possible for the
manipulator to become unstable when contacting passive environments, and making it unamenable to the passive network approach taken in this paper.

Instead, in order to overcome the bias error due to gravity loading, direct gravity compensation will be used. Thus, when the gravity vector, \( g_a \), is included in the dynamics, i.e.,

\[
 f_{ca} = M_a \dot{v}_a + C_a v_a + J^T_{g/a} f_g + g_a,
\]

then the computed force term compensates, i.e.,

\[
 f_{ca} = f^c_{ca} + g_a,
\]

which reduces to the standard force equation:

\[
 f^c_{ca} = M_a \dot{v}_a + C_a v_a + J^T_{g/a} f_g.
\]

For this reason, the gravity will not be included in either the dynamics representation nor the controller representation, with the assumption that when gravity does affect the dynamics, then it is directly compensated for in the controller.

### 4.1.3 Force control and the use of force feedback

Until now the term "force control" has been used to describe the general problem of controlling a manipulator in tasks in which the dynamics of the environment significantly affect the performance of the task. This term can also mean, in a strict sense, the case when a manipulator is servoed based on a force trajectory, rather than a position trajectory. The simplest form of force control is

\[
 f_{ca} = f_{da} - B_a v_a,
\]

where \( f_{da} \) is an independent effort source denoting the force trajectory, and \( B_a \) is a constant damping matrix. Because of the high gearing and large inertias of the manipulator, however, the control law Eq. (4.7) is ineffective at transmitting forces to the end effector. To overcome the large impedance of the manipulator, feedback of the measured gripper force, \( f_g \), is applied. The new force control law is then given by

\[
 f_{ca} = (1+\alpha)(f_{da} - B_a v_a) - \alpha J^T_{g/a} f_g,
\]

where \( \alpha \in \mathbb{R} \) is the force feedback constant. Setting this equal to the manipulator equation Eq. (4.1) gives

\[
 f_{da} = \frac{d}{dt} \left[ \frac{M_a}{1+\alpha} v_a \right] + \frac{D_a}{1+\alpha} v_a + B_a v_a + J^T_{g/a} f_g.
\]

The controlled system can be represented in network form as shown in Fig. 4.4.
4.1.4 Computations in gripper space: impedance control

In Subsection 3.1.3.2, it was shown that the equivalent compliance as seen from the manipulator’s gripper, assuming PD feedback is used at each joint, is given by

\[ C_g = J_{g/a} C_a J_{g/a}^T. \]

This is created by pulling the actuator space compliance, \( C_a = K_a^{-1} \), into the gripper space. Unfortunately, because \( J_{g/a} \) depends functionally on the position of the manipulator, the compliance, \( C_g \), also varies as a function of manipulator position. In many force control tasks, however, the gripper space compliance should remain constant, so that, independent of the manipulator gripper position, a given applied force at the tip will always result in the same deflection. To achieve this, it is necessary to coordinate the joint torques. Consider the stiffness control law (Salisbury 1980):

\[
f_{c_a} = J_{g/a}^T \int_0^t K_g J_{g/a} (\omega_d - v_a) dt - B_a v_a
= J_{g/a}^T (K_g (\omega_d - x_g)) - B_a v_a,
\]  

where \( K_g \) and \( B_a \) are constant matrices. This control law is represented by the network shown in Fig. 4.5a. The equivalent compliance, as seen from the gripper space, is given by

\[ C_g = J_{g/a} (J_{g/a}^T K_g J_{g/a})^{-1} J_{g/a}^T, \]

and if \( J_{g/a} \) is invertible, then \( C_g = K_g^{-1} \), which is constant.

It is also possible to obtain constant gripper space damping or higher order filtering behavior by computing all of the controller dynamics in gripper space. This approach has been
given the name "impedance control" (Hogan 1985) and is shown in Fig. 4.5b, where the control signal is given by

\[ f_{ca} = -J_{g/a}^T (Z_{1g} v_g + Z_{2g} (v_g - w_d)) , \tag{4.11} \]

and \( Z_{1g} \) and \( Z_{2g} \) are impedance operators. The most common implementation of impedance control is nothing more than PD control implemented in task space, where \( Z_{1g} = 0 \) and \( Z_{2g} = B_g + K_g/s \).

It is also possible to compute a force trajectory in gripper space. In this case, the control signal is given by

\[ f_{ca} = J_{g/a}^T (f_d - Z_{1g} v_g) , \tag{4.12} \]

where \( Z_{1g} \) is used to provide damping and filtering.

Force feedback can be used for both force and position control in gripper space. Impedance control with force feedback is given by,

\[ f_{ca} = -(1+\alpha)(J_{g/a}^T (Z_{1g} v_g + Z_{2g} (v_g - w_d))) - \alpha J_{g/a} f_g . \tag{4.13} \]

Although it is possible to obtain a controller impedance which appears constant in gripper space by computing the controller in gripper space, it is not possible to make the manipulator's inertia appear constant in gripper space, unless the gripper acceleration is known. Suppose the gripper acceleration, \( \ddot{v}_g \), is known, and a positive definite inertia term, \( M_{1g} \), is found, such that

\[ M_g + M_{1g} = M_{2g} , \tag{4.14} \]

where \( M_{2g} \) is constant. For instance, let \( M_{1g} \triangleq 2 \| M_g \| I_g - M_g \). In this case, the passive control law,
\[ f_{ca} = -J_{g/\alpha}^T (M_1 \dot{v}_g + Z_1 v_g + Z_2 (v_g - \omega_g)) , \quad (4.15) \]

maintains a constant controller and manipulator impedance in gripper space. Although intriguing, the use of acceleration feedback will not be pursued further in this work.

### 4.1.5 Hybrid control

In most cases involving environmental interaction, neither pure position control nor pure force control is desirable. The term hybrid position/force control, or hybrid control for short, describes a robot control algorithm which uses both position error servoing and force error servoing. In implementations, the task space, \( \mathbb{H}_t \), is decomposed into two spaces, a force controlled subspace, \( \mathbb{H}_f \), and a position controlled subspace, \( \mathbb{H}_p \). The task space is decomposed using the partition of unity operators, \( P_{f/\mu} \) and \( P_{p/\mu} \), so that

\[
\begin{bmatrix}
v_p \\
v_f \\
\end{bmatrix} = \begin{bmatrix} P_{p/\mu} \\
P_{f/\mu} \\
\end{bmatrix} \nu_t .
\quad (4.16)
\]

From \( P_{p/\mu} \) and \( P_{f/\mu} \), the selection matrices, \( \Omega_p \) and \( \Omega_f \), are defined below:

\[
\Omega_p = P_{p/\mu}^* P_{p/\mu} ,
\quad (4.17)
\]

\[
\Omega_f = P_{f/\mu}^* P_{f/\mu} .
\quad (4.18)
\]

For example, if \( \mathbb{H}_t = \mathbb{R}^3 \) and the first two components are position controlled, then

\[
P_{p/\mu} = \begin{bmatrix} 1 & 0 \\
0 & 1 \\
\end{bmatrix} , \quad P_{f/\mu} = \begin{bmatrix} 0 & 0 & 1 \\
0 & 1 & 0 \\
\end{bmatrix} ,
\]

\[
\Omega_p = \begin{bmatrix} 1 & 0 \\
0 & 1 \\
0 & 1 \\
\end{bmatrix} = \text{diag} (1, 1, 0),
\]

and

\[
\Omega_f = \begin{bmatrix} 0 \\
0 & 0 & 1 \\
1 \\
\end{bmatrix} = \text{diag} (0, 0, 1).
\]

From the definitions of the partition operators, it follows that

\[
\Omega_p + \Omega_f = P_{p/\mu}^* P_{p/\mu} + P_{f/\mu}^* P_{f/\mu} = (P_{p/\mu}^* + P_{f/\mu}^*) (P_{p/\mu} + P_{f/\mu}) = I_t .
\]

If the selection matrices are transformed to gripper space using the unitary transformation, \( J_{g/\mu} \), so that

\[
\Omega_p = J_{g/\mu}^* \Omega_p J_{g/\mu} ,
\]

\[
\Omega_f = J_{g/\mu}^* \Omega_f J_{g/\mu} ,
\]


then it still holds true that
\[ \Omega_p g + \Omega_f g = I_g. \]

A number of different Hybrid position/force control laws have been proposed since their first appearance in 1981 (Raibert and Craig 1981). The original hybrid law was written
\[
f_{ca} = B_a J_{i/d}^T \Omega_{p_i} J_{i/g} (\nu_{d_g} - v_g) + K_a J_{i/d}^T \Omega_{p_i} J_{i/g} (\nu_{d_g} - x_g) \\
+ J_{i/a}^T \Omega_f f_{d_i} + \alpha J_{i/a}^T \Omega_f (f_{d_i} - f_i).
\]

Asada and Slotine (1986) suggest the hybrid law
\[
f_{ca} = B_a J_{i/d}^T \Omega_{p_i} J_{i/g} (\nu_{d_g} - v_g) + K_a J_{i/d}^T \Omega_{p_i} J_{i/g} (\nu_{d_g} - x_g) \\
+ \alpha J_{i/a}^T \Omega_f (f_{d_i} - f_i).
\]
The task space version of hybrid control is given by
\[
f_{ca} = J_{i/a}^T (\Omega_{p_i} (B_t J_{i/g} (\nu_{d_g} - v_g) + K_t J_{i/g} (\nu_{d_g} - x_g)) \\
+ \Omega_f (f_{d_i} + \alpha(f_{d_i} - f_i))).
\]

All of these approaches have been implemented with some success, and reasonably decompose the task space into the force controlled and position controlled subspaces. In general, however, none of these approaches preserve the passivity of the system. Furthermore, there are many reasons that passivity is lost. First, the operation, \( \Omega_{p_i} B_t \) (or similarly \( \Omega_{p_i} K_t \)), can preserve the passivity of the damping term only in the special case that \( \Omega_{p_i} \) and \( B_t \) commute. This will occur if \( B_t \) is block diagonal, with no coupling terms between position and force directions. Second, the term, \( \Omega_{f} f_{i} = \Omega_{f} Z_t v_t \), will skew the contact impedance, \( Z_t \), unless the contact impedance is block diagonal. Third, when the gripper-to-task space Jacobian, \( J_{i/g} \), evolves over time, it is no longer true that
\[ \nu_{d_g} - x_t = J_{i/g} (\nu_{d_g} - x_g). \]
The stiffness operator, \( K_t \), however, is initially computed in task space. Maintaining passivity requires the addition of a fictitious damping term, which arises when the desired stiffness, \( K_t \), is moved to the gripper space frame, \( H_g \).

Developing a hybrid control law which overcomes all these difficulties requires a number of changes. First, instead of using the selection-matrices, \( \Omega_f \) and \( \Omega_p \), the partitions of identity, \( P_{f/it} \) and \( P_{p/it} \), are used directly. This allows more freedom in choosing nondiagonal damping matrices, since \( P_{f/it} B_t P_{p/it} \) is always semipositive definite. Second, rather than applying force-feedback in the force-controlled direction only, it is applied equally in all directions. This prevents the task contact impedance from appearing nonpassive to the manipulator, and
prevents damage when the manipulator unexpectedly collides with an object in a nonforce
controlled direction. In any case, the additional force feedback does not significantly alter the con-
control signal, since forces in position controlled directions are typically negligible. Finally, to
account for a rotating task frame, an additional corrective term is computed which involves the
integral of a skew-symmetric projection of the task space error.

In task space, the new control law is given by

\[ f_{ca} = (1+\alpha)J_{g/i}^T J_{l/i/g}^T P_{p/i}^T (B_p (\omega_p - v_p) + \int_0^t K_p (\omega_p - v_p) \, dt) \]

\[ + (1+\alpha)J_{g/i}^T J_{l/i/g}^T P_{p/i}^T (f_{df} - B_f v_f) - \alpha J_{g/i}^T a_f g \]

\[ = (1+\alpha)J_{g/i}^T (B_{1g} (\omega_g - v_g) + J_{p/i/g}^T K_p \int_0^t J_{p/i/g} (\omega_g - v_g) \, dt) \]

\[ + (1+\alpha)J_{g/i}^T (J_{f/i/g} f_{df} - B_{2g} v_g) - \alpha J_{g/i}^T a_f g, \quad (4.22) \]

where

\[ B_{1g} = J_{p/i/g}^T B_p J_{p/i/g}, \quad B_{2g} = J_{f/i/g}^T B_f J_{f/i/g}, \]

\[ J_{p/i/g} = P_{p/i} J_{i/i/g}, \text{ and } J_{f/i/g} = P_{f/i} J_{i/i/g}. \]

Integrating the second term in (4.22) by parts and using

\[ \dot{J}_{p/i/g} = P_{p/i} \dot{J}_{i/i/g} = P_{p/i} S_t J_{i/i/g}, \]

where \( S_t \) is skew-symmetric, gives

\[ f_{ca} = J_{g/i}^T (B_{1g} (\omega_g - v_g) + K_g (\omega_g - \omega_{1g}) - J_{p/i}^T K_p P_{p/i} \int_0^t S_t J_{i/i/g} (\omega_g - \omega_{1g}) \, dt) \]

\[ + (1+\alpha)J_{g/i}^T (J_{f/i/g} f_{df} - B_{2g} v_g) - \alpha J_{g/i}^T a_f g, \]

where \( K_g = J_{p/i/g} K_p J_{p/i/g} \). The network diagram for this control law is shown in Fig. 4.6.

Fig. 4.6 Passive hybrid controller implementation.
4.2 Computed Torque Approaches

Ideally, a manipulator should respond to input commands with critically damped outputs. This guarantees that a manipulator will neither overshoot its target, nor will it be unduly sluggish in reaching its final destination. Unfortunately, the inertia matrix for the system can vary widely for different manipulator configurations. For constant valued stiffness and damping matrices, the degree of damping changes constantly. To prevent underdamped responses at full manipulator extension, a constant damping matrix must be set to a large value, even though this results in a system which is overdamped for almost all other configurations.

The solution to this problem is to vary the stiffness and damping in harmony with the inertia, so that the resulting system is critically damped at all configurations. This approach is called "computed torque."

4.2.1 Standard computed torque

The standard form of the computed torque law (Luh, et al. 1980) for the dynamic system Eq. (4.1) is given by

\[ f_{\dot{c}_a} = \dot{M}_a \dot{c}_a + \dot{C}_a v_a, \]  

(4.23)

where \( \dot{M}_a \) and \( \dot{C}_a \) are estimates of the manipulator inertia matrix and coriolis and centripetal force matrix. The term, \( f_{\dot{c}_a} \), is an "outer loop" torque, which for position control takes the form

\[ f_{\dot{c}_a} = B_o (\dot{w}_a - v_a) + K_o (\dot{x}_a - x_a), \]  

(4.24)

where \( B_o \) and \( K_o \) are constant damping and stiffness matrices. If \( \dot{M}_a = M_a \), and \( \dot{C}_a = C_a \), then the system reduces to

\[ \dot{v}_a + B_o v_a + K_o x_a = B_o \dot{w}_a + K_o \dot{x}_a + M_a^{-1} J_g^T a \dot{f}_g. \]

For a manipulator interacting with an environment there are two main problems with this computed torque algorithm. First, the control law Eqs. (4.23) and (4.24) is, in general, not a passive control law. The consequences of this nonpassivity are discussed in the next subsection.

The second problem is that the apparent compliance as seen from the gripper depends functionally on the gripper position. A small force, \( \delta f_g \), applied at the end effector results in the deflection, \( \delta x_a \triangleq K_o^{-1} M_a^{-1} J_g^T a \delta f_g \). Since \( \delta x_g = J_{g/a} \delta x_a \), it follows that the gripper compliance is given by

\[ C_g = J_{g/a} K_o^{-1} M_a^{-1} J_g^T, \]  

(4.25)
which varies with the manipulator inertia, \( M_a \), and the manipulator Jacobian, \( J_{g/a} \). Applying the same analysis to the damping and inertia terms, and assuming the invertibility of \( J_{g/a} \), results in the following equivalent damping and inertia terms in gripper space:

\[
M_g = J_{g/a}^{-1} M_a J_{g/a}^{-1}; \quad B_g = J_{g/a}^{-1} M_a B_o J_{g/a}^{-1}.
\]

(4.26)

As these equations illustrate, when computed torque is computed in joint space, all of the impedance terms vary as a function of manipulator position.

Computing the dynamics and inverse dynamics in gripper space (Khatib 1987) does little to improve this situation. The dynamic equation for the system in gripper space -- which requires invertibility of the manipulator Jacobian, \( J_{g/a} \), and is obtained by pulling the inertia term, \( M_a \), through the transformer, \( J_{g/a} \) -- is given by

\[
f_c = J_{g/a}^T (\frac{d}{dt}(M_g \nu_g) + D_g \nu_g + f_g) = J_{g/a}^T (M_g \nu_g + C_g \nu_g + f_g),
\]

(4.27)

where \( M_g \triangleq J_{g/a}^{-1} M_a J_{g/a}^{-1} \), and \( D_g \triangleq J_{g/a}^{-1} D_a J_{g/a}^{-1} - \frac{d}{dt}(J_{g/a}^{-1})M_a J_{g/a}^{-1} \). The inverse dynamics controller in task space is given by

\[
f_c = J_{g/a}^T \dot{M}_g (B_1 (\nu_{g} - \nu_{g}) + K_1 (\alpha_{d} - x_{g})).
\]

(4.28)

From Eqs. (4.27) and (4.28), the equivalent gripper stiffness and damping terms can be computed:

\[
C_g = K_1 M_g^{-1}; \quad B_g = M_g^{-1} B_1.
\]

(4.29)

Both the joint space form and the gripper space form of computed torque cause the gripper impedance terms to be configuration dependent. Furthermore, as is evident from the equations Eqs. (4.25) and (4.29), the gripper impedances are no longer necessarily positive definite. The consequences of this nonpositive definiteness are related to the nonpassivity of the computed torque controller, and are investigated in the next section.

4.2.2 The instability problem with standard computed torque

The computed torque equations Eqs. (4.23) and (4.24) can be represented in network form by using a two-port amplifier, \( A_a = \dot{M}_a^{1/4} \), and a transformer, \( J_{a/o} = \dot{M}_a^{-1/4} \), as shown in Fig. 4.7.

Because the system requires a nonscalar valued amplifier element, passivity for the system is generally lost. Only in the special case that \( \dot{M}_a^{1/4} \) commutes with \( M_a \) and no contact force, \( f_g \), is applied, can passivity be maintained. In this case pulling the inertia term, \( M_a \), through the amplifier and the transformer results in the transformed inertia,

\[
M_o = J_{o/a}^T A_a^{-1} M_a A_a J_{o/a}
\]
which is positive definite. Realistically, \( \hat{\mathbf{M}}_a^{1/2} \) will only commute with \( \mathbf{M}_a \) if a perfect estimate is made, \( \hat{\mathbf{M}}_a = \mathbf{M}_a \), or no estimate is made, \( \hat{\mathbf{M}}_a = \alpha \mathbf{I}_a \). In general, not only will the estimate be inexact, but a contact force will be applied, and passivity is lost. This loss of passivity can lead to disastrous consequences, as illustrated in the following example.

Suppose the computed torque scheme of equation Eqs. (4.23) and (4.24) is used on a constant, gravityless, two-DOF, linear system with inertia matrix,

\[
\mathbf{M}_a = \begin{bmatrix}
10 & 3 \\
3 & 1
\end{bmatrix}.
\]

Let the stiffness and damping matrices be chosen as

\[
\mathbf{B}_o = \begin{bmatrix}
2 & 2 \\
2 & 3
\end{bmatrix}, \quad \mathbf{K}_o = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}.
\]

Now, suppose that the diagonal of the inertia matrix is used as an estimate, i.e.,

\[
\hat{\mathbf{M}}_a = \begin{bmatrix}
10 & 0 \\
0 & 1
\end{bmatrix}.
\]

The dynamic equation for the system with \( \mathbf{v}_a = \mathbf{x}_a = 0 \) is

\[
\dot{\mathbf{v}}_a = -\mathbf{M}_a^{-1} \hat{\mathbf{M}}_a (\mathbf{B}_o \mathbf{v}_a + \mathbf{K}_o \mathbf{x}_a)
\]

\[
= \begin{bmatrix}
1 & -3 \\
-3 & 10
\end{bmatrix} \begin{bmatrix}
10 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
2 & 2 \\
2 & 3
\end{bmatrix} \mathbf{v}_a + \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix} \mathbf{x}_a
\]

\[
= \begin{bmatrix}
14 & 11 \\
-40 & -30
\end{bmatrix} \mathbf{v}_a + \begin{bmatrix}
10 & -3 \\
-30 & 10
\end{bmatrix} \mathbf{x}_a,
\]

which leads to the characteristic equation

\[
s^4 - 16s^4 + 740s^2 + 350s + 10 = 0.
\]
Thus, the system has two unstable poles, in spite of the fact that positive definite stiffness and damping matrices were used and that a seemingly reasonable estimate for the inertia was chosen.

Researchers have recognized the problem of instability for computed torque, and controller architectures based on variable structure and Lyapunov theory have been designed to overcome the controller/plant mismatch (Spong, et al. 1987). By obtaining an upper bound for the deviation between the estimated force and the actual force, it is possible to design a robust position controller. These approaches, however, do not adequately address the problems of environmental contact, since no upper bound for force deviation can be obtained for an arbitrarily stiff environment.

Robust, stable behavior of a computed torque algorithm is possible by using a passivity approach without requiring additional controller loops. If a poor estimate of the inertia is used, only the performance characteristics of the system will degrade; the stability will remain intact. The design of such passive computed torque controllers is the subject of the next few sections.

4.2.3 Passive implementation of computed torque

The standard computed torque approach is nonpassive and is potentially unstable, because it requires the use of a nonscalar valued gain element. To avoid the use of this element, it is necessary to compute the square root of the manipulator inertia, $M_a^{1/2}$. In this section, a passive implementation of the standard computed torque approach is used which utilizes $M_a^{1/2}$ as a transformer element.

As was shown in Section 3.4, the manipulator inertia can be represented in PHIDE form by using the transformer $M_a^{1/2}$. This corresponds to the dynamic equation,

$$f_{ca} = M_a^{1/2} \frac{d}{dt} (M_a^{1/2} v_a) + D_{1a} v_a + J_{g/a} f_g$$

$$= M_a^{1/2} \frac{d}{dt} (I_m v_m) + D_{1a} v_a + J_{g/a} f_g,$$  \hspace{1cm} (4.30)

where $v_m \triangleq M_a^{1/2} v_a$. The term, $D_{1a}$, is obtained by first pulling $D_a$ and $M_a$ into the new space, $H_m$, resulting in the term,

$$D_m = M_a^{-1/2} D_a M_a^{1/2} - \frac{d}{dt} (M_a^{-1/2})^T I_m M_a^{-1/2}.$$ \hspace{1cm} (4.31)

As was shown in Theorem 3.1, $D_m$ is skew-symmetric. The damping term in $H_a$ is obtained by pulling $D_m$ through the transformer, $M_a^{1/2}$, to get

$$D_{1a} = M_a^{1/2} D_m M_a^{1/2},$$  \hspace{1cm} (4.32)
which is also skew-symmetric.

A passive computed torque controller for this system can be obtained by closing the position and velocity feedback loops in the space, $\mathbf{H}_m$, i.e., the space in which the manipulator inertia appears constant. This entails using a transformer, $\hat{M}_a^{1/2}$, to couple constant valued stiffness and damping elements, $K_m$ and $B_m$, to the network. The skew-symmetric damping term, $D_1$, can be offset by using a skew-symmetric series damping term with value, $D_1^T$. The resulting system is shown in Fig. 4.8.

![Diagram of passive computed torque network](image)

**Fig. 4.8.** Passive computed torque network.

The control signal for the implementation is given by

$$f_{ca} = \hat{M}_a^{1/2}B_m \hat{M}_a^{1/2}(v_d - v_a) + \hat{M}_a^{1/2}\int_0^t K_m \hat{M}_a^{1/2}(v_d - v_a) \, dt - D_1^T v_a$$

$$= \hat{M}_a^{1/2}(B_m (v_d - v_m) + K_m (x_d - x_m)) - D_1^T v_a.$$  \hspace{1cm} (4.33)

If the inertia and damping estimates are known exactly, then equating the dynamic equation (4.30) with the controller equation (4.33) gives

$$B_m v_d + K_m x_d = \dot{v}_m + B_m v_m + K_m x_m + M_a^{-1/2} f_{g/a} f_g.$$  \hspace{1cm} (4.34)

The choice $B_m = 2K_m^{1/2}$ will then result in a critically damped system. On the other hand, if the estimate is not exact, the system is still guaranteed to be stable and passive.

Unfortunately, there is an implementation difficulty in using the control law equation (4.33). The position error, $x_d - x_m$, can not be obtained directly, but only through integration of the velocity signal,

$$x_d - x_m = \int_0^t v_d - v_m \, dt = \int_0^t \hat{M}_a^{1/2}(v_d - v_a).$$  \hspace{1cm} (4.35)

To convert the control law equation (4.33) from a form which uses the position error indirectly
to a form which uses it directly, requires the network equivalent of pulling the stiffness element, \(K_m\), through the time-varying transformer, \(\dot{\dot{M}}_a\). As was shown in Subsection 2.3.1, this results in the addition of a "fictitious compliance" element, \(G_a\). Implementing this requires the use of a dynamic controller. The equations are given below:

\[
f_{c,a} = B_a (v_{d,a} - v_a) + f_1 a - D^a T^a v_a, \tag{4.36}
\]

\[
f_1 a = K_a (x_{d,a} - x_a - x_1 a), \tag{4.37}
\]

\[
x_1 a = \int_0^t G_a f_1 a \, dt, \tag{4.38}
\]

where \(G_a \triangleq -\dot{\dot{M}}_a^{-1} M_a^{-1} \dot{M}_a^{1/2}, K_a \triangleq \dot{\dot{M}}_a^{1/2} K_m \dot{M}_a^{1/2}, \) and \(B_a \triangleq \dot{\dot{M}}_a^{1/2} B_m M_a^{1/2}\).

The equivalent compliance in gripper space, \(C_g\), is given by,

\[
C_g = J_{g/a} (M_a^{1/2} K_m M_a^{1/2})^{-1} J_{g/a}' \tag{4.39}
\]

As in the case with the standard form of computed torque, this compliance will vary with configuration. Because it is necessarily positive semidefinite, however, it cannot lead to a loss of stability.

### 4.2.4 Achieving critical damping in a PHIDE network

In order to motivate a new, variable bandwidth approach to computed torque, a simple "MBK" (Mass-Damping-Stiffness) network in PHIDE form (Fig. 4.9) will be investigated. Let the vector, \(v_k\), represent the composite stiffness velocity, and let \(v_m\) represent the composite inertia velocity. The Jacobians, \(J_{k/b}\) and \(J_{m/b}\), couple the identity valued stiffness and inertia elements in the spaces, \(H_k\) and \(H_m\), to the damping space, \(H_b\). The dynamic equation for the system is

\[
J_{m/b} \frac{d}{dt} (J_{m/b} v_b) + B_b v_b + J_{k/b}' \int_0^t J_{k/b} v_b \, dt = 0. \tag{4.40}
\]

**Definition 4.1** If \(H_k = H_m = H_b\) and \(B_b \triangleq J_{k/b}' J_{m/b} + J_{m/b}' J_{k/b}\) are positive definite, then the system shown in Fig. 4.9 is critically damped.

For a linear time invariant system, this definition is equivalent to the standard definition of critical damping, since in the frequency domain the dynamic equation can be written

\[
0 = (J_{m/b}' J_{m/b} s^2 + (J_{k/b}' J_{m/b} + J_{m/b}' J_{k/b}) s + J_{k/b}' \int_0^t J_{k/b} \, dt) x_b(s)
\]

\[
= (J_{m/b}' s + J_{k/b}')(s J_{m/b} + J_{k/b}) x_b(s),
\]

and the poles of the system appear in pairs along the real axis at points where \(s J_{m/b} + J_{k/b}\) drops rank. Because \(B_b\) is positive definite, it follows that \(J_{m/b}' J_{k/b}\) is positive definite, and
Fig. 4.9 MBK network in PHIDE form.

thus,

\[ J_{k/b} J_{m/b} = J_{m/b} J_{k/b} J_{m/b} J_{k/b} J_{m/b} \]

is positive definite and the poles of the system are in the left-hand plane.

Critically damped LTI systems obtain the highest level of bandwidth possible without incurring overshoot, and thus represent an ideal response for manipulators. For systems which are not LTI but have slowly varying terms, such as the manipulator matrix and the manipulator Jacobian, the system should behave in approximately the same fashion.

The requirement that \( B_b \) is positive definite is not unduly restrictive, since once a PHIDE representation is obtained, as shown in Fig. 4.9, there always exists an equivalent representation involving transformers, \( J_{k/b} U_k J_{k/b} \) and \( J_{m/b} U_m J_{m/b} \), where \( U_k \) and \( U_m \) are unitary matrices, such that \( B_b \Delta J_{k/b} J_{m/b} + J_{m/b} J_{k/b} \) is positive definite. This is stated in the following theorem.

**Theorem 4.1** Let Fig. 4.9 represent a MBK PHIDE network, in which \( H_m = H_k = H_b \), and assume that the connecting transformers, \( J_{k/b} \) and \( J_{m/b} \), are nonsingular. Then, it is always possible to find an equivalent representation with connecting transformers, \( J_{k/b} \) and \( J_{m/b} \), such that \( B_b \Delta J_{k/b} J_{m/b} + J_{m/b} J_{k/b} \) is positive definite.

**Proof** Consider the singular value decomposition,

\[ J_{k/b} J_{m/b} = U_b \Sigma_b V_b \],

where \( \Sigma_b = \text{diag}(\sigma_{b1}, \cdots, \sigma_{bn}) \) is the diagonal matrix of singular values. Define the unitary matrices, \( U_k = U_b \), and \( U_m = \text{sgn}(\Sigma_b) V_b \), where,

\[ \text{sgn}(\Sigma_b) \triangleq \text{diag}(\text{sgn}(\sigma_{b1}), \cdots, \text{sgn}(\sigma_{bn})). \]
and set

\[ J_{k/l}^* = U_b J_{k/l}, \quad J_{m/l}^* = U_m J_{m/l}. \]  

(4.42)

With these definitions, \( B_b^* \) is given by

\[ B_b^* = J_{k/l}^* J_{m/l}^* + J_{m/l}^* J_{k/l}^* \]
\[ = J_{k/l}^* U_b^T U_m J_{m/l} + J_{m/l}^* U_m^T U_b J_{k/l} \]
\[ = J_{m/l}^* (J_{m/l}^* J_{k/l}^* U_b^T U_m + U_m^T U_b J_{k/l} J_{m/l}) J_{m/l} \]
\[ = J_{m/l}^* (V_b^T \Sigma_b U_b U_b^T \text{sgn}(\Sigma_b) V_b + \text{sgn}(\Sigma_b) U_b U_b^T \Sigma_b V_b) J_{m/l} \]
\[ = 2 J_{m/l}^* V_b^T \Sigma_b \text{sgn}(\Sigma_b) V_b J_{m/l}, \]  

(4.43)

and is positive definite, since \( J_{m/l} \) is nonsingular, \( V_b \) is unitary, and \( \Sigma_b \text{sgn}(\Sigma_b) = \text{diag} (|\sigma_1|, \ldots, |\sigma_n|) \) is positive definite. \( \square \)

Underdamped and overdamped systems can be obtained by defining the damping operator,

\[ B_b = \zeta (J_{k/l}^* J_{m/l} + J_{m/l}^* J_{k/l}), \]  

(4.44)

where \( \zeta \) is the damping ratio. This is a direct extension of the use of \( \zeta \) in second-order systems for the scalar case, where the characteristic equation can be written

\[ s^2 + 2 \zeta \omega_0 + \omega_0^2 = 0, \]

and the characteristic frequency (bandwidth), \( \omega_0 \), is given by \( \omega_0 = (K/M)^{1/2} = J_{k/l} J_{m/l} \). The system is critically damped if \( \zeta = 1 \), is underdamped if \( \zeta < 0 \), and is overdamped if \( \zeta > 0 \).

By using the network shown in Fig. 4.9 as a model, it is possible to create passive representations of computed torque laws in joint space and gripper space. For instance, the constant bandwidth computed torque controller of the previous subsection can be represented by the network shown in Fig. 4.9 by setting \( H_b = H_a \), \( J_{m/l} = M_a^{1/2} \), and \( J_{k/l} = K_m^{1/2} M_a^{1/2} \). Thus, it follows that the critical damping controller has been chosen as

\[ B_a = J_{k/l}^* J_{m/l} + J_{m/l}^* J_{k/l} \]
\[ = M_a^{1/2} K_m^{1/2} M_a^{1/2} + M_a^{1/2} K_m^{1/2} M_a^{1/2} \]
\[ = 2 M_a^{1/2} K_m^{1/2} M_a^{1/2}. \]

In the next subsection, two new computed torque controllers are developed which use the same network, but different values, for \( J_{k/l} \).
4.2.5 Variable frequency/constant stiffness computed torque

The motivation for using computed torque is to obtain constant damping levels and constant bandwidth for all configurations of the manipulator. One would expect, however, that a manipulator would have a lower bandwidth at configurations where the inertia is large, such as at full extension, and to have a higher bandwidth at configurations where the inertia is small, such as at full compression. To control the manipulator to behave differently, as is done with the standard computed torque approach, fights the fundamental nature of the robot. The use of the standard computed torque formulation also ignores the limitations of the manipulator actuators. Maintaining a constant bandwidth when the inertia is large will likely lead to torque saturation, and maintaining a constant bandwidth when the inertia is small under utilizes the actuators. Under utilization is a problem, since carrying the extra weight of the higher torque actuator reduces the payload capacity of the manipulator.

By using the approach of the previous subsection, it is possible to obtain a bandwidth which varies with configuration, but maintains a constant level of damping. Thus, overshoot is prevented and maximum usage from the actuators is obtained. In this section, two approaches to a variable bandwidth computed torque method are introduced. The first approach is most useful for pure position control, since it computes more easily, uses the actuator position errors, and follows the changing manipulator inertia. The second approach is most useful for force control, since it maintains a constant level of gripper compliance, something that no other computed torque approach has been able to achieve. Both approaches are implemented passively, and thus, as is shown in Section 4.3, ensure the stability of the control algorithm when in contact with arbitrary environments. Furthermore, the implementation difficulties (which occurred when the standard, constant bandwidth computed torque algorithms were implemented passively) do not plague these variable bandwidth computed torque approaches.

Consider the two systems shown in Fig. 4.10. Both represent PHIDE network representations of MBK circuits, which correspond to a manipulator under PD control. The system shown in Fig. 4.10a is obtained by the control law,

\[ f_c = B_a (\omega_a - \nu_a) + K_d^{\frac{1}{2}} \int_0^t K_d^{\frac{1}{2}} (\omega_a - \nu_a) dt - D\tilde{I}_d \nu_a \]

\[ = B_a (\omega_a - \nu_a) + K_a (\omega_a - x_a) - D\tilde{I}_d \nu_a. \]  

(4.45)

If the damping term, \( B_a \), is computed using the techniques shown in Subsection 4.2.4, i.e.,

\[ B_a = \zeta (J_{k/b} U_k^T U_m J_{m/b} + J_{m/b} U_m^T U_m J_{k/b}) \]

\[ = \zeta (K_d^{\frac{1}{2}} U_k^T \text{sgn}(\Sigma_b) V_b M_d^{\frac{1}{2}} + M_d^{\frac{1}{2}} V_k^T \text{sgn}(\Sigma_b) U_b K_d^{\frac{1}{2}}), \]  

(4.46)
Fig. 4.10 Variable bandwidth computed torque networks: a) actuator space controller, b) gripper space controller.

where

$$U_b^T \Sigma_b V_b = J_{k/b} J_{m/b}^{-1} = K_a^{1/2} M_a^{-1/2}, \quad (4.47)$$

then a prescribed level of damping, $\zeta$, is reached for the manipulator for all configurations. If $K_a$ is scalar valued, then the computation of $B_a$ is considerably simplified, since

$$B_a = K_a^{1/2} M_a^{1/2} + M_a^{1/2} K_a^{1/2}$$

is already positive definite. The same approach is used for the system shown in Fig. 4.10b, except that the stiffness term is first computed in gripper space, rather than in actuator space. This ensures a constant gripper compliance for all configurations of the manipulator. The control law is given by

$$f_{c_a} = B_a (\omega_d - \nu_d) + J_{g/a}^T K_a^{1/2} \int_0^t K_a^{1/2} J_{g/a} (\omega_d - \nu_d) dt - D_{1/2} \nu_d \quad (4.48)$$

and the damping term, $B_a$, is defined by

$$B_a = J_{k/b} U_b^T U_m J_{m/b} + J_{m/b}^T U_m^T U_m J_{k/b}$$

$$= K_a^{1/2} J_{g/a}^T U_b^T \Sigma_b V_b M_a^{1/2} + M_a^{1/2} V_b^T \Sigma_b \text{sgn}(\Sigma_b) U_b K_a^{1/2} J_{g/a}, \quad (4.49)$$

where

$$U_b^T \Sigma_b V_b = J_{k/b} J_{m/b}^{-1} = J_{g/a} K_a^{1/2} M_a^{-1/2}. \quad (4.50)$$
4.3 Asymptotic Stability for Arbitrary Environments

Stability of a robot system is a prerequisite for any control algorithm. When the robot interacts with an arbitrary environment, however, it is difficult to prove. Because the robot is nonlinear and the environment is nonlinear, frequency domain arguments are invalid. Stability for nonlinear systems can be shown by using three different approaches: dissipativeness and passivity (Popov 1962, Vidyasagar 1977, and Hill and Moylan 1976); the small gain theorem (Zames 1966); and Lyapunov theory (Vidyasagar 1978).

Consider the network shown in Fig. 4.11a, where $N_{1g}$ represents the robot network, and $N_{2g}$ represents the environment network. This is equivalent to the input-output stability diagram shown in Fig. 4.11b.

![Diagram showing input-output stability](image)

**Fig. 4.11.** Input-output stability: a) network; b) block diagram.

The system shown in Fig. 4.11 is input-output stable by reasons of passivity if both blocks $N_{1g}$ and $N_{2g}$ are passive, and one of the blocks is strictly passive, i.e., if there exists $\alpha$ such that

$$\int_0^t <f_g, v_g>_g dt \geq \alpha \int_0^t <v_g, v_g>_g .$$

(4.51)

By using passive controllers, it is possible to obtain passivity in the manipulator network, $N_{1g}$. It is not possible, however, to achieve strict passivity without disabling the manipulator's ability to track a trajectory. Similarly, although the environment is passive, it is not strictly passive, since it is unrealistic to assume that all motion in the environment is damped. Therefore, it is not possible to show input-output stability using passivity theory.

Input-output stability can also be shown via the small gain theorem. This approach has been used by Kazerooni (1987) in an attempt to prove the stability of robot contact. If a bound, $\overline{K}_a$, exists on the contact stiffness, i.e.,

$$\|(1+\alpha)J^T g f_g\| \leq \overline{K}_a \|x_a\|,$$

(4.52)

and if the position gain, $K_a$, is chosen so that

$$K_a \leq 1/\overline{K}_a,$$

(4.53)
then, assuming that the manipulator inertia and damping can be ignored, the control law,

\[ f_{c_a} = -K_a(x_a - x_{d_a}) \]

results in a total system which is input-output stable.

There are only two problems with this result. The assumptions are too strong, and the result is too weak. Determining an upper bound Eq. (4.52) relating force and position is difficult, if not impossible, for the full nonlinear system, and restricts the class of environments for which the stability result is applicable. Furthermore, the bound given in Eq. (4.53) is exceedingly conservative.

Researchers have also used Lyapunov theory to verify stability in robot control laws. In Fasse and Hogan (1988) Lyapunov functions are hypothesized for both the robot and the environment. A so-called "Lyapunov energy" is introduced which, when equal for both the robot and the environment, allows the sum of the individual Lyapunov functions to be used as a total Lyapunov function. Because of the difficulty in obtaining a general Lyapunov function, only an LTI system is analyzed.

In this section, Lyapunov theory will be combined with the Hilbert network model in order to prove the global stability of robot force control. The Lyapunov function is obtained directly from the PHIDE representation of the environment, and is correctly coupled to the manipulator by using Tellegen's theorem. Although the derivative of the Lyapunov function is only negative semidefinite, it is still possible to prove asymptotic stability of manipulator velocities by applying additional arguments.

Consider the generic passive force control law,

\[ f_{c_a} = -\alpha f^T_g f_g - B_a v_a + f_{c1_a}, \]  \hspace{1cm} (4.54)

where \( \alpha \) is the force feedback constant, \( B_a \) is any positive definite damping operator, and \( f_{c1_a} \) is an arbitrary passive controller. This is represented in Fig. 4.12. All of the passive controllers represented in this work can be put in this form. As the following theorem proves, asymptotic stability of manipulator velocities for all controllers of this type is guaranteed.

**Theorem 4.2 (Contact Stability)** Let Eq. (4.1) represent the dynamic equations for the manipulator, and assume that a passive control law of the form Eq. (4.54) has been implemented, where \( B_a \) is positive definite. Then, assuming that

(i) both the controller and the environment seen at the gripper can be modeled by a PHIDE network,
and that

(ii) the controller force, $f_{c1a}$, and the contact force, $f_g$, are a bounded function of the controller and gripper states, respectively,

the manipulator's joint velocity is asymptotically stable.

Proof

Let $x_{ak} \in H_{ak}$ and $x_{gk} \in H_{gk}$ represent the composite stiffness positions for the actuator controller and the gripper environment, and let $v_{am} \in H_{am}$ and $v_{gm} \in H_{gm}$ represent the composite inertia velocities for the actuator controller and the gripper environment. Define the state vector $x$ as follows:

$$x \triangleq (x_{ak} | v_{am} | v_a | x_{gk} | v_{gm}).$$

A positive definite Lyapunov function for the system is then

$$V(x) = \frac{1}{2}(|| x_{ak} ||^2 + || v_{am} ||^2 + v_d^T M_a v_a + \alpha || x_{gk} ||^2 + || v_{gm} ||^2).$$

Taking the derivative along the trajectories of the system gives

$$\dot{V}(x) = <x_{ak}, v_{ak}>_{ak} + <v_{am}, \dot{v}_{am}>_{am} + v_d^T M_a \dot{v}_a + v_d^T \dot{M}_a v_a$$

$$+ (1+\alpha)(<x_{gk}, v_{gk}>_{gk} + <v_{gm}, \dot{v}_{gm}>_{gm}).$$

Substituting for $M_a \dot{v}_a$, from Eq. (4.1) and Eq. (4.54), and replacing $x_{ak}, \dot{v}_{am}, x_{gk},$ and $\dot{v}_{gm}$ with $f_{ak}, f_{am}, f_{gk},$ and $f_{gm}$ respectively, gives

$$\dot{V}(x) = <f_{ak}, v_{ak}>_{ak} + <v_{am}, f_{am}>_{am}$$

$$- v_d^T (C_a v_a + B_a v_a - f_{c1a} + (1+\alpha) f_f^T f_g ) + \frac{1}{2} v_d^T \dot{M}_a v_a$$

$$+ (1+\alpha)(<f_{gk}, \dot{v}_{gk}>_{gk} + <v_{gm}, \dot{v}_{gm}>_{gm}).$$
Applying Lemma 2.1 and using the skew-symmetry of $C_a - \frac{1}{2}\dot{M}_a$ gives

$$
\dot{V}(x) = -<f_{ab}, v_{ab}>_{ab} + <f_{c1a}, v_a>_{a} - v_d^T B_a v_a - v_d^T f_{c1a} - (1+\alpha)v_d^T J_{gb}^T g - v_d^T (C_a - \frac{1}{2}\dot{M}_a) v_a - (1+\alpha)(<f_{gb}, v_{gb} > - <f_g, J_{gb}^T v_a>_g).
$$

$$\leq -v_d^T B_a v_a.$$

Because the derivative of the Lyapunov function is negative semidefinite, all of the signals, $x_{ak}, v_{am}, x_{gk}, v_{gm},$ and $v_a$, are bounded. Furthermore, since the controller and gripper states are bounded, it follows from assumption (ii) that $f_a$ and $f_g$ are bounded.

The joint acceleration is given by

$$
\dot{v}_a = -M_a^{-1}(C_a v_a + B_a v_a + J_{gb}^T f_g - f_{c1a}),
$$

and is bounded, since all terms on the right are bounded. By integrating $\dot{V}$ and using the positive definiteness of $B_a$, it is clear that $v_a$ is square integrable. A square integrable function with a bounded derivative, however, is asymptotically stable. Thus, $v_a$ is asymptotically stable. $\square$

A few comments are in order regarding this proof. Condition (ii) of the proof is extremely easy to meet, since it requires only that the contact force does not approach infinity for finite velocities and deflections. Condition (i), that both the manipulator and environment can be represented by a PHIDE network, is also not very restrictive. Certainly, any LTI system falls under this category, and as has been shown, many complex, nonlinear systems also satisfy this assumption.

The result of the proof, namely that the velocity, $v_a$, is asymptotically stable, is the best that can be expected for an arbitrary environment. In order for the position, $x_a$, to be asymptotically stable, it would be necessary to know the equilibrium position a priori. However, a passive environment can prevent a manipulator from reaching any preordained equilibrium position. This is called a collision. Of course, since $v_a$ is asymptotically stable, the position will settle down at some point.
5. TELEOPERATION

The term "teleoperator" means many things to many people. Remotely operated vehicles, unmanned spacecraft, and even long distance telephone operators might all be called teleoperators. Here, the term "teleoperator" refers to a master-slave configuration, in which the master and the slave are pedestal type robots.

Early teleoperators were mechanically linked (Vertut and Coiffet 1986). By using clever mechanical designs, the master on one side of a wall can be connected to a slave on the other side of a wall via a small conduit. The wall protects the human operator from exposure to a hazardous environment. Because the mechanical linkages directly transmit contact information, the human operator can maintain a high amount of dexterity. Unfortunately, purely mechanical teleoperators are limited, since they can be used only when the operator and the task are in close proximity. They cannot be used in space, under water, or in roving about a nuclear plant.

To allow teleoperation in remote environments, the behavior of the mechanical system is mimicked by the use of robots. Two configurations are possible. First, a joystick or teach pendant can be used for the master, which inputs desired velocities for the slave. Second, a back-drivable robot or "hand controller" can be used to directly command position set-points. Although far simpler to implement, the first approach has some significant disadvantages. When a teach pendant is used, the human operator must mentally differentiate his desired trajectory, a process which is unnatural for a human and usually requires numerous corrective motions. Furthermore, when a teach pendant is used, it is impossible to provide direct contact information to the operator. For these reasons, the second method is typically used for teleoperation, and is the only method considered in this thesis.

A number of strategies have been used in teleoperator control. The simplest is the open-loop approach. In this method, position set-points are obtained by measuring position encoders on the links of the master manipulator and are used for trajectory generation of the slave manipulator. The slave is then controlled using a PD controller architecture based on the error between the desired slave position and the actual slave position. No torques are used to back-drive the master manipulator, and thus no motors on the master are needed. There are two problems with this approach. First, if the slave robot has a lower bandwidth than the operator, then the two systems can get out of sync, since there is nothing to prevent the master from moving faster than the slave. Second, there is no sense of contact, no telepresence, and the operator's ability to perform a task is substantially reduced. To overcome these shortcomings, force-reflection, i.e., force information transmitted via motors on the master, is needed. When this is done, the teleoperator is said to be controlled bilaterally.
Force reflection can take two forms. The position error torque which drives the robot can also be used to drive the master. This term is called a *coordinating torque* and ensures that the master and slave stay in sync. In principle, the coordinating torque should also transmit contact force information, because if the slave encounters an object, the contact force counteracts the applied actuator force, and a position error results. In reality, however, the large gear ratios used on robots prevent end effector forces from substantially affecting joint torques, and thus little contact information is provided.

The second form of force reflection uses direct feedback of the measured force obtained at the slave's end effector. This provides a highly sensitive contact force reading to the operator, but leads to unstable contact behavior (Hannaford and Anderson 1987).

In both of these cases, the human operator, the master, the slave, and the environment are all a part of a coupled system. Analysis of the system requires modeling not only of the master and the slave, but also of the human operator and the environment. Instability in the master leads to instability in the slave, and vice versa.

When a time delay exists between the master and the slave, the system is complicated further. For time delays as small as 40 msec, the bilaterally controlled teleoperator can go unstable, and yet time delays for reaching space stations, or underwater stations exceed many tenths of a second. Furthermore, standard model based approaches to time delay, such as the Smith or Reswick controllers (Kolmanovskii and Nosov 1986), are not applicable, since the environment and the human operator are too nonlinear and too unpredictable for obtaining a reasonable model. Instability can be avoided if the system is run open-loop, i.e., with no force reflection, but then the ability of an operator to perform a task is severely diminished. Until now, this time delay problem in the bilateral control of teleoperators has been considered intractable.

In this chapter a new control law for teleoperators is presented which not only guarantees stable contact behavior with a high degree of sensitivity to the environment, but also overcomes the instability caused by time delay. The control law uses a passive controller architecture for both the master and the slave and for the communication circuit connecting the two systems. This is done by mimicking the passive behavior of a transmission line by using the line's input/output equations in the teleoperator control law. Considerable insight into the problem is gained by using scattering theory to analyze a single DOF LTI system. The analysis shows how existing control laws lead to instability, and how this instability is overcome using the new control law. Using a Lyapunov argument, stability for the entire n-DOF coupled nonlinear teleoperator system is proved. The approach parallels Theorem 4.3, except that here both the human operator and the environment have arbitrary behavior, and the effects of the
transmission line need to be included. Finally, experimental results are given which demonstrate both the instability that can occur when no compensation is given, and the stability that is obtained when the proposed control law is applied.

5.1 The Teleoperator System

A teleoperator system represented by the network diagram of Fig. 5.1 consists of five sub-networks: the human operator, \( N_h \); the master, \( N_m \); the communication block, \( N_c \); the slave, \( N_s \); and the environment, \( N_e \). The subnetworks are connected by the four transformers, \( J_{m/h}, J_{c/l/m}, J_{s/l/c}, \) and \( J_{e/l/s} \). The operator commands a velocity forward to the environment, through the master, communication block and slave. Likewise, the force sensed at the environment is transmitted back through these blocks, to the human operator. The variables used in the teleoperator are given in Table 5.1.

![Block diagram of the teleoperator system.](image)

The elaborate diagram of Fig. 5.1 is obtained by the simultaneous control of two separate robots, as is shown in Fig. 5.2. One robot, the master, is in contact with the human operator, and the other robot, the slave, is in contact with the environment. The dynamic equations for the two uncontrolled systems are given below. For the master network,

\[
\begin{align*}
    fcm &= -\frac{d}{dt}(M_m v_m) - D_m v_m + J^{-1}_{m/h} f_h \\
    &= -M_m \dot{v}_m - C_m v_m + J^{-1}_{m/h} f_h,
\end{align*}
\]

(5.1)

and for the slave network,

\[
    fcs = \frac{d}{dt}(M_s v_s) + D_s v_s + J^{-1}_{e/is} f_e
\]
<table>
<thead>
<tr>
<th></th>
<th>Variables</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>human operator</td>
<td>$v_h$</td>
<td>$n_{hm}$</td>
</tr>
<tr>
<td></td>
<td>$f_h$</td>
<td>no. inertial states</td>
</tr>
<tr>
<td></td>
<td>$v_{hm}$</td>
<td>$n_{hk}$</td>
</tr>
<tr>
<td></td>
<td>$x_{hk}$</td>
<td>no. stiffness states</td>
</tr>
<tr>
<td>master</td>
<td>$f_{1m}$</td>
<td>$M_m$</td>
</tr>
<tr>
<td></td>
<td>$f_{2m}$</td>
<td>inertia matrix</td>
</tr>
<tr>
<td></td>
<td>$v_m$</td>
<td>$D_m$</td>
</tr>
<tr>
<td></td>
<td>$v_m$</td>
<td>fictitious damping</td>
</tr>
<tr>
<td></td>
<td>$f_{cm}$</td>
<td>$C_m$</td>
</tr>
<tr>
<td></td>
<td>$x_m$</td>
<td>coriolis/centrip. force</td>
</tr>
<tr>
<td></td>
<td>$x_m$</td>
<td>$B_m$</td>
</tr>
<tr>
<td></td>
<td>$v_{c1}$</td>
<td>damping</td>
</tr>
<tr>
<td>commun. block</td>
<td>$f_{c1}$</td>
<td>$T$</td>
</tr>
<tr>
<td></td>
<td>$v_{c2}$</td>
<td>time delay</td>
</tr>
<tr>
<td></td>
<td>$x_{c2}$</td>
<td></td>
</tr>
<tr>
<td>slave</td>
<td>$v_{1s}$</td>
<td>$M_s$</td>
</tr>
<tr>
<td></td>
<td>$v_{2s}$</td>
<td>inertia matrix</td>
</tr>
<tr>
<td></td>
<td>$f_{1s}$</td>
<td>$C_s$</td>
</tr>
<tr>
<td></td>
<td>$f_{2s}$</td>
<td>coriolis/centrip. force</td>
</tr>
<tr>
<td></td>
<td>$x_{2s}$</td>
<td>$B_{1s}$</td>
</tr>
<tr>
<td></td>
<td>$x_{2s}$</td>
<td>error damping</td>
</tr>
<tr>
<td></td>
<td>$f_{cs}$</td>
<td>$B_{2s}$</td>
</tr>
<tr>
<td></td>
<td>$v_{e}$</td>
<td>rate damping</td>
</tr>
<tr>
<td>environment</td>
<td>$f_e$</td>
<td>$K_s$</td>
</tr>
<tr>
<td></td>
<td>$v_{em}$</td>
<td>stiffness gain</td>
</tr>
<tr>
<td></td>
<td>$x_{hk}$</td>
<td>$D_s$</td>
</tr>
<tr>
<td></td>
<td>$x_{hk}$</td>
<td>fictitious damping</td>
</tr>
<tr>
<td>Jacobians</td>
<td>$J_{m/h}$</td>
<td>$n_{em}$</td>
</tr>
<tr>
<td></td>
<td>$J_{c1/m}$</td>
<td>no. inertial states</td>
</tr>
<tr>
<td></td>
<td>$J_{s/c2}$</td>
<td>$n_{ek}$</td>
</tr>
<tr>
<td></td>
<td>$J_{c1/s}$</td>
<td>no. stiffness states</td>
</tr>
<tr>
<td></td>
<td>$J_{c2/s}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$J_{e/c}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$J_{s/e}$</td>
<td></td>
</tr>
</tbody>
</table>
\[ = M_s v_2^s + C_s v_2^s + J_{\theta/s}^T f_e. \] (5.2)

The communication block, the Jacobians, \( J_{c/m} \) and \( J_{s/c} \), and a good portion of the master and slave networks are all created in the controller. The appropriate equations for \( f_{c/m} \) and \( f_{c/s} \) give this result.

![Diagram](image)

Fig. 5.2 The uncontrolled teleoperator system.

Ideally, the environmental velocity, \( v_e \), should track the human velocity, \( v_h \), and the human applied force, \( f_h \), should track the environmental contact force, \( f_e \). To maintain a passive architecture the coordinating torque, rather than the environmental contact force, is used in the control law. A passive control for the teleoperator is developed below.

Let the master torque command, \( f_{c/m} \), consist of a local damping term, \( B_m \), and the reflected force term obtained from the communication block:

\[ f_{c/m} = B_m v_m + f_{2/m}, \] (5.3)

where

\[ f_{2/m} = J_{c/1/m}^T f_{c/1}, \] (5.4)

and set the input velocity of the communication block to

\[ v_{c/1} = J_{c/1/m} v_m. \] (5.5)

The Jacobian, \( J_{c/1/m} \), should be used to transfer signals in the master manipulator’s joint space back to world coordinates. Thus, \( J_{c/1/m} = J_{m/h}^T \).

Let the slave torque command consist of a local PD control based on the error between the slave position, \( x_2^s \), and the slave set-point, \( x_{1/s} \), and in addition, use a local force loop gain, \( \alpha \), to feed back the contact force, \( f_e \):

\[ f_{c/s} = -(1+\alpha)B_2 v_2^s + (1+\alpha)f_{1/s} - \alpha J_{\theta/s}^T f_e, \] (5.6)

where the coordinating torque, \( f_{1/s} \), is given by
\[ f_{1s} = B_{1s} (v_{1s} - v_{2s}) + K_s (x_{1s} - x_{2s}) \]
\[ = B_{1s} \Delta v_s + K_s \Delta x_s. \]  \hfill (5.7)

The slave set-point velocity is obtained from the communication block,
\[ v_{1s} = J_{s/c2} v_{c2}, \]  \hfill (5.8)
and the coordinating torque term, \( f_{1s} \), is transmitted to the communication block by the equation,
\[ f_{c2} = J_{s/c2}^T f_{1s}. \]  \hfill (5.9)

The Jacobian, \( J_{s/c2} \), maps the velocities of the communication circuit, which should be in world coordinates, to the slave's joint space. Thus, \( J_{s/c2} = J_{e1s}^{-1} \).

If no time delay exists between the master and the slave, then no special control law is needed for the communication circuit. The communication circuit law is simply
\[ v_{c2}(t) = v_{c1}(t) \]  \hfill (5.10)
and
\[ f_{c1}(t) = f_{c2}(t). \]  \hfill (5.11)

Unfortunately, when the master and slave are remotely located, signals between the two systems can be transmitted only after a delay of \( T \) seconds. With this delay, the equations (5.10) and (5.11) become
\[ v_{c1}(t) = v_{c2}(t-T) \]  \hfill (5.12)
and
\[ f_{c1}(t) = f_{c2}(t-T). \]  \hfill (5.13)

As is shown in Section 5.2, the delayed signals Eqs. (5.12) and (5.13) are nonpassive and result in an unstable system. To overcome this nonpassivity, the input-output form of the transmission line equations (given previously as Eqs. (3.14) and (3.15)),
\[ v_{c2}(t) = v_{c1}(t-T) + Z_c^{-1}(f_{c2}(t) - f_{c1}(t-T)) \]  \hfill (5.14)
\[ f_{c1}(t) = f_{c2}(t-T) + Z_c (v_{c2}(t) - v_{c1}(t-T)), \]  \hfill (5.15)
should be used in place of Eqs. (5.12) and (5.13).

The network model for the controlled system, with the passively controlled sources replaced by passive elements and using the transmission line equations, is shown in Fig. 5.3.
Although the coordinating torque term, $f_{1s}$, and not the contact force, $f_e$, is transmitted through the communication block, this does not lead to poor sensitivity to the environment. The local force feedback in the slave, $\alpha$, causes the coordinating torque term, $f_{1s}$, to closely track the contact force, $f_e$.

Teleoperation is often performed in environments which either require a greater amount of strength than the human operator can provide, such as in the assembly of large structures, or require significantly more dexterity than the human can provide, such as in micro-manipulation. In either case power scaling is necessary. This power scaling is achieved with a two-port amplifier element and a transformer. As was shown in Theorem 2.8, the two-port gain element does not affect the passivity of the system, as long as the amplifier gain is scalar valued. Thus, instead of Eqs. (5.4) and (5.5) the gain scaling equations,

\begin{align}
  v_{c1} &= \lambda_1 J_A J_{c1/m} v_m \\
  f_{2m} &= \lambda_1^{-1} J_{T1/m} J_A f_{c1},
\end{align}

are used, where $\lambda_1$ is the amplifier gain, and $J_A$ is a transformer. Typically, only the prismatic velocities should be scaled, since identical orientations need to be maintained for the master and the slave. Thus, $J_A$ should be set to

$$J_A \triangleq \text{diag} (\lambda_2 I_3, \lambda_1^{-1} I_3),$$

where $\lambda_2$ is an arbitrary scalar.
5.2 Frequency Domain Analysis

The standard teleoperator system considered here consists of two n-DOF manipulators, and is highly nonlinear. Nevertheless, considerable insight into the teleoperation time-delay problem is obtained by looking at a single DOF linear time-invariant system. Furthermore, because of the notation and network approach developed in this work, Eqs. (5.1)-(5.15) and the networks shown in (Figs. 5.1 and 5.2), which were used to describe the full nonlinear system, can also describe the single DOF linear system, except that the underlying spaces, $H_h$, $H_m$, $H_{c1}$, $H_{c2}$, $H_s$, and $H_e$, now equal $\mathbb{R}$, and no fictitious damping terms are necessary.

For simplicity, assume that the transformers, $J_{m/h}$, $J_{c1/m}$, $J_{s/c2}$, and $J_{e/s}$, are equal to unity. The dynamic equations Eqs. (5.1) and (5.2) for the single DOF teleoperator system can then be written as

\begin{align}
fc_m &= -M_m \ddot{v}_m + f_h \\
f_c &= M_s v_2 + f_e 
\end{align}

and the applied control Eqs. (5.3) through (5.9) is given by

\begin{align}
f_{c_m} &= B_m v_m + f_{c_1} \\
v_{c_1} &= v_m \\
f_c &= -(1+\alpha)B_2 v_2 + (1+\alpha)f_1 - \alpha f_e \\
f_1 &= K_s \int (v_1 - v_2) dt + B_1 (v_1 - v_2) \\
v_1 &= v_{c_1} \\
f_{c_2} &= f_s
\end{align}

The single DOF system is shown in Fig. 5.4.

![Fig. 5.4 Circuit representation of single DOF teleoperator.](image)
The advantages of the force feedback term, \( \alpha \), are easy to analyze using the LTI system and frequency domain analysis. By combining Eqs. (5.19) and (5.21) and using the environment’s impedance relationship, \( f_e(s) = Z_e(s)v_2(s) \), the contact force and the coordinating torque term can be related:

\[
f_{1s} = \left( \frac{M_s s + (1+\alpha)B_2}{(1+\alpha)Z_e(s)} + 1 \right) f_e; \]

and for suitably large \( \alpha \), and sufficiently small series damping, \( B_2 \),

\[
f_{1s} \equiv f_e.
\]

Therefore, when a local force loop is closed around the slave, as is done in Eqs. (5.6) and (5.21), sensitivity to the environment is obtained, and the coordinating torque signal becomes a valid signal for conveying contact force information.

To analyze the behavior of the single DOF teleoperator system, first assume that no time delay compensation is used. A scattering theory analysis of the two-port given by Eqs. (5.12) and (5.13) shows that the two-port communication circuit is nonpassive. The associated hybrid matrix is obtained directly as

\[
H_{c1c2}(s) = \begin{bmatrix} 0_{c1} & e^{-sT}I_{c1/c2} \\ -e^{-sT}I_{c2/c1} & 0_{c2} \end{bmatrix}.
\]

Using Eq. (2.23), the scattering matrix for the system is obtained:

\[
S_{c1c2} = \begin{bmatrix} I_{c1} & 0 \\ 0 & -Z_{c2} \end{bmatrix} \begin{bmatrix} Z_{c1} & 0 \\ 0 & -Z_{c2} \end{bmatrix} \begin{bmatrix} I_{c1} & 0 \\ 0 & -Z_{c2} \end{bmatrix}^{-1} \begin{bmatrix} I_{c1} & 0 \\ 0 & -Z_{c2} \end{bmatrix}
\begin{bmatrix} e^{-sT}I_{c1/c2} \\ -e^{-sT}I_{c2/c1} Z_{c2}^{-1} \end{bmatrix} \begin{bmatrix} Z_{c1} & 0 \\ 0 & -Z_{c2} \end{bmatrix}^{-1} \begin{bmatrix} I_{c1} & 0 \\ 0 & -Z_{c2} \end{bmatrix}
\begin{bmatrix} e^{-sT}I_{c1/c2} \end{bmatrix} \begin{bmatrix} e^{-sT}I_{c2/c1} -Z_{c2} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ -Z_{c2} \end{bmatrix}
\begin{bmatrix} -\tanh(sT)I_{c1} & \text{sech}(sT)I_{c1/c2} \\ \text{sech}(sT)I_{c2/c1} & \tanh(sT)I_{c2} \end{bmatrix}.
\]

(5.24)

Computing the induced frequency norm (Def. 1.14) gives

\[
\| S_{c1c2} \|_{c1c2} = \sup_{\omega} \lambda^\frac{1}{2} \begin{bmatrix} j \tan(\omega T)I_{c1} & \sec(\omega T)I_{c1/c2} \\ \sec(\omega T)I_{c2/c1} & -j \tan(\omega T)I_{c2} \end{bmatrix} \begin{bmatrix} -j \tan(\omega T)I_{c1} & \sec(\omega T)I_{c1/c2} \\ \sec(\omega T)I_{c2/c1} & j \tan(\omega T)I_{c2} \end{bmatrix}
\begin{bmatrix} \tan^2(\omega T) + \sec^2(\omega T)I_{c1} & 2j \tan(\omega T)\sec(\omega T)I_{c1/c2} \\ -2j \tan(\omega T)\sec(\omega T)I_{c2/c1} & \tan^2(\omega T) + \sec^2(\omega T)I_{c2} \end{bmatrix},
\]

(5.25)
and thus \( \| S_{c1 \times c2} \|_{c1 \times c2} = \sup_{\omega} (|\tan(\omega T)| + |\sec(\omega T)|) = \infty \). Therefore, the scattering operator for this system is unbounded, and hence the system is not passive. In practice, the signals entering the communication block are band-limited. Since \( |\tan(\omega T)| + |\sec(\omega T)| > 1 \) for any \( \omega, T > 0 \), the communication block is never passive for any range of frequencies. Furthermore, the instability problems are insurmountable as \( \omega T \to \pi/2 \). Using the control laws equations (5.12) and (5.13), only a system with extremely band-limited behavior is able to remain stable. The criterion \( \omega T < \pi/2 \) represents a stability barrier that all systems without time delay compensation must satisfy.

Applying the same analysis to the new control law Equations (5.14) and (5.15) shows that the resulting system is stable for any time delay. In the frequency domain, the transmission line equations are given by

\[
\begin{align*}
f_{c1}(s) &= Z_c \tanh(sT)v_m(s) + \text{sech}(sT)f_{1s}(s) \\
&= Z_c \tanh(sT)v_{c1}(s) + \text{sech}(sT)f_{c2}(s) \\
-v_{1s}(s) &= -\text{sech}(sT)v_m(s) + Z_c^{-1}\tanh(sT)f_{1s}(s) \\
&= -\text{sech}(sT)v_{c1}(s) + Z_c^{-1}\tanh(sT)f_{c2}(s).
\end{align*}
\]

After a little algebraic manipulation (applying Eq. (2.23)), Equations (5.26) and (5.27) can be represented as,

\[
\begin{bmatrix}
f_{c1}(s) - Z_c v_{c1}(s) \\
f_{c2}(s) + Z_c v_{c2}(s)
\end{bmatrix} =
\begin{bmatrix}
0_{c1} & e^{-sT}I_{c1 \times c2} \\
e^{-sT}I_{c2 \times c1} & 0_{c2}
\end{bmatrix}
\begin{bmatrix}
f_{c1}(s) + v_{c1}(s) \\
f_{c2}(s) - v_{c2}(s)
\end{bmatrix},
\]

and thus the scattering matrix for the communication block, \( S_{c1 \times c2} \), is given by

\[
S_{c1 \times c2} =
\begin{bmatrix}
0_{c1} & e^{-sT}I_{c1 \times c2} \\
e^{-sT}I_{c2 \times c1} & 0_{c2}
\end{bmatrix}.
\]

The frequency norm (Def. 1.14) of Eq. (5.29) is

\[
\| S_{c1 \times c2} \|_{c1 \times c2} = \sup_{\omega} \lambda^4(S_{c1 \times c2}^*(j\omega)S_{c1 \times c2}(j\omega))
\]

\[
= \sup_{\omega} \lambda^4 \begin{bmatrix}
I_{c1} & 0 \\
0 & I_{c2}
\end{bmatrix} = 1.
\]

Therefore, by Thm. 2.4, the two-port (Eqs. (5.14) and (5.15) is passive. In addition, by setting \( s=0 \) in Eqs. (5.26) and (5.27) it is clear that in the steady-state \( f_{c1} = f_{1s} \) and \( v_{1s} = v_m \).
5.3 Stability Analysis

The following proof is based on a Lyapunov argument, and determines the response of the system to a set of initial conditions. It is assumed that the human and environment can be modeled by passive networks with independent sources, and at time $t=0$, all independent sources are set to zero. It follows the proof of Theorem 4.2, but includes the additional complexities of having two robot systems, two arbitrary networks, and a distributed parameter communication system. This proof does not include any power scaling. As was shown in Section 2.5, a two-port gain element could be added to the system, and the stability proof need only be modified by scaling all the signals on one-half of the network by the power gain. Caution needs to be taken, however, that all directions are scaled equally, since non-scaler power scaling can lead to instabilities in an otherwise passive system.

Theorem 5.1 (Teleoperator Stability) Let Eqs. (5.1) and (5.2) represent the dynamic equations for the master and slave, and assume that the force reflecting control law with time delay compensation given by the equations Eqs. (5.3)-(5.9) and (5.14)-(5.15), has been applied. Then, assuming that

(i) both the human operator and the environment can be modeled by a PHIDE network, and that

(ii) the environmental force, $f_e$, and the applied human force, $f_h$, are a bounded function of the human and environmental states, respectively,

the master and slave velocities are asymptotically stable.

Proof Let $x_{hk} \in H_{hk}$ and $x_{ek} \in H_{ek}$ represent the composite stiffness positions for the human operator and the environment, and let $v_{hm} \in H_{hm}$ and $v_{em} \in H_{em}$ represent the composite inertia velocities for the human operator and the environment. Define the state vector $x$ as follows:

$$x \triangleq (x_{hk} | v_{hm} | x_{ek} | v_{em} | v_m | v_2 s | \Delta x_s | f_c | v_c).$$

A positive definite Lyapunov function for the system is then

$$V(x) = \frac{1}{2} \| x_{hk} \|^2 + \| v_{hm} \|^2 + \| x_{ek} \|^2 + \| v_{em} \|^2 + v_m^T M_m v_m + \frac{1}{1 + \alpha} v_2^T M_s v_2 s + \Delta x_s^T K_s \Delta x_s + \langle f_c, K_c^{-1} f_c \rangle + \langle v_c, M_c v_c \rangle).$$

From (Lemma 3.1) this is equivalent to

$$V(x) = \frac{1}{2} \| x_{hk} \|^2 + \| v_{hm} \|^2 + \| x_{ek} \|^2 + \| v_{em} \|^2 + v_m^T M_m v_m + \frac{1}{1 + \alpha} v_2^T M_s v_2 s.$$
Taking the derivative along the trajectories of the system gives

\[
\dot{V}(x) = \langle x_{hk}, \dot{v}_{hk} \rangle_{hk} + \langle v_{hm}, \dot{v}_{hm} \rangle_{hm} + \langle x_{ek}, \dot{v}_{ek} \rangle_{ek} + \langle v_{em}, \dot{v}_{em} \rangle_{em} \\
+ v_m^T M_m \dot{v}_m + v_m^T \dot{M}_m v_m + \frac{1}{2} \frac{1}{1+\alpha} v_1^T M_1 v_2 \\
+ \frac{1}{2} \frac{1}{1+\alpha} v_2^T M_2 v_2 + \Delta v_k^T K_s \Delta x_s + f_c^T v_c - f_c^T v_2c.
\]

Substituting in for $M_m \dot{v}_m$, $M_s \dot{v}_2$ from Eqs. (5.1) and (5.2), and replacing $x_{hk}$, $\dot{v}_{hm}$, $x_{ek}$, and $\dot{v}_{em}$ with $f_{hk}$, $f_{hm}$, $f_{ek}$, and $f_{em}$ respectively, gives

\[
\dot{V}(x) = \langle f_{hk}, \dot{v}_{hk} \rangle_{hk} + \langle v_{hm}, \dot{v}_{hm} \rangle_{hm} + \langle f_{ek}, v_{ek} \rangle_{ek} + \langle v_{em}, \dot{v}_{em} \rangle_{em} \\
- v_m^T (C_m v_m - f \dot{1}_m + B_m v_m + f \dot{2}_m) + \frac{1}{2} v_m^T \dot{M}_m v_m \\
- v_1^T (\frac{1}{1+\alpha} C_s v_2 - B_1 \Delta v_s + B_2 v_2 - K_s \Delta x_s - J_c^T f_{cc}) \\
+ \frac{1}{2} \frac{1}{1+\alpha} v_2^T M_2 v_2 + v_1^T K_s \Delta x_s - v_2^T K_s \Delta x_s + f_c^T v_c_1 - f_c^T v_2c.
\]

Substituting from Eqs. (5.3)-(5.9) and applying Lemma 2.1 gives

\[
\dot{V}(x) \leq -v_m^T f_{hk} + v_1^T f_{ek} - v_m^T B_m v_m + v_m^T \dot{J}_1 \dot{m} f_{c1} \\
- v_1^T B_2 v_2 + (v_1 - \Delta v_s)^T f \dot{1}_s - v_2^T \dot{J}_1 \dot{m} f_{c2} \\
+ \Delta v_2^T (f \dot{1}_s - B_1 \Delta v_s) + f_1^T f_{c1} \dot{m} v_m - f \dot{1}_s J_{s/c} v_2 c_2 \\
\leq -v_m^T B_m v_m - v_2^T B_2 v_2 - \Delta v_2^T B_2 \Delta v_s.
\]

Because the derivative of the Lyapunov function is negative semidefinite, all of the signals, $x_{hk}$, $v_{hm}$, $x_{ek}$, $v_{em}$, $v_m$, $v_2$, and $\Delta x_s$ are bounded. Because $B_m$, $B_1$, and $B_2$ are positive definite, it is also clear from integrating $\dot{V}$ that $v_m \in L_2^\infty[0,\infty)$, $v_2 \in L_2^\infty[0,\infty)$ and $\Delta v_s \in L_2^\infty[0,\infty)$. Since the human operator states and the environmental states are bounded, it follows from assumption (ii) that $f_{hk}$ and $f_{c2}$ are bounded.

If it can be shown that $\dot{v}_m \in L_\infty^\infty[0,\infty)$ and $\dot{v}_2 \in L_\infty^\infty[0,\infty)$ then both the master velocity, $v_m$, and the slave velocity, $v_2$, are asymptotically stable, since a bounded $L_2^\infty[0,\infty)$ function with a bounded derivative is asymptotically stable.

The defining equation for $\dot{v}_m$ is

\[
\dot{v}_m = -M_m^{-1}(C_m v_m + B_m v_m + \dot{J}_1 \dot{m} f_{c1} - J_{m/h} f_{hk}).
\]

All of the terms in this expression are bounded by the previous Lyapunov argument, except
Similarly,
\[
f_{c1}(t) = f_{c2}(t-T) - Z_c v_{c2}(t-T) + Z_c v_{c1}(t)
\]
\[
= B_{c2}(t-T) v_{c2}(t-T) - Z_c v_{c2}(t-T) - J_{c2}^T B_1 v_{2s}(t-T)
\]
\[
+ K_s \Delta v_s(t-T) + Z_c J_{c1/m} v_m(t).
\]
Because the last three terms are bounded, there exists \( \alpha_2 \), such that
\[
\| f_{c1}(t) \|_c \leq \| Z_c - B_{c2}(t-T) \|_c \| v_{c2}(t-T) \|_c + \alpha_2.
\]  \( (5.30) \)

Combining these two inequalities together gives
\[
\| v_{c2}(t) \|_c \leq \frac{\| Z_c - B_{c2}(t-T) \|_c}{\| Z_c + B_{c2}(t-T) \|_c} \| v_{c2}(t-T) \|_c + \alpha_3,
\]
where \( \alpha_3 \triangleq \alpha_2 + \alpha_1/\| Z_c + B_{c2}(t-T) \|_c \). From this argument, \( v_{c2} \) cannot escape in finite time, and assuming full rank on the Jacobian, \( J_{c2} \), then \( B_{c2} \) is positive definite, and \( v_{c2} \) is also bounded, since in this case
\[
\frac{\| Z_c - B_{c2}(t-T) \|_c}{\| Z_c + B_{c2}(t-T) \|_c} < 1.
\]

Once a bound on \( v_{c2} \) is obtained, equation Eq. \( (5.30) \) implies that bound exists on \( f_{c1} \), and thus \( v_m \) and \( v_{2s} \) are asymptotically stable. \( \square \)
5.4 Experimental Results

The control law given by Eqs. (5.3-5.9) have been implemented on a single DOF master slave system developed at the Jet Propulsion Laboratory in Pasadena. The master and slave are both single-DOF dc-motor driven systems with position encoders and PWM drivers. Contact force information is obtained via a strain gauge bridge connected to the base of a beam which protrudes from the slave subsystem. The controller runs at a 500 Hz interrupt rate, and calculations are done using double precision floating point numbers. A preliminary version of this system is described further in (Hannaford and Anderson 1987). Figure 5.5 shows a diagram of the setup.

![Diagram of Teleoperator demonstration setup](image)

Fig. 5.5 Teleoperator demonstration setup.

In order to analyze the fidelity of the teleoperator system, two different tasks were attempted. The first involves hard contact. The slave is commanded forward until a hard surface is contacted. The force is ramped up and then back down until contact is lost. The process is repeated twice in twenty seconds. Ideally, the slave position, \( x_{2s} \), tracks the master position, \( x_m \), until contact. Once contact is made, the master motor torque, \( f_{2m} \), should track the contact force, \( f_e \). Figure 5.6 shows the response for the system without time delay.
Fig. 5.6. Hard contact test for system without time delay: a) $x_m$ (solid) and $x_{2s}$ (dashed) in degrees vs. time in seconds; b) $f_e$ (solid) and $f_{2m}$ (dashed) in oz-in. vs. time in seconds.
The second task involves back-drivability. The master is allowed to move freely, while the slave is hammered twice and then moved sinusoidally. Ideally, the master position should track the slave position without going unstable. Figure 5.7 shows the results of the back-drivability test without time delay.

![Graph showing back-drivability test](image_url)

**Fig. 5.7.** Back-drivability test for system without time delay: $x_m$ (solid) and $x_{2s}$ (dashed) in degrees vs. time in seconds.
The back-drivability test was performed using the same local controllers for the master and the slave, but with a time delay of 40 msec in the communication block, and without time delay compensation. The results are given in Fig. 5.8. The system, as is apparent from the plots, is easily destabilized by back-driving the slave. Further experimentation showed that the system could be destabilized with as little as 25 msec time delay in the communication loop.

![Diagram](image)

**Fig. 5.8.** Back-drivability test for the system with 40 msec time delay: \( x_m \) (solid) and \( x_{2_s} \) (dashed) in degrees vs. time in seconds.
The hard contact and back-drivability tests were done using the system with the time delay compensation Eqs. (5.14) and (5.15). Figure 5.9 shows the back-drivability test response of the system with 40 msec of time delay. Unlike the system without time-delay compensation, this system could not be destabilized. Figure 5.10 shows the hard contact test responses of the system with 40 msec of time delay. The desired position and force commands are tracked without noticeable degradation.

Fig. 5.9 Back-drivability test for system with 40 msec delay and compensation: $x_m$ (solid) and $x_{2s}$ (dashed) in degrees vs. time in seconds.
Fig. 5.10  Hard contact test for system with 40 msec of time delay and compensation: a) $x_m$ (solid) and $x_{2s}$ (dashed) in degrees vs. time in seconds; b) $f_e$ (solid) and $f^{2m}_e$ (dashed) in oz-in vs. time in seconds.
The time delay was increased further to 200 msec, and the same tests were repeated. Figure 5.11 shows the back-drivability test responses of the system with 200 msec of time delay. Still, the system could not be destabilized. What appears to be an increasing signal magnitude is due only to an increase in the applied force, $f_e$, and not to instability. Figure 5.12 shows the hard contact test responses of the system with 200 msec of time delay. The position and force commands are still transmitted with high fidelity.

Time delays of up to two seconds were tested. In all cases the system could not be destabilized. Unfortunately, the tracking of force and position begins to degrade for the higher time delays. Since this is not predicted in the continuous time theory, and was not the case in simulations, it is believed that this degradation is due to the particular discrete implementation used on the system.

![Graph](image)

Fig. 5.11. Back-drivability test for system with 200 msec delay and compensation: $x_m$ (solid) and $x_{2e}$ (dashed) in degrees vs. time in seconds.
Fig. 5.12. Hard contact test for system with 200 msec of time delay and compensation: a) $x_m$ (solid) and $x_{2s}$ (dashed) in degrees vs. time in seconds; b) $f_e$ (solid) and $f_{2m}$ (dashed) in oz-in vs. time in seconds.
5.5 Conclusions

This chapter has introduced a new control law for controlling a teleoperator with time delay. The teleoperation system was divided into five blocks: the operator, master, communication block, slave, and environment. By using the Hilbert network model and applying scattering theory, it was shown that the instability which occurs in bilateral teleoperation in the presence of time delay is due to a nonpassive communication block.

By active control, a communication block which mimics a lossless transmission line has been achieved and is therefore guaranteed to be a passive element independent of time delay. Thus, stability for the teleoperator has been obtained without restricting the bandwidth of signals.

In so far as the proposed control law maintains passivity for the master and slave subsystems, the closed-loop system is stable for a wide variety of perturbations. For instance, the communication delay can be unknown and time varying and the environment and human operator dynamics can vary within the entire class of passive systems. In addition, the damping and stiffness gains can be arbitrarily assigned. Given this robust stability, the need to model the human and environment and to tune the damping and stiffness gains is dictated solely by performance, rather than stability issues.
6. CONCLUSIONS

The development of the Hilbert network theory and its application to robotics and tele-operation have led to a number of original results.

A network model for an n-DOF manipulator has been developed which leads to a simple and intuitive method for obtaining dynamic equations. Furthermore, characteristics such as the manipulator Jacobian and the equivalent joint stiffness are derivable from this model. By using the network model in accordance with a recent solution for closed-link robots, an even simpler method of solution for closed link manipulators has been discovered.

When the network model is used to describe environments, an intuitive model for robot/environment interaction is obtained. The model allows representation of either the conventional, 6-DOF impedance representation, or a more realistic distributed contact model. Complexities, such as nonlinear stiffnesses, coulomb friction, and switching phenomena, are all representable using the Hilbert network. In addition, the network model can be used to model distributed parameter systems, and examples for beams, membranes, and Maxwell's equations have been given.

By applying network ideas to contemporary robot control strategies, a number of potential problems have been recognized, and solutions to these problems have been developed. In hybrid control strategies, either the improper use of a selection matrix or the unequal application of force feedback can lead to a nonpassive system. Furthermore, there are a number of problems with using task space equations which have not been previously considered. In this work, a passive controller implementation of hybrid control has been introduced which addresses all these issues.

It has been shown how the standard implementation of computed torque can lead to instability when a poor model of the manipulator inertia is used. Based on an understanding of "passive controller" structures, a new implementation of the computed torque algorithm has been presented which guarantees stability, regardless of the inertia estimate. In addition, an entire new class of "variable bandwidth," computed torque controllers has been introduced, which ensures critical damping while allowing the system bandwidth to change with the configuration. This approach fully utilizes the robot motors, and makes it possible to achieve constant gripper compliance.

Based on the passive implementations of standard force control laws, a proof of the asymptotic stability of robots in contact with arbitrary environments has been given. It has been shown that if the controller is implemented passively, and if the environment can be represented by a PHIDE network, then the velocities of the robot are asymptotically stable.
The application of network ideas to teleoperators has led to a method for solving the time-delay instability problem in force-reflecting teleoperation. First, by using scattering theory it is shown how standard implementations of teleoperator controllers are nonpassive for any amount of time delay. For these systems, an upper bound for the system bandwidth is achieved which is inversely proportional to time delay. Then, using the same scattering theory motivation, a solution for the time-delay problem is derived. A control law, which mimics the behavior of a lossless transmission line, is shown to accommodate the effects of time delay. Asymptotic stability for the teleoperator is obtained by assuming that both the manipulator and the environment can be modeled by PHIDE networks. Both the instability problem and its solution have been demonstrated experimentally using a single-DOF teleoperator.

6.1 Pros and Cons of the Network Approach

There are a number of advantages and disadvantages in using a network approach for developing control laws in robotics and teleoperation. The network approach can lead to considerable insight into the problems of and solutions for designing robot force control laws. By utilizing passive controller architectures, i.e., architectures which have a network representation without the use of dependent sources, it is possible to ensure stability of robot motion operating in arbitrary environments.

Network models are useful for showing the underlying structure of a system. In the case of LTI systems, an input-output description of the system is perfectly valid, but for nonlinear systems, knowledge of the exact structure of a system is often crucial for designing effective control laws. By using the network models, nonlinear control laws such as computed torque and hybrid control can be intuitively understood. Furthermore, it is much easier to analyze unexpected phenomena. For instance, a researcher armed only with the set of dynamic equations for the entire manipulator would be hard pressed to obtain control laws for dealing with potential collisions between interior links of the robot and the environment. Using the network model of the manipulator, however, the determination of the resulting forces is simple.

A network approach that requires that controllers be implemented passively, however, has its disadvantages. For instance, Proportional-Integral-Derivative (PID) control is not a passive controller structure, but it is commonly used in robot position control to overcome bias forces, such as gravity. The PID control structure is not commonly used for force controllers, however, because the controller force can ramp up without bound when the robot prematurely contacts a hard surface. Furthermore, even in the case of free space motion, a manipulator under PID control can go unstable if it is carrying an object which is too heavy.
Often the requirement of passivity for the manipulator is unattainable. When actuators and sensors are not collocated, for instance, passivity is often lost. In robots this can occur if joint position differs from the motor rotor position due to joint flexibility, and only the measured joint position is used in the control law. The problem is even more severe when force feedback is applied around a robot which has either compliant joints or compliant links. Because actuators exist only at the joints, the use of a constant force feedback term of the form, $J^T g \alpha f_g$, results in a nonpassive control law. Motor dynamics can also lead to a system which is not passive. Nevertheless, in most cases the difference between the real system and the passive representation of the system is minimal, and far more is to be gained by using the passive controller representation. When instability problems do occur, they can then be analyzed with regard to a deviation from the passive model.

Passive implementations of complex control architectures are often more computationally intensive than nonpassive control architectures. Although the simplest possible control architecture, the joint PD control, is passive, other algorithms require significantly more calculations to maintain passivity. In the passive implementation of computed torque, it is necessary to compute the square root of the estimated inertia matrix and perform twice as many matrix multiplications as in the standard implementation. It is possible that robust control laws (Spong, et al. 1987), which can also ensure stability for poor estimates of the manipulator inertia, offer a better allocation of computer resources. The passive implementation of hybrid control also involves more computations than the standard implementation. There may be cases where the guaranteed stability obtained by using a passive controller does not justify the additional computational effort.

6.2 Work to Be Done

The Hilbert network formulation presented here is applicable to more than robots and teleoperators, and certainly much can be done to extend these ideas to other areas. Even in the domain of robotics, the work presented here is far from complete.

First, the underlying mathematical framework of the Hilbert networks should be made rigorous. The entire formulation has been based out of engineering, rather than mathematical, concerns, and sometimes the term "Hilbert space" has been used loosely. Problems such as the use of unbounded operators, the allowable types of functional and time-varying dependencies, and the uniqueness and existence of solutions when the manipulator contacts environments with infinite stiffness or with coulomb friction need to be carefully treated.

The proof of asymptotic stability of manipulator velocities for arbitrary environments should be extended to include stability of position and force set-points. By extending the
definitions of "capacitive" and "inertial" environments used in (Anderson and Spong 1988) to the full nonlinear model, it seems reasonable to expect that zero error to a position set-point could be achieved when the environment is inertial along a given direction, and that zero error to a force set-point could be achieved when the environment is capacitive along a given direction in task space. This result was proved for the single DOF case in (Anderson and Spong 1988).

The distributed contact models used for describing environment contact should be used in developing a hybrid control algorithm. This is a more natural model for representing the environment, and should simplify the task of planning force trajectories. The algorithm should compensate for Jacobian singularities, and use the gripper position, $x_g$, directly.

The loss of passivity is an important problem in robotics, since it can lead to instability. In cases where it is not possible to obtain a passive controller architecture for the system, it is necessary to guarantee stability by other means. Because the loss of passive behavior is typically observed only at high frequencies, it is reasonable to assume that singular perturbation methods could be applied to the problem. By guaranteeing stability for the low frequency portion using passivity methods, and using the small gain theorem to guarantee stability for the high frequency portions, some type of overall stability should be attainable. Unfortunately, due to the nonlinear nature of the manipulator, and the possibility of arbitrary environments, this is not an easy proposition to prove.

The passivity based control laws presented here have been developed using continuous time theory. Implementations of these control laws, however, are done in discrete time. In most instances no difficulty arises, since suitably fast sampled-data representations can approximate the continuous time control laws for the band-limited robot. In the case of the teleoperator, however, the time-delay compensation introduced in Chapter 5 has some implementation problems. The discrete time control law used in the experimental simulation does not respond as the continuous time theory predicts. Although stable behavior is obtained for arbitrarily large time delays, good tracking is lost as the time delay exceeds 200 msec. More work needs to be done to analyze these effects and to determine if a better discretization routine can be obtained.

Two types of sensory feedback have been considered in this thesis: position (velocity) and force feedback. Other types of sensory feedback are available to the robot, however, such as vision, proximity sensors, and tactile sensors. A development of network theory based on these additional sensors would greatly enhance the growth of sophisticated robot controllers.

The modelling of flexible link behavior represents another possible benefit of the Hilbert network theory. The flexible link models that have been developed to date consider only one
or two links of the robot to be flexible (Krishnan and Vidyasagar 1988, Eppinger and Seering 1988), and the models for the flexibility used are typically linear Euler-Bernoulli beam equations. Actual robots, however, may have flexibility in every beam, and the type of flexibility includes torsional modes and complex coupled nonlinear behavior. By using Hilbert network elements in infinite dimensional spaces, it should be possible to model complex flexible behavior, without first having to develop the underlying equations.

Network approaches can also be used for understanding adaptive control in continuous time. It is interesting to note, that the gradient form of parameter estimation (Anderson et al. 1986) is directly representable in a network diagram. Let

$$f(s) = Z(s)v(s) = \phi^T(s)\theta,$$

where \(v(s)\) is the input to the plant, \(f(s)\) is the output to the plant, \(Z(s)\) is the strictly positive real representation of the plant, \(\phi\) is the regressor vector, and \(\theta\) is the parameter vector. Let \(\hat{\theta}\) represent the parameter estimate, and consider the parameter update law,

$$\frac{d}{dt}\hat{\theta} = \epsilon \phi(f)\phi^T(\theta - \hat{\theta}).$$

This system can be represented by the network shown in Fig. 6.1.

![Network diagram](image)

**Fig. 6.1.** Network model for the gradient method of adaptive control.

From this model it is clear that \(\phi\) must span the entire subspace of the stiffness element in order for the estimate, \(\hat{\theta}\), to approach the actual parameter vector \(\theta\) (i.e., be persistently exciting). It is reasonable to expect that network representations of other parameter estimation and adaptive control approaches exist, since the primary condition needed in adaptive control, i.e., the strictly positive real (SPR) condition, is the same condition needed for realizing a passive network.

Besides representing existing adaptive control structures, new adaptive controller approaches might be suggested based on the network model. For instance, most adaptive control methods assume that unknown parameters are fixed or slowly varying, but as is evident
from the network model of a system, many parameters can be varied quickly. Consider the following scenario. A robot with one DOF is controlled to be critically damped while executing position motions by the use of a PD control law. When the robot contacts a stiff environment, however, the total stiffness in the control loop is increased, resulting in an underdamped system. When this contact is detected, the damping gain should be increased quickly to avoid overshoot and oscillations. This represents an adaptive control. Because of the passive structure, the system will remain asymptotically stable, as long as the damping remains positive. In the PHIDE network representation of the system, any of the transformers and any of the damping terms can be varied without affecting the system's stability, assuming that positive damping is maintained. This conclusion is reached by using the energy function for a Lyapunov function for the system, and then applying Tellegen's theorem to the derivative of the Lyapunov function.

6.3 A Final Note

It is perhaps fitting that the Hilbert network approach presented here has been developed for the problems which arise in force control of robotics and teleoperation. The level of systems integration required in robotics is possibly greater than in any other discipline. Controlling forces in particular, involves contact with environments as diverse as satellites, grinding tools, and paint brushes. Furthermore, the area of control theory is integrally related with system modeling. No control can be developed, unless a model for the system is initially obtained. Thus, a modeling technique, which is so closely related to the control design, is no surprise. It is hoped that the ideas presented here can lead to many new discoveries, not only in robotics, but also in general systems integration.
BIBLIOGRAPHY


VITA

Robert Anderson was born in Saint Joseph, Michigan, on May 26, 1961. He received the B.S. degree, Magna Cum Laude in Applied Mathematics and Engineering Science from the University of California, San Diego, in 1983. He was awarded the M.S. degree in Electrical Engineering from the University of Illinois at Urbana-Champaign in 1986.

As an undergraduate, Robert worked with the Naval Ocean Systems Center and ORINCON Corp. in San Diego, California. Since 1985, he has been a Teaching and Research Assistant at the University of Illinois, while spending summers working at the Jet Propulsion Laboratory in Pasadena, California.

Robert received the "Best Student Paper Award" at the 1988 IEEE Conference on Decision and Control in Austin, Texas, for his paper entitled, "Bilateral Control of Teleoperators with Time Delay." He is a member of Phi Beta Kappa and IEEE.

PUBLICATIONS


